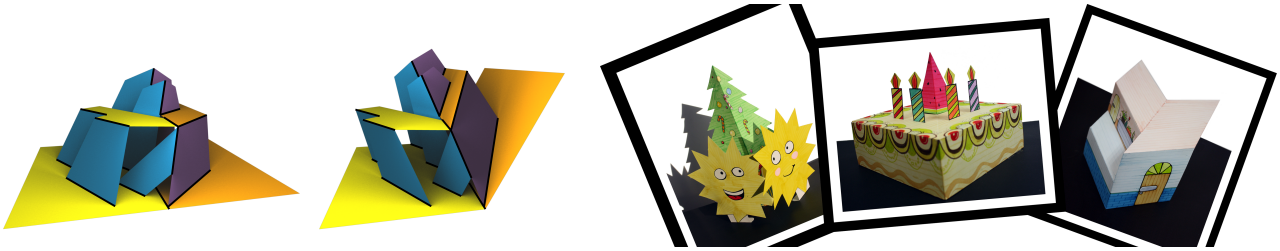
A Geometric Study of V-style Pop-ups: Theories and Algorithms

Xian-Ying Li[1] Tao Ju[2] Yan Gu[1] Shi-Min Hu[1]
[1]Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing
[2]Department of Computer Science and Engineering, Washington University in St. Louis



(a) (b) (c)

Figure 1: *Left (a,b): a v-style pop-up at its fully opened state (a), and an intermediate state of closing (b). Right (c): actual handmade v-style popups guided by our theories.*

Abstract

Pop-up books are a fascinating form of paper art with intriguing geometric properties. In this paper, we present a systematic study of a simple but common class of pop-ups consisting of patches falling into four parallel groups, which we call v-style pop-ups. We give sufficient conditions for a v-style paper structure to be pop-uppable. That is, it can be closed flat while maintaining the rigidity of the patches, the closing and opening do not need extra force besides holding two patches and are free of intersections, and the closed paper is contained within the page border. These conditions allow us to identify novel mechanisms for making pop-ups. Based on the theory and mechanisms, we developed an interactive tool for designing v-style pop-ups and an automated construction algorithm from a given geometry, both of which guaranteeing the pop-uppability of the results.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems;

Keywords: pop-up, computer art, geometric modeling

Links: DL PDF WEB

1 Introduction

If books are windows to the world, then pop-up books are probably the most beautiful and delicate ones. With special paper mechanisms, vivid 3D scenes may "jump out" from a pop-up book and also be flattened and stored in pages when the book is closed (Figure 2). The history of this kind of "movable" books can be traced back to the Catalan mystic and poet Ramon Llull in the early 14th century. Today's pop-up books continue to grab the fascination of readers worldwide, children and adults alike, with amazing titles authored by artists like Robert Sabuda, David Carter and Matthew Reinhart.

Manual design of pop-ups has been mostly based on experience and trial-and-error. A number of basic mechanisms for creating simple pop-ups have been introduced by artists based upon their experiences [Hiner 1985; Birmingham 1997; Carter 1999]. However, putting these mechanisms together to build a desirable pop-up is never an easy task. Even for experienced masters, it would take months of work to complete the designs in a pop-up book. Part of the difficulty is that human experiences quickly become insufficient to tell if a design can be correctly "popped-up" once the design gets slightly more complex than just a few basic mechanisms. The only way to verify a design is to actually make the pop-up by paper and try folding it, which is an extremely time-consuming process.

From a geometric perspective, there are number of essential and intriguing properties of a pop-up:

1. The pop-up can be closed down to a flat surface and opened up again without tearing the paper or introducing new creases other than those in the design.
2. The closing and opening of the pop-up do not need extra forces other than holding and turning the two book pages.
3. The paper does not intersect during closing or opening.
4. When closed, all pieces of the pop-up are enclosed within the book page.

There has only been limited study of pop-ups as a geometric problem. Existing works focus on the analysis of a small set of known mechanisms (e.g., v-folds), with the objective of providing interactive design environments that replace actual paper-making during the design process by virtual simulations. However, little effort has been made to understand the geometric properties that a collection of paper pieces should possess in order to be "pop-uppable". Without such study, computer-assisted tools at best offer faster feedback in the trial-and-error design, but cannot give intelligent advice as *how* a design can be improved to satisfy the desired properties, or offer guarantees on the validity of a design.

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

Figure 2: *Examples from pop-up books.*

Our long-term goal is to answer the following question: what type of paper structure admits a folding process with the four properties listed above? This question has been recently considered for some simple types of pop-ups, such as those consisting of two groups of parallel pieces [Li et al. 2010] or having a 2-dimensional structure [Hara and Sugihara 2009] (see review next). In this paper, we consider a more general and common class of pop-ups, which we call *v-style* pop-ups. Such pop-ups consist of planar paper pieces that stay parallel with either the two book pages or with two other additional moving planes when the pop-up is being opened or closed. A schematic example is shown in Figure 1 (a,b) in two different states during the closing process. The paper pieces are colored by the different parallel groups. Note that the pop-up has a "V" shape, due to the additional two moving planes (blue and purple) besides the two book pages (yellow and orange).

We have observed that the v-style is a common motif in pop-up books. Some pop-ups consist solely of a v-scaffold (e.g., Figure 2(a)), while others often contain a v-scaffold as its back-bone structure and additional decorating pieces attached on the scaffold (e.g., the waves in Figure 2(b), round badges in Figure 2(c), and glasses and wheels in Figure 2(d)).

Contributions Our main contribution is a theoretical study of the geometric structure of v-style pop-ups. We give explicit, verifiable conditions under which a given paper structure admits a folding process with the four properties listed above (Section 3). With these conditions, we explore a suite of pop-up building mechanisms, many of which have not been reported before (Section 3.3, mechanisms S1, S2, D3). To our knowledge, this is the first study of pop-up geometry that is not based on known pop-up mechanisms. Guided by the theoretical analysis, our algorithmic contributions include an interactive tool for creating v-style pop-ups that guarantees the validity of the design, and an automated algorithm for constructing a v-style pop-up from a given 3D model (Section 4).

1.1 Related works

Paper crafting We first review some related forms of paper crafting that have attracted computational studies.

Origami is the traditional Japanese art of paper folding. The central problem in origami is folding and foldability, which has been extensively studied in the literature [Hull 2006; Demaine and O'Rourke 2007; O'Rourke 2011]. Different from origami, pop-ups can be made by cutting and gluing multiple pieces together, hence possessing a richer geometric structure. On the other hand, the closing process of a pop-up is more restricted than origami, as it requires all pieces to flatten simultaneously and with only forces on two pieces (the book pages). As a result, the theories related to these two art forms are likely to differ significantly.

Paper architecture (PA) is a special type of pop-up that is made from a single piece of paper by cutting and folding. A class of paper architecture that has attracted much attention in the past is *parallel PA*, whose pieces maintain parallel to the two book pages during opening and closing. The simple mechanisms of parallel PA enabled development of automated algorithms that construct a PA from a given 3D model [Mitani et al. 2003; Li et al. 2010] as well as interactive design tools [Mitani and Suzuki 2004]. In particular, Li et al. [Li et al. 2010] studied the sufficient conditions for a parallel PA to be stable during opening or closing, which we extend onto the much general class of v-style pop-ups. In contrast to parallel PAs, v-style pop-ups allow multiple pieces of paper and contain two more parallel groups of planes. These differences make problems like foldability and collision significantly harder to address on a v-style pop-up, which is the focus of our paper.

Computational pop-ups The computational literature on pop-ups has been scarce at best. Existing works focus almost exclusively on the few known mechanisms, and in particular the v-fold [Lee et al. 1996; Glassner 2002b] and its variants, such as lattice-type folds [Mitani and Suzuki 2003] and cube-type folds (with open or closed tops) [Okamura and Igarashi 2009]. These studies lead to interactive systems for virtually building pop-ups using these mechanisms and simulating the opening/closing of the results [Glassner 2002a; Glassner 2002b; Hendrix and Eisenberg 2006]. The recent system of Okamura and Igarashi [2009] further provides user feedback of whether the closing simulation of the pop-up encounters intersections. Note that all these systems are limited to the known mechanisms, and do not offer validity guarantees on the designs.

There has been little research into the geometric structure of a general pop-up paper. The difficulty facing such endeavor is partly revealed by the work of Uehara and Teramoto [2006], who showed that determining the foldability of a general pop-up is NP hard. The only other work we know in this direction is that of Hara and Sugihara [2009], who studied pop-ups with a 2-dimensional structure (that is, all pieces are parallel to the central seam between the two book pages), and developed an algorithm that constructs a pop-up with any given exterior shape by a sequence of v-folds. In contrast, a v-style pop-up has a much richer, 3-dimensional geometric structure, and hence the geometric problems such as foldability and intersections are much more involved than on a 2-dimensional pop-up. In this paper, we aim at developing *sufficient* conditions for a v-style structure to be pop-uppable, which leads to interactive tools and automated algorithms with guaranteed validity of the result.

2 Definitions

We first introduce geometric formulations of a paper (called a *scaffold*) and the properties on this paper that would make it "pop-uppable". To simplify analysis, we inherit the same assumptions

as in previous computational studies on paper art [Belcastro and Hull 2002; Li et al. 2010] that the paper has zero thickness, zero weight, and is rigid (i.e., cannot be warped). Note that, in practice, the physical property of the paper does play an important role in making paper pop-ups. Also, there are many pop-up artworks that involve warping of the paper. Nevertheless, we believe that the formulations and analysis here using the simplified assumptions would build the basis for studying these other more complex scenario in the future.

2.1 Scaffold

A *scaffold* is a collection of planar polygons, called *patches*, that are connected at straight line segments, called *hinges*. The hinges may lie either interior to a patch or on its border. A scaffold always contains two patches, known as *ground* and *backdrop* (the two pieces held in hand), which are two identical rectangles abutting on an edge that is called a *center hinge*. The angle between these two patches is called the *fold angle*.

To maintain generality, we permit the patches in the scaffold to overlap or intersect away from the hinges. However, points on different patches are topologically distinct even if they share the same spatial location, except when they are located on the hinges. In other words, the scaffold is a 2D structure immersed (and not necessarily embedded) in 3D. This relaxation allows us to examine intersection-free property of a folding process independently from other properties, such as foldability.

2.2 Pop-uppable scaffold

During the design process, the user typically creates the pop-up in its opened state (the state that is being viewed). Hence we are interested in knowing whether a scaffold representing the open-state pop-up can be *closed* with the four properties mentioned in the Introduction. In the following, we formulate these properties as conditions of a transformation (called *fold transform*) on a scaffold.

Definition 1 *A* **fold transform** $f(S,t)$ *on a scaffold* S *is a continuous deformation of* S *where the deformation is identity when* $t = 0$ *and is a combination of translations and rotations on each patch of* S *for any* $t \in [0,1]$.

Note that a fold transform necessarily maintains the rigidity and hinge connectivity of the patches. The first property in the Introduction can thus be phrased as a fold transform that closes the scaffold to a flat plane:

Definition 2 *A* **flattening transform** $f(S,t)$ *on a scaffold* S *is a fold transform with two additional properties:*

1. *The fold angle decreases monotonically to 0 as* t *increases from 0 to 1.*
2. *All patches in* $f(S,1)$ *are co-planar.*

To describe the second property, we formulate a stability condition that asks every intermediate state of the transform to be stable with respect to the ground and backdrop (similar to that in [Li et al. 2010]):

Definition 3 *A fold transform* $f(S,t)$ *on a scaffold* S *is said to be* **stable**, *if for any* $t \in (0,1)$, *there does not exist any non-identity fold transform on* $f(S,t)$ *that keeps the ground and backdrop in* $f(S,t)$ *still.*

The third and forth properties can be formulated as follows:

Definition 4 *A fold transform* $f(S,t)$ *on a scaffold* S *is said to be* **intersection-free**, *if for any two topologically distinct points* p, q *on* S, *their deformed locations on* $f(S,t)$ *are spatially distinct for any* $t \in (0,1)$.

Definition 5 *A flattening transform* $f(S,t)$ *on a scaffold* S *is said to be* **enclosing**, *if all patches in* $f(S,1)$ *lie interior to the ground patch.*

Note that, by these definitions, the inverse of a stable, intersection-free flattening transform is another stable, intersection-free fold transform with a monotonically increasing fold angle. In other words, if a scaffold S has a flattening transform that is stable, intersection-free and enclosing, it can be closed down as well as opened up with the four properties enlisted in the Introduction. We call such scaffold *pop-uppable*.

3 Theoretical foundation

In this section, we examine a particularly simple, yet expressive pop-up style – the *v-style pop-up*. As noted before, such a pop-up contains up to four groups of planar pieces such that pieces within each group stay parallel with each other throughout the closing (and opening) process. This section will attempt to answer the question *what kind of scaffold is poppupable in v-style?*

We start by defining a special class of scaffold, the *v-scaffold*, which describes the open state of a v-style pop-up. In the main part of this section, we give sufficient, verifiable conditions for a v-scaffold to have a flattening transform in v-style that is stable, intersection-free, and enclosing.

3.1 V-scaffold

A *v-scaffold* is a scaffold where each patch is labelled as either G, B, L, R, such that patches with labels G and B are parallel to the ground and backdrop respectively, patches with labels L and R are parallel to two additional planes (called *left* and *right* planes), and the ground and the backdrop are respectively labelled G and B. To avoid degeneracy, we ask that the left and right planes are not parallel to each other or to the central hinge, and that the hinges are located only between patches with different labels.

An example of v-scaffold is shown in Figure 1 (a), where different labels are indicated by colors. Note that while we show scaffolds with fold angle π in all figures, the results in this section are applicable to any positive fold angle no greater than π.

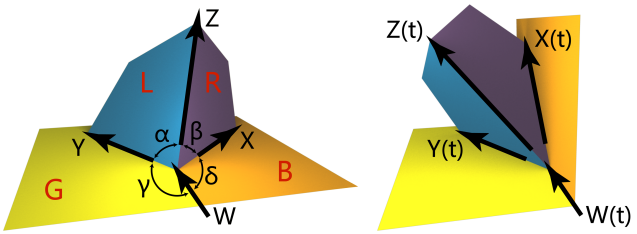


Figure 3: *(a) The patch labels, axes, and their angles in a v-scaffold. (b) Vectors* $X(t), Y(t), Z(t), W(t)$ *at* $t = 0.5$.

In the following discussion, we will utilize four vectors X, Y, Z, W that are essential to the structure of a v-scaffold, as illustrated in Figure 3. Here, X is parallel to both the right plane and the backdrop, Y is parallel to both the left plane and the ground, Z is parallel to both the left and right planes, and W is parallel to the central hinge. We orient them so that, originating from some point on the central hinge, X and Y respectively point into the half-plane of B

and G, Z points towards above the ground and backdrop, and W points away from the user. Each vector has unit length, and they are called the *axes* of the v-scaffold. In addition, we call axes pairs $\{Y, W\}$, $\{X, W\}$, $\{Y, Z\}$, and $\{X, Z\}$ respectively the *supporting axes* of patches with labels G, B, L, R, and *opposing axes* of patches with labels R, L, B, G.

3.2 Flattening transforms

The first question we consider is: can a v-scaffold be folded flat while maintaining its rigidity and hinge connectivity (i.e., the first property in Introduction)? Without simulations, this can be difficult to answer even for simple v-scaffolds. For example, the two v-scaffolds in Figure 4 look quite similar. The only difference is that the blue and purple patches are disjoint in (a), but are connected with a hinge (the one in red) in (b). However, the one in (a) has a flattening transform while the one in (b) doesn't.

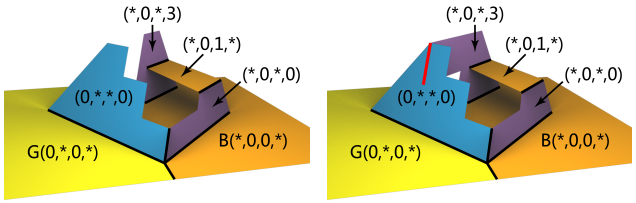


Figure 4: *A v-scaffold that satisfies the Decomposition Condition (a) and another that does not (b). The v-scaffold in (b) connects the blue and purple patches, which are disjoint in (a).*

The following gives a geometric answer to the question:

Proposition 1 *A v-scaffold S has a flattening transform, and patches with identical labels maintain parallel to each other during the transform, if the following two conditions hold:*

1. ***[Angle Condition]*** *Let the angles between the four axes be $\alpha, \beta, \gamma, \delta$ (see Figure 3). They satisfy $\gamma + \delta > \pi$, $\alpha + \beta + \gamma + \delta > 2\pi$, and $\gamma - \alpha = \delta - \beta > 0$.*

2. ***[Decomposition Condition]*** *Each point p on S can be decomposed into a linear combination of the four axes,*

$$p = x_p X + y_p Y + z_p Z + w_p W, \tag{1}$$

such that the coefficients $\{x_p, y_p, z_p, w_p\}$ are continuous on S, and that points on a same patch share the same coefficients on the two opposing axes of that patch.

Proof: First of all, we will construct four vector functions: $X(t), Y(t), Z(t), W(t)$ for $t \in [0, 1]$, such that $Y(t) = Y$, $W(t) = W$, $X(t)$ is rotated from X around axis W by angle $-\theta * t$ where θ is the fold angle of S, and $Z(t)$ is an vector spanning angle α, β with $Y(t), X(t)$ respectively. It can be verified that the Angle Condition above ensures the existence of $Z(t)$ for $t \in [0, 1]$, and that $Z(1)$ (as well as $X(1), Y(1), W(1)$) is parallel to the ground plane. Note that the four vectors $X(t), Y(t), Z(t), W(t)$ maintain constant pairwise angles $\alpha, \beta, \gamma, \delta$ for $t \in [0, 1]$. Figure 3 (b) illustrates these vectors for $\theta = \pi$ and $t = 0.5$.

Next, we show that the following deformation for each point p on S is a flattening transform in v-style:

$$f(p, t) = x_p X(t) + y_p Y(t) + z_p Z(t) + w_p W(t) \tag{2}$$

We first show f is a fold transform. It is easy to see that $f(p, 0) = p$ for any p. The continuity of the coefficients $\{x_p, y_p, z_p, w_p\}$ in the Decomposition Condition ensures that f maintains the hinge connectivity of S. To show patch rigidity, consider two points p, q on a same patch in S. Due to Decomposition Condition, the difference vector $f(p, t) - f(q, t)$ can be represented as linear combination of two of the vector functions $(X(t), Y(t), Z(t), W(t))$ whose coefficients are invariant for $t \in [0, 1]$. Since the four vectors maintain their lengths and pairwise angles, $f(p, t) - f(q, t)$ has an invariant magnitude. Hence f is a fold transform. To see that f has a v-style, note that patches with identical labels will stay parallel to two of the vector functions corresponding to the supporting axes of those patches. Finally, f is a flattening transform because the dihedral angle between the plane formed by $X(t), W(t)$ and the plane formed by $Y(t), W(t)$ decreases to 0 at t increases to 1, and all four vectors $X(t), Y(t), Z(t), W(t)$ are parallel to a same plane at $t = 1$. □

Similar conditions to the Angle Condition has been studied before for simple *v-folds* [Huffman 1976; Lee et al. 1996; Glassner 2002b], which consist of only four patches (e.g., Figure 3 (a)). Our novel contribution is the Decomposition Condition, which offers key insights on when a complex v-scaffold can be folded flat. Intuitively, the decomposition in Equation 1 "lifts" the scaffold S to a 4-dimensional space where $\{x_p, y_p, z_p, w_p\}$ are the coordinates of p. Note that such lifting is not unique: p can be lifted to any point along a 4D hyper-line. Let $\{a, b, c\}$ be scalars satisfying $aX + bY + cZ + W = 0$. Then p can be represented by coordinates $\{x_p, y_p, z_p, w_p\} + u * \{a, b, c, 1\}$ for any u. The Decomposition Condition requires that the lifting maintains the continuity of the patches in S, and that each lifted patch (a hyper-surface) is parallel to two of the four axes in 4D.

While the Angle Condition in Proposition 1 is easy to verify for a given v-scaffold, it is not immediately clear how to check the Decomposition Condition. Next we present an equivalent condition that gives rise to an explicit checking procedure using the patch labels and connectivity in an input v-scaffold. The same condition will help us later to discover means to design a v-scaffold that is equipped with a flattening transform by construction.

Let us introduce some further notions. Consider a patch h in a v-scaffold S, and denote its two opposing axes as Φ, Ψ. Two scalars $\{\phi_h, \psi_h\}$ are called *coefficients* of h if the point $\phi_h \Phi + \psi_h \Psi$ lies on the supporting plane of h. Note that if S satisfies the Decomposition Condition, the two common coefficients shared by points on h become the coefficients of h. Note also that, once the coefficient $\{\phi_h, \psi_h\}$ of h is fixed, the coefficients of all points on h that have values $\{\phi_h, \psi_h\}$ on the axes Φ, Ψ are uniquely determined. Finally, let us call the labels in each pair $\{G, L\}$, $\{L, R\}$, $\{R, B\}$, $\{G, B\}$ *adjacent labels*, and labels in each pairs $\{G, R\}$, $\{L, B\}$ *opposite labels*. It is not difficult to derive (see Appendix A) that:

Corollary 1 *A v-scaffold S meets the Decomposition Condition if and only if there exist some coefficients for each patch of S such that, for any two patches g, h connected by a hinge,*

1. *g, h must have adjacent labels.*

2. *The hinge is parallel to the axis supporting both g, h.*

3. ***[Hinge Condition]*** *Let g have coefficients $\{\phi_g, \omega_g\}$ on its opposing axes Φ, Ω, and h have coefficients $\{\psi_h, \omega_h\}$ on its opposing axes Ψ, Ω, where Ω is the common axis opposing both g, h. Then $\omega_g = \omega_h$, and the point $\phi_g \Phi + \psi_h \Psi + \omega_h \Omega$ lies on the supporting line of the hinge.*

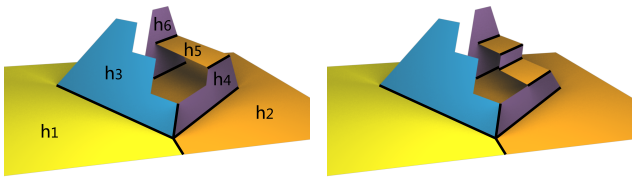
To utilize Corollary 1, the key observation is that if one of the two patches connected by a hinge (assuming conditions (1,2) are satisfied) has known coefficients, the coefficients of the other patch satisfying the Hinge Condition is uniquely determined. Another useful observation is that if a set of patch coefficients satisfying

Corollary 1 exists, there exists such a satisfying set for any possible coefficients of a particular patch (in the "lifting" analogy, the lifted scaffold in 4D can be freely translated without violating continuity and planarity). So checking with Corollary 1 (and hence the Decomposition Condition) on a v-scaffold S can proceed as follows. First, determine that conditions (1,2) are satisfied. Next, starting with any coefficients of one patch, determine the coefficients of the connecting patches, and iterate till all patches are associated with some coefficients. Then check the Hinge Condition for each pair of connecting patches. If the checks all pass, then the scaffold meets the Decomposition Condition; otherwise it does not.

As examples, we apply this checking procedure to the two examples in Figure 4. In each case, we assume the origin lies at some point on the central hinge, and start by giving coefficients $\{0, 0\}$ to the ground patch. The set of patch coefficients computed in the iterative manner in (a) satisfy the Hinge Condition for each pair of connected patches, but the ones in (b) fail the condition between the blue and purple patches (whose hinge is highlighted in red). We label each patch's coefficients by a 4-tuple with a star (*) indicating the supporting axis of the patch (of which no coefficient is defined).

3.3 Stability

While the Angle and Decomposition conditions ensure that the v-scaffold can be folded flat in a rigid and connected way, the folding may need additional force other than fixating the ground and backdrop. For example, without additional force, the staircases in Figure 5 (b) would collapse.



(a) (b)

Figure 5: *A v-scaffold that satisfies the stability condition in Proposition 2 (a) and another one that does not (b).*

The next condition, which is similar to the one proposed for stable paper architecture [Li et al. 2010], further ensures that the folding is force-free besides holding the ground and backdrop pieces:

Proposition 2 *Let S be a v-scaffold satisfying the conditions in Proposition 1. The fold transform of Equation 2 is stable, if there is an ordering of patches in S, $\{h_1, h_2, ...\}$ such that h_1, h_2 are the ground and backdrop patch, and that for every $k > 2$,*

1. *Either h_k has non-co-linear hinges with two patches h_i, h_j where $i, j < k$, or*
2. *h_k has a hinge with h_{k+1}, and the two patches have hinges respectively with h_i, h_j where $i, j < k$, and none of these hinges are co-linear.*

Proof: The proof is similar to the one in [Li et al. 2010]. The key is to observe that a patch h_k in the first case cannot be deformed in a rigid way if both h_i, h_j are kept still, and neither can a pair of patches h_k, h_{k+1} in the second case if both h_i, h_j are held still. By induction, if the ground and backdrop are held still at $t = 0$, no other patch of S can be deformed rigidly. To show this is true for all $t < 1$ (hence the fold transform is stable), we need to show that the non-co-linearity among the hinges is preserved in both cases above during fold transform. Since the two non-co-linear hinges on a same patch stay non-co-linear during the fold transform, we only need to show the hinges between h_k, h_i and between h_{k+1}, h_j in the second case cannot become co-linear during the fold transform. This is true, because otherwise h_k, h_{k+1} would become co-planar, implying three of the four vector functions $X(t), Y(t), Z(t), W(t)$ are co-planar, which is not possible for any $t \in (0, 1)$ based on our definition of these functions.

In the examples of Figure 5, the patches in (a) have an ordering that satisfies the stability condition above (as labelled in the picture), but the ones in (b) don't. Hence the v-scaffold in (a) can be stably flattened. In general, the condition can be verified on any given v-scaffold by exhaustively examining possible ordering of the patches.

The formulation of Proposition 2 lends naturally to a constructive way of building a stable v-scaffold. Combining with the results before, we next explore mechanisms of taking an existing v-scaffold S that has a stable, flattening transform (e.g., one that contains only the ground and backdrop), and adding new patches so that the updated scaffold also can be stably flattened. Each mechanism considers two patches of S and connects them with either one or two new patches, following the two cases of Proposition 2, provided that the labels and coefficients of these patches meet the stated conditions. Examples of these mechanisms are illustrated in Figure 6.

Single-patch mechanisms Each of these mechanisms connects two patches g_1, g_2 of S with a new patch h if the stated conditions are satisfied.

Mechanism S1: g_1, g_2 have identical labels, and the label of h is adjacent to that of g_1, g_2. Denote the opposing axes of g_1 (and g_2) as Φ, Ψ, the coefficients of g_1 as $\{\phi_{g1}, \psi_{g1}\}$, and the coefficients of g_2 as $\{\phi_{g2}, \psi_{g2}\}$. Denote the opposing axes of h as Φ, Ω (sharing Φ with g_1), and its coefficients as $\{\phi_h, \omega_h\}$. They need to satisfy:

$$\phi_{g1} = \phi_{g2} = \phi_h, \quad \psi_{g1} \neq \psi_{g2}$$

That is, given g_1, g_2 satisfying $\phi_{g1} = \phi_{g2}$ and $\psi_{g1} \neq \psi_{g2}$, any h is connectable to both g_1, g_2 (since only one of h's two coefficients, ϕ_h, is constrained).

Mechanism S2: g_1, g_2 have opposite labels, and the label of h is adjacent to both of g_1, g_2. Denote the opposing axes of g_1 as Φ, Ψ and its coefficients $\{\phi_{g1}, \psi_{g1}\}$, the opposing axes of g_2 as Ω, Υ and its coefficients $\{\omega_{g2}, \upsilon_{g2}\}$, and the opposing axes of h as Φ, Ω (sharing Φ with g_1 and Ω with g_2) and its coefficients $\{\phi_h, \omega_h\}$. They satisfy:

$$\phi_{g1} = \phi_h, \quad \omega_{g2} = \omega_h$$

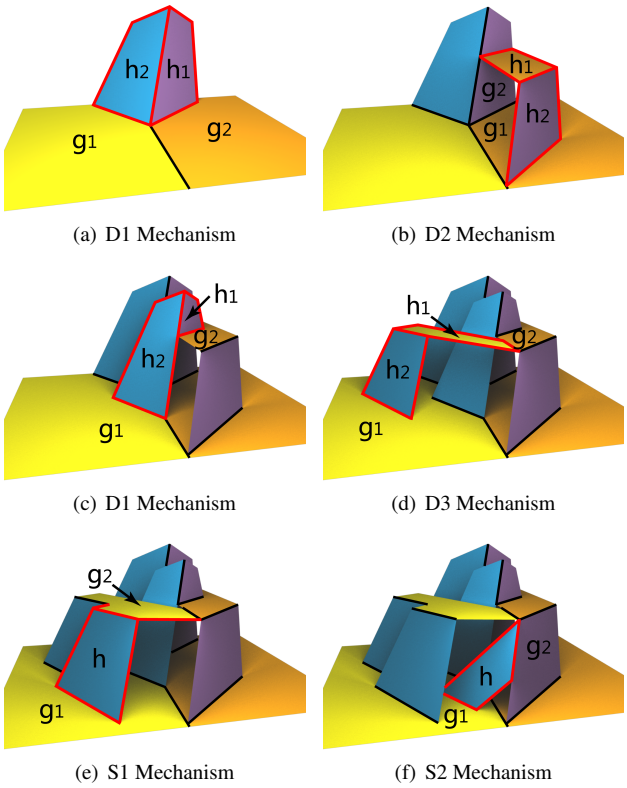
That is, given any g_1, g_2, the connectable h is uniquely determined (since both of h's two coefficients are constrained).

Double-patch mechanisms Each of these mechanisms connects two patches g_1, g_2 of S whose labels are adjacent, with two new patches h_1, h_2 if the stated conditions are satisfied. Denote the opposing axes of g_1 as Φ, Ψ and its coefficients $\{\phi_{g1}, \psi_{g1}\}$, and the opposing axes of g_2 as Φ, Ω and its coefficients $\{\phi_{g2}, \omega_{g2}\}$.

Mechanism D1: h_1 has the opposite label with g_1, and h_2 has the opposite label with g_2. Denote the opposing axes of h_1 as Υ, Ω and its coefficients $\{\upsilon_{h1}, \omega_{h1}\}$, and the opposing axes of h_2 as Υ, Ψ and its coefficients $\{\upsilon_{h2}, \psi_{h2}\}$. They satisfy:

$$\psi_{g1} = \psi_{h2}, \quad \omega_{g2} = \omega_{h1}, \quad \upsilon_{h1} = \upsilon_{h2}$$

That is, given any g_1, g_2 and any scalar υ_{h1}, the connectable h_1, h_2 are uniquely determined.



(a) D1 Mechanism (b) D2 Mechanism

(c) D1 Mechanism (d) D3 Mechanism

(e) S1 Mechanism (f) S2 Mechanism

Figure 6: *The double-patch mechanisms (a,b,c,d) and single-patch mechanisms (e,f). The newly added patches are highlighted.*

Mechanism D2: h_1 has the same label with g_1, and h_2 has the same label with g_2. Denote the coefficients of h_1 as $\{\phi_{h1}, \psi_{h1}\}$, and the coefficients of h_2 as $\{\phi_{h2}, \omega_{h2}\}$. They satisfy:

$$\phi_{g1} = \phi_{g2} = \phi_{h1} = \phi_{h2}, \;\; \psi_{g1} \neq \psi_{h1}, \;\; \omega_{g2} \neq \omega_{h2}$$

That is, given g_1, g_2 satisfying $\phi_{g1} = \phi_{g2}$, any h_1, h_2 are connectable to g_1, g_2.

Mechanism D3: h_1 has the same label with g_1, and h_2 has the opposite label with g_2. Denote the coefficients of h_1 as $\{\phi_{h1}, \psi_{h1}\}$, the opposing axes of h_2 as Υ, Ψ and its coefficients $\{\upsilon_{h2}, \psi_{h2}\}$. They satisfy:

$$\psi_{g1} = \psi_{h1} = \psi_{h2}, \;\; \phi_{g1} \neq \phi_{g2} = \phi_{h1}$$

That is, given g_1, g_2 satisfying $\phi_{g1} \neq \phi_{g2}$ (complementary to Mechanism D2), the connectable h_1 is uniquely determined while any h_2 is connectable.

Mechanisms D1 and D2 are commonly known as *v-fold* and *box-fold*. Note that our definition of D1 is more general than the common v-fold, in that the two existing patches g_1, g_2 need not to be connected by a hinge (compare Figure 6 (a) and (c)). The other mechanisms (S1, S2, D3) are novel means of pop-up that have not been reported before. It is easy to verify (via Corollary 1) that, if S satisfies the conditions in Propositions 1 and 2, the new scaffold following these mechanisms still meets those conditions.

3.4 Intersections and enclosure

The v-scaffold constructed using the mechanisms above may not be able to realize in practice, due to possible intersections during the fold transform. Also, the flattened scaffold may not be completely enclosed within the border of the ground (and backdrop). Nonetheless, the explicit form of the fold transform (Equation 2) allows us to devise explicit conditions that check for possible intersections during the transform and enclosure at the flattened state *without* need for simulation.

Let S be a v-scaffold satisfying the conditions in Proposition 1. The transform of Equation 2 is enclosing if every point p with coefficients $\{x_p, y_p, z_p, w_p\}$, the point $x_p X(1) + y_p Y(1) + z_p Z(1) + w_p W(1)$ lies interior to the ground patch. Note that this is straightforward to check as the vectors $X(1), Y(1), Z(1), W(1)$ are explicitly known.

Furthermore, a pair of distinct points p, q on S will come to intersection during the fold transform of Equation 2 if and only if the difference in their coefficients, denoted as $\{x', y', z', w'\}$, satisfy:

$$x'X(t) + y'Y(t) + z'Z(t) + w'W(t) = 0 \qquad (3)$$

for some $t \in (0, 1)$. If no such pair of points exist, the fold transform is intersection-free. Using the geometric relationship between the vectors $X(t), Y(t), Z(t), W(t)$, we can remove the dependency of t in the above equation and formulate explicit criteria involving only $\{x', y', z', w'\}$ (see proof in Appendix B):

Proposition 3 *Let S be a v-scaffold satisfying the conditions in Proposition 1. Two disjoint points p, q on S with coefficient difference $\{x', y', z', w'\}$ will meet during the fold transform of Equation 2 if and only if all of the following hold:*

1. *Either* $\{x' > 0, y' > 0, z' < 0, w' < 0\}$ *or* $\{x' < 0, y' < 0, z' > 0, w' > 0\}$
2. $x'^2 + x'z' \cos\beta - x'w' \cos\delta = y'^2 + y'z' \cos\alpha - y'w' \cos\gamma$ *(*$= \Theta$*, as a notation)*
3. $z'^2 + x'z' \cos\beta + y'z' \cos\alpha = w'^2 - x'w' \cos\delta - y'w' \cos\gamma$
4. Θ *satisfies:* $-x'y' \cos(\gamma - \delta) < \Theta < -x'y' \cos(\gamma + \delta)$

4 Computer-assisted design

Using the results developed above, we next present an interactive graphical tool as well as an automated algorithm for designing v-style pop-ups. To simplify the task of intersection checking, we consider a special configuration of the axes in a v-scaffold such that X, Y, Z are mutually orthogonal and aligned with the three Cartesian axes, and W lies on the angular bisector of X, Y. We call this configuration a *Cartesian* v-scaffold. As we see next, this configuration leads to simple criteria of point-wise intersections and efficient means for checking patch-wise intersections.

Based on the intersection criteria, we develop an interactive tool where the user can build up complex Cartesian v-scaffolds using the mechanisms introduced in Section 3.3, while being ensured that the result is also intersection-free and enclosing. Finally, we devise an automated procedure of applying these mechanisms that generate a v-style pop-up reproducing a given voxelized model.

4.1 Intersection checking

In a Cartesian v-scaffold, $\alpha = \beta = \pi/2$ and $\gamma = \delta = 3\pi/4$. Given two points p, q with coefficient differences $\{x', y', z', w'\}$, the intersection criteria of Proposition 3 simplify to:

1. Either $\{x' > 0, y' > 0, z' < 0, w' < 0\}$ or $\{x' < 0, y' < 0, z' > 0, w' > 0\}$
2. $x' = y'$

3. $z'^2 = w'(\sqrt{2}x' + w')$

4. $\|w'\| > \sqrt{2}\|z'\|$

In our automated construction algorithm (Section 4.3), we need to determining whether a newly added patch collides with any existing patches during the fold transform. Since it is infeasible to exhaustively check the above conditions for every pair of points, we use the following *tile-based* test, which is efficient to run and is also conservative; the patches that pass the test are guaranteed to never come into intersection during the fold transform.

Given two patches g, h, we consider a square tiling of the supporting plane of each patch. We first identify the set of tiles that overlap each patch. For each of g's tile s, we next identify all tiles on the supporting plane of h that could come into intersection with s, and mark them as *collision* tiles. Then, if h's tiles avoid all collision tiles from g, the two patches would never intersect.

The key step in the test is computing collision tiles between some tile s and some plane l. Denote B as the set of points on l that come to intersection with some point on the *border* of tile s. The key observation is that, for an interior point p of tile s, any point q on l that collides with p during folding lies in the convex hull of B. To see this, note that at the moment when p coincides with q, the plane l would also intersect with s along a line segment that contains p and ends on the border of s. This observation leads to a two-step process of finding collision tiles. First, we identify all tiles on l that overlap with B. Note that the coefficients of the point q on l intersecting with any given point p, if exists, can be uniquely determined from the conditions above with the additional constraint that q shares two coefficients with those of l on the opposing axes of l. We apply a bisection approach along each straight edge of s to identify all tiles on l that contain points on B. Next, we add in all other tiles on l that fall into the convex hull of the tiles found in the first step, which become the collision tiles.

Besides determining intersection between two patches, another utility of collision tiles is to determine a "safe" region on a given plane where the patch can be located that would avoid intersection with existing patches. This is very useful in an interactive design environment (Section 4.2), as the design will be ensured to be intersection-free without trial-and-error.

4.2 Interactive design

Here we present an interactive tool where the user can build up a Cartesian v-scaffold step-by-step using the five mechanisms discussed in Section 3.3. At each step, the tool makes automated suggestions of possible locations for adding patches, and guides the user in designing the patch shape so that the scaffold is intersection-free and enclosing by construction. We will explain the design process using the screen shot of the tool in Figure 7. The tool consists of several sub-windows that will be explained below.

The design process starts with a ground and a backdrop patch. In each step of the design, the user picks any two existing patches g_1, g_2 (gray in Window A), and the tool automatically identifies possible mechanisms that can be applied to these patches (based on their labels and coefficients which are internally maintained). Once a mechanism is selected by the user, the tool computes and visualizes the supporting planes of new patches that can be connected to g_1, g_2 (red in Window A) and the hinge lines (black in Window A). For discrete computation, the tool restricts patches to lie on grid planes in a Cartesian grid with a user-specified grid spacing. Note that, in some mechanisms, the supporting planes of one (as in S1 and D3) or both (as in D2) of the connectable patches are not

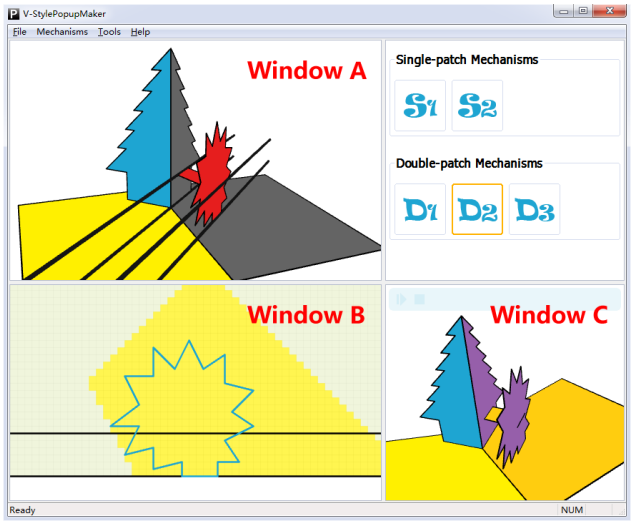


Figure 7: *A screen shot of the interactive design tool.*

unique. In these cases, the user can slide among the possible planes using the mouse cursor.

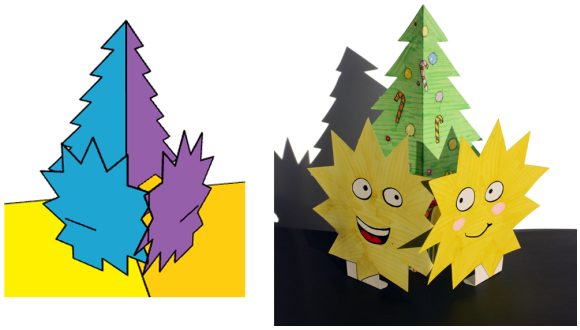
Once the supporting planes of the new patches are selected, the tool determines their labels and coefficients, and applies the tile-based procedure above to identify collision tiles (i.e., grid squares) on these planes with all existing patches. In addition, tiles that fail the enclosure condition in Section 3.4 are also detected. These two kinds of tiles are collectively called *dirty* tiles. This is a very efficient process that takes typically less than a second. The user can then draw the 2D shape of the patch in a separate window (Window B) within a region where the dirty tiles are excluded (yellow in Window B). When done, the new patches are then added to the scaffold, and their hinges with g_1, g_2 are automatically created.

The tool is simple to use and *error-proof*; any user input would result in a pop-uppable design. The tool can also simulate the folding transform (by simply following Equation 2) to visualize how the scaffold is closed and opened (Window C). We demonstrate a number of example pop-ups we designed using the tool in Figure 8. While seemingly simple, these pop-ups reveal several features that would be difficult to achieve without the tool. For example, the right yellow star in (a) is close to but not intersecting with the ground patch during folding, thanks to the detection of the dirty tiles. Also, the pair of yellow and blue patches on the top-left in (c) are connected with the neighboring patches using the new D3 mechanism, which requires a delicate patch configuration.

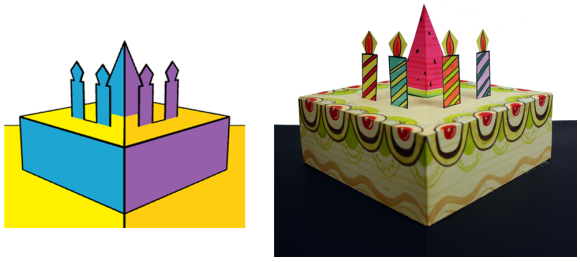
4.3 Automated construction

We next present an algorithm that invokes the same mechanisms but in an automated way, aiming at representing a given voxelized model. Voxelized model can be easily converted from other geometric representations such as polygonal meshes [Ju 2004]. Such automated algorithms have been proposed previously for constructing parallel paper architecture [Mitani et al. 2003; Li et al. 2010], a special kind of v-scaffold with patches parallel to only ground and backdrop. While paper architectures can only represent voxel faces with two orientations, our algorithm generates more general Cartesian v-scaffolds that capture voxel faces in five out of the six possible orientations.

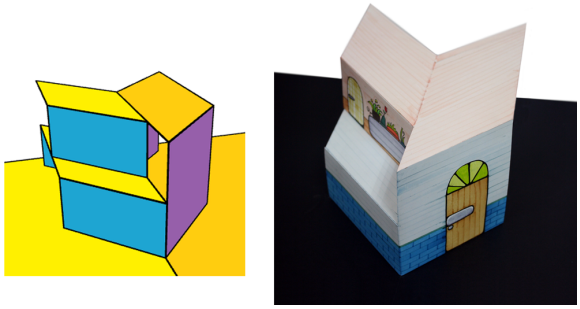
The input of the algorithm is a collection of voxels V (unit cubes) that form one 6-connected component. To avoid intersection in the result, we further require that the corner of the bounding box of V



(a) A Christmas tree and two spirits



(b) A birthday cake with candles



(c) A two-storey house

Figure 8: *Some pop-ups designed using the interactive tool, and the actual pop-up made with textured paper.*

with the minimum coordinates lies at a corner of some voxel in V. This corner is called the *pivot*. The algorithm outputs a pop-uppable v-scaffold whose patches cover all exterior faces of V except those oriented towards the negative Z axis, and no parts of any patch extrude outside the solid represented by V.

The algorithm proceeds in three steps. Patches are first constructed to cover the exterior faces of V parallel to Z axis (called the *vertical* faces), primarily using mechanisms S1 and D2. Then patches covering exterior faces oriented towards the positive Z axis (called the *horizontal* faces) are added using mechanism D1. These two steps also guarantee that the scaffold is intersection-free. Finally, the ground and the backdrop of the scaffold are determined, so that they enclose all patches when folded. Each step is explained below.

Vertical faces We first identify a set of vertical faces F (whether interior or exterior of V) with the following two properties. First, each connected set of voxels in the same layer in V are grouped into rectangular *blocks* bounded by faces in F, such that every two blocks can be connected via a path of other blocks where, for any two consecutive blocks in the path, the rectangle base of one block (e.g., Block 1 in Figure 9 (a)) shares a corner with the base of the other block (e.g., Block 2 in Figure 9 (a)). Second, given two connected sets of blocks in adjacent layers whose bases overlap, there is a pair of blocks (e.g., Block 1,2 in Figure 9 (b)), one in each layer, whose bases share a corner and have non-zero overlap area. We obtain F by first adding faces that divide each layer of voxels into rectangular blocks, then iteratively refining the blocks until these two properties are met.

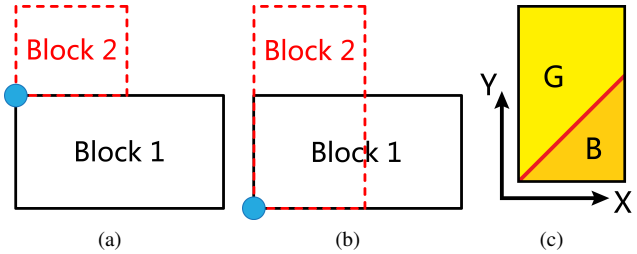


Figure 9: *Illustrations used in the automated algorithm.*

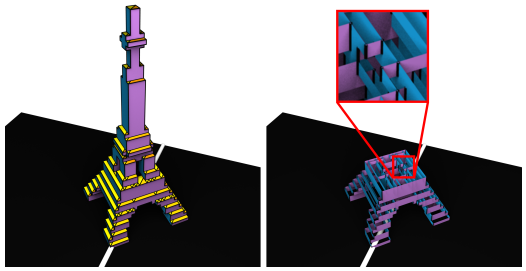
Once the faces are identified, we group all connected vertical faces with a common orientation into a single patch. A patch parallel to X, Z (or Y, Z) axes is labelled R (or L), whose coefficient on the W axis is zero and the coefficient on the Y (or X) axis is the same as the Y (or X) coordinate of the grid plane of the patch. The two properties of F above ensure that all the blocks can be ordered, starting with the block that contains the pivot (noted as b), such that each block shares at least two bounding patches with a block earlier in the sequence.

Assuming the pivot lies on the central hinge, we first add the two patches h_1, h_2 containing the pivot to the scaffold and connect them with the ground and backdrop (using mechanism D1). We then add the two remaining patches bounding b and connect them with h_1, h_2 (using mechanism D2). Next, we go through the sequence of blocks in order, applying either mechanism S1 or D2 to add the remaining patches bounding each block to the scaffold (there are at most two remaining patches for a block). In this way, all patches will be included in the scaffold, and are connected with hinges along their intersections.

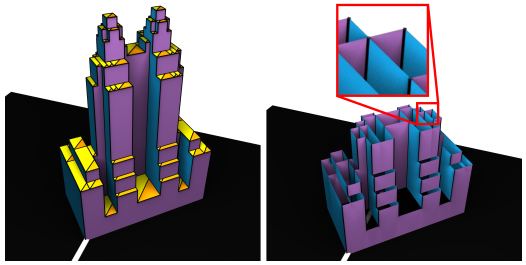
Since all patches built this way have common coefficients on the W axis, they will not intersect during folding: as stated in the first condition in Section 4.1, folding in a Cartesian v-scaffold requires a non-zero difference in each coefficient. Using the inequalities in the same condition, it can also be verified that no intersection will occur between any of the patch and the ground (or backdrop) once the bounding box requirement mentioned earlier is met.

Horizontal faces To capture the horizontal faces, we consider each connected piece of horizontal faces. If a piece is not a rectangular shape, we divide it into rectangles along the grid lines, and call these rectangles *cells*. If the vertical faces underneath the edge of a cell are not part of the existing scaffold, we subdivide the block underneath the edge along the supporting line of the edge, creating a patch of new vertical faces. The patch is added to the scaffold and connected to the bounding patches of the original block (using mechanism S1). With the completion of this block-division process, each cell of horizontal faces lies directly on top of a block.

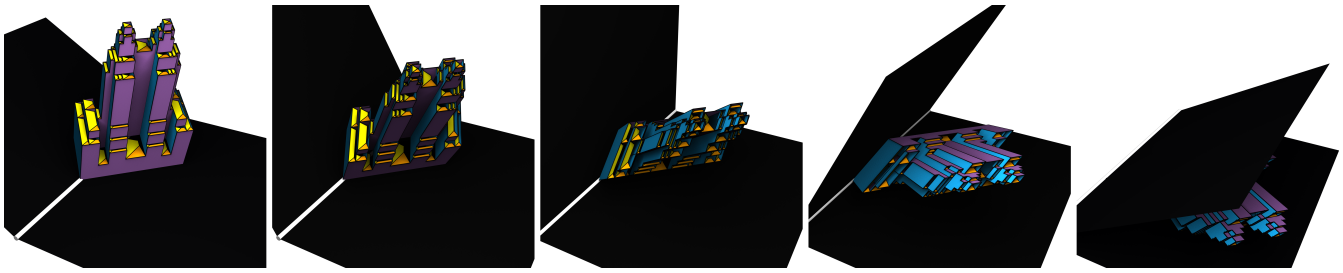
For each cell, we create two patches with labels G, B that are connected by a diagonal hinge, as shown in Figure 9 (c). The coefficients of each patch are the same as the Cartesian coordinates of the end of the hinge with a lower X coordinate (highlighted in (c)). The two patches are then connected to the two vertical bounding patches of the block beneath the cell (using mechanism D1). Note that intersection may occur between these two new patches and existing ones, which we detect using the tile-based procedure above.



(a) The Eiffel Tower



(b) The Waldorf-Astoria Hotel



(c) Intermediate states of the Waldorf-Astoria Hotel at $t = 1/6, 2/6, 3/6, 4/6$ and $5/6$ (from left to right)

Figure 10: *Two pop-ups constructed by the automated algorithm and their interior vertical supports (a,b). Intermediate states of the Waldorf-Astoria Hotel are shown in (c). The ground and backdrop are colored black, with the central hinge colored white.*

If there is intersection with another patch h, we subdivide the cell into two smaller cells (adding new vertical patches beneath the division line as done above), create two patches for each new cell and perform intersection check again. The division line is chosen as the intersection of the supporting plane of h with the cell, if it exists, otherwise as the mid-line of the cell along its longer edge. The subdivision process is guaranteed to terminate, since there is no intersection at the level of a single-face cell.

Ground and backdrop After all patches are added into the scaffold, we determine the bounding box (oriented along the W axis) that contains all patches in the closed state based on the fold transform in Equation 2. The bounding box is used as the bound of the ground patch, and is reflected over the central hinge to define the backdrop.

Two examples of v-style pop-ups constructed automatically are shown in Figure 10. To better visualize the structure of these pop-ups, we also show only the patches parallel to the Z axis, as well as several intermediate states of folding.

5 Conclusion and discussions

In this paper, we study the geometric structure of v-style pop-ups and propose sufficient, verifiable conditions under which a v-style paper (scaffold) is pop-uppable. To our knowledge, this is the first geometric study of a general class of pop-up structures that is not based on known pop-up mechanisms. Our study confirmed existing mechanisms and discovered several novel mechanisms that have not been reported before. Guided by the analysis and the mechanisms, we present an interactive design tool and an automated construction algorithm, both enforcing pop-uppability during the design and construction.

We believe this work opens up a novel and broad venue of future research on pop-ups. On the theoretical end, it is possible to improve the stability conditions in Proposition 2 to include more complex ways in which a group of more than two patches can be stabilized by hinging on other patches. The expanded conditions will lead to new mechanisms that add more than two patches at a time. A bigger challenge is to explore how the theories in v-style pop-ups can be extended onto more complex pop-ups, e.g., with five or more parallel groups of patches. Considering the physical properties of the paper (e.g., thickness, weight, and warping) is another direction of extension that has significant practical value.

On the algorithmic side, the current automatic construction algorithm can be easily extended for creating non-Cartesian v-style pop-ups, by adopting the general intersection criteria in Proposition 3. On the other hand, asking users to choose among a library of mechanisms may not be very convenient from the interaction point of view, particularly for novices. How to provide more intuitive pop-up design tools is another interesting direction. For example, it would be desirable to let the users create an arbitrary patch and have the tool automatically find the appropriate mechanisms and additional patches to include the new patch in the design.

Acknowledgements

We thank Jie Yin and Xiang Guo for useful discussions, and all the reviewers for their helpful comments. This work was supported by the National Basic Research Project of China (2011CB302203), the Natural Science Foundation of China (U0735001), and the National Science Foundation of USA (IIS-0846072).

References

BELCASTRO, S., AND HULL, T. 2002. Modelling the folding of paper into three dimensions using affine transformations. *Linear Algebra and its Applications 348*, 273–282.

BIRMINGHAM, D. 1997. *Pop Up! A Manual of Paper Mechanisms*. Tarquin Publications, UK.

CARTER, D. 1999. *The Elements of Pop-up*. Little Simon, New York.

DEMAINE, E., AND O'ROURKE, J. 2007. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*. Cambridge University Press, Cambridge.

GLASSNER, A. 2002. Interactive pop-up card design, part 1. *IEEE Comput. Graph. Appl. 22*, 1, 79–86.

GLASSNER, A. 2002. Interactive pop-up card design, part 2. *IEEE Comput. Graph. Appl. 22*, 2, 74–85.

HARA, T., AND SUGIHARA, K. 2009. Computer aided design of pop-up books with two-dimentional v-fold structures. In *7th Japan Conference on Computational Geometry and Graphs*.

HENDRIX, S. L., AND EISENBERG, M. A. 2006. Computer-assisted pop-up design for children: computationally enriched paper engineering. *Adv. Technol. Learn. 3*, 2, 119–127.

HINER, M. 1985. *Paper Engineering*. Tarquin Publications, UK.

HUFFMAN, D. A. 1976. Curvature and creases: A primer on paper. *IEEE Trans. Computers C-25*, 1010–1019.

HULL, T. 2006. *Project Origami: Activities for Exploring Mathematics*. A.K. Peters.

JU, T. 2004. Robust repair of polygonal models. *ACM Trans. Graphics 23*, 3, 888–895.

LEE, Y. T., TOR, S. B., AND SOO, E. L. 1996. Mathematical modelling and simulation of pop-up books. *Computers & Graphics 20*, 1, 21–31.

LI, X.-Y., SHEN, C.-H., HUANG, S.-S., JU, T., AND HU, S.-M. 2010. Popup: automatic paper architectures from 3d models. *ACM Trans. Graphics 29*, 4, 111:1–9.

MITANI, J., AND SUZUKI, H. 2003. Computer aided design for 180-degree flat fold origamic architecture with lattice-type cross sections. *Journal of graphic science of Japan 37*, 3, 3–8.

MITANI, J., AND SUZUKI, H. 2004. Computer aided design for origamic architecture models with polygonal representation. In *CGI '04: Proceedings of the Computer Graphics International*, IEEE Computer Society, Washington, DC. USA, 93–99.

MITANI, J., SUZUKI, H., AND UNO, H. 2003. Computer aided design for origamic architecture models with voxel data structure. *Transactions of Information Processing Society of Japan 44*, 5, 1372–1379.

OKAMURA, S., AND IGARASHI, T. 2009. An interface for assisting the design and production of pop-up card. *Lecture Notes in Computer Science 5531*, 2, 68–78.

O'ROURKE, J. 2011. *How to Fold It: The Mathematics of Linkages, Origami, and Polyhedra*. Cambridge University Press, UK.

UEHARA, R., AND TERAMOTO, S. 2006. The complexity of a pop-up book. In *18th Canadian Conference on Computational Geometry*.

A Proof of Corollary 1

Proof: We first show necessity. If S meets the Decomposition Condition, points on the hinge of g, h will have common coefficients on the axes opposing either patch. Hence g, h cannot have opposite labels, or otherwise all hinge points will have identical coefficients on all axes and thus degenerate to a single point. Assuming g, h have adjacent labels, points on the hinge will have common coefficients on three axes, hence parallel to the remaining axis that is supporting both g, h. Designate the two common coefficients of points on each patch as the coefficient of each patch. The Hinge Condition can be easily derived from the continuity of point coefficients in the Decomposition Condition.

To show sufficiency, we first determine the coefficients of each point independently from the coefficients of each patch containing the point (which is unique, as noted above). All we then need to show is that a point p on a hinge between some patches g, h receives the same coefficients from both patches. By conditions (2,3), p can be represented as $\phi_g\Phi + \psi_h\Psi + \omega_h\Omega + u\Upsilon$ for some u where Υ denotes the axis supporting both g, h. Note that the set of coefficients$\{\phi_g, \psi_h, \omega_h, u\}$ (ordered by axes $\Phi, \Psi, \Omega, \Upsilon$) coincide with the coefficients of both g and h on their opposing axes. Hence it is the same set of coefficients determined independently from g and h. □

B Proof of Proposition 3

Proof: Let $F(t) = x'X(t) + y'Y(t) + z'Z(t) + w'W(t)$. We need to show that $F(t) = 0$ for some $t \in (0, 1)$ if and only if the four conditions in Proposition 3 hold simultaneously.

We will show necessity and sufficiency separately. The proof uses the following identities that hold for any $t \in (0, 1)$:

$$F(t) \cdot (x'X(t)) - x'y'X(t) \cdot Y(t) = x'^2 + x'z' \cos\beta - x'w' \cos\delta \quad (4)$$

$$F(t) \cdot (y'Y(t)) - x'y'X(t) \cdot Y(t) = y'^2 + y'z' \cos\alpha - y'w' \cos\gamma \quad (5)$$

$$F(t) \cdot (z'Z(t)) - z'w'Z(t) \cdot W(t) = z'^2 + x'z' \cos\beta + y'z' \cos\alpha \quad (6)$$

$$F(t) \cdot (w'W(t)) - z'w'Z(t) \cdot W(t) = w'^2 - x'w' \cos\delta - y'w' \cos\gamma \quad (7)$$

Necessity: Suppose there is some $t_0 \in (0, 1)$ such that $F(t_0) = 0$. First, based on the angle condition of a v-scaffold, the four vectors $X(t_0), Y(t_0), Z(t_0), W(t_0)$ are linearly independent, and all point inside the space bounded by the rotated ground and backdrop planes. Hence x', y', z', w' must be all non-zero, and cannot have the same sign. By the same angle restrictions, vectors $X(t_0), Y(t_0)$ point to different sides of the plane formed by vectors $Z(t_0), W(t_0)$, and $Z(t_0), W(t_0)$ point to different sides of the plane formed by $X(t_0), Y(t_0)$. As a result, x', y' need to have the same sign, and so do z', w'. Hence the first condition holds.

To obtain the second and third conditions, simply subtract Equations 5 from 4 and Equations 7 from 6, and consider that $F(t_0) = 0$. To obtain the last condition, note that the angle between $X(t), Y(t)$ achieves its maximum at $t = 0$, where the angle is $2\pi - \gamma - \delta$, and its minimum at $t = 1$, where the angle is $\|\gamma - \delta\|$. The condition can be derived accordingly by noting that $\Theta = -x'y'X(t_0)Y(t_0)$ in Equation 4.

Sufficiency: We show that, if the four conditions are met, there is some $t_0 \in (0, 1)$ such that $F(t_0) = 0$. Combining the second condition and Equations 4 and 5 yields:

$$F(t) \cdot (x'X(t)) = F(t) \cdot (y'Y(t)) = x'y'X(t) \cdot Y(t) + \Theta$$

Note that the fourth condition implies that there exists some $t_0 \in (0, 1)$ such that $\Theta = -x'y'X(t_0) \cdot Y(t_0)$. Substituting $t = t_0$ into the equation above yields:

$$F(t_0) \cdot (x'X(t_0)) = F(t_0) \cdot (y'Y(t_0)) = 0 \quad (8)$$

Combining the third condition and Equations 6 and 7 yields:

$$F(t_0) \cdot (z'Z(t_0) - w'W(t_0)) = 0 \quad (9)$$

On the other hand, the first condition (particularly that z', w' have the same sign) implies that vector $z'Z(t_0) - w'W(t_0)$ points away from the plane defined by the vectors $x'X(t_0), y'Y(t_0)$, and hence the three vectors $x'X(t_0), y'Y(t_0), z'Z(t_0) - w'W(t_0)$ are linearly independent. Equations 8 and 9 suggest that $F(t_0)$ is orthogonal to these three vectors, which implies that $F(t_0) = 0$. □