



Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Issue on CAD/Graphics 2015

Fresh Press Modeler: A generative system for physically based low fidelity prototyping

Lujie Chen ^{a,*}, Lawrence Sass ^b^a Singapore University of Technology and Design, 8 Somapah Road, Singapore^b Massachusetts Institute of Technology, MA, USA

ARTICLE INFO

Article history:

Received 30 March 2015

Received in revised form

15 June 2015

Accepted 3 July 2015

Keywords:

Press fit

Planar structures

Digital fabrication

Laser cutting

ABSTRACT

For designers, digital manufacturing machines, such as laser cutters and CNC machines, support rapid prototyping of low-cost, low fidelity physical models. These machines can be used as an alternative to additive manufacturing. Unfortunately there are few CAD tools that provide access for fabrication of complex 3D geometries with these 2D fabrication machines. The literature contains a few novel systems that generate planar structures as models built of layered material or as interlocking planes with unique joining features. In this paper, a *Fresh Press* modeler is presented as a novel system that generates tailored geometry for ease of assembly. A major benefit of *Fresh Press* is the ability to produce fabrication data leading to a watertight planar structure. Assembly between planes is sustained by interlocking finger joints generated on each planar component of the model. The *Fresh Press* modeler parameterizes planar surfaces and interlocking features for user control and model quality. We end by demonstrating the system with examples of solid models and negative models used for mold making.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

The demand for physical prototypes by designers is growing. It is common knowledge that designers of all scales need artifacts for reflection and decision making as part of an iterative feedback loop in design [1]. Mostly there is a growing need for low-cost, low fidelity physically models [2].

Unfortunately, the limited volume and size of models available from common 3D printers do not satisfy the needs of designers of very large products such as boats, planes and buildings. Instead, many designers construct models as planar surfaces using conventional hand-held tools. Architects, for example, craft model parts using knives and saws, later assemble them by hand with conventional tools and adhesives.

Alternatively, today it is possible to manufacture models directly from CAD data from 3D digital data using laser cutters and CNC machines. In this case, fabrication data are prepared using keyboard driven modeling and drafting techniques. Then, designers use laser cutters to fabricate model parts which are assembled to a physical model. Due to the manual modeling involved in the process, this approach is often limited to making models that are not larger than a meter square.

There are few commercial systems that support the production of artifacts greater than one cubic meter at a reasonable price. A few companies have begun to address this problem by manufacturing

very large scale 3D printing machines that produce high fidelity, yet costly models [3]. A notion in the rapid prototyping community has been that design success is related to the quality of prototyping; however, studies in a well cited paper suggested that simple, low fidelity models manufactured quickly lead to good, well informed design outcomes [2].

The research community has begun addressing the need for methods to generate low fidelity prototypes as alternatives to the widely accepted additive methods found with 3D printing. Novel methods are graphics based algorithms used to generate models as a collection of objects from a starting model. Current approaches are used to manufacture furniture, toys and models as interlocking, interlacing and sewn physical objects [4–7]. However, these approaches and tools do not address the need for a system that generates components that assemble easily as a watertight model.

A simple method of low fidelity modeling using CAD software and a laser cutter is demonstrated as a desktop-sized staircase model in Fig. 1. This model was generated first in CAD as a solid mesh model. Interlocking 2D components were drafted for cutting in a matter of days by keyboard and mouse entry using the initial mesh model as a guide. After, the components were laser cut from drafted parts in less than two hours, painted and assembled the next day in less than an hour. A low fidelity model was produced in approximately three days.

A production system that can generate physical artifacts quickly impact architects, civil and product engineers. Finished volumes of a watertight structure can be used by the design community as low fidelity models for design review of many sizes. We believe that the woodworking industry and concrete casting industries could benefit

* Corresponding author.

from rapid construction methods to deliver solid shapes and forms for casting [8]. A successful system of model production will address learning aspects associated with initial modeling and physical output.

The *Fresh Press* modeler is a production system that creates a 3D watertight model of interlocking planar structures from a solid mesh model. The system takes a triangle mesh model as an input, extracts planar surfaces of the model, generates finger joints [9] at the intersection of the surfaces to enable interlocking connection. The outcome is digital data of the planar surfaces with joints ready for fabrication. Basic user controls define material thickness, the number of interlocking fingers along an edge and the tolerance of cutting. The physical structure can then be assembled manually to produce a representation of the original model. A snapshot of the process is shown in Fig. 2.

2. Related work

Interlocking planar structures derived from 3D mesh models were introduced broadly by architects and engineers as a means to design and produce complex structures efficiently [10]. A system approach used to generate wooden structures was demonstrated and organized as a grammar, the main result of which was a plywood cabin built of interlocking parts [11,12]. Rules within the system were designed to guide modeling when decomposing an initial shape model into interlocking components. The system was not automated or programmed, instead components were

modeled by keyboard and mouse entry aimed at describing opportunities that could come from the system.

2.1. Automated systems

Initial automated systems that generated planar structures with interlocking features were developed soon after. Oh et al. developed a semi-automated modeling system, in which connections between planes were identified by the modeler [4]. Automation supported a need to offload laborious modeling tasks to the computer. The program worked best to generate small toys and furniture. Generating slots and joints from complex starting models revealed limitations in the system that were addressed later by Lau et al. [13]. Last, an automated planar structure modeler was developed by Schulz et al. [14], whose approach was creating an expert system with a large collection of parametric components, such as boards, wheels and connectors (e.g. screws and hinges).

2.2. Cross planar structures

Cross planar models can be defined as a class of models built of intersecting layers of material across two directions allowing rapid generation of abstract shapes from mesh models. McCrae et al. [15] developed an interactive system requiring a user to pick a subset of planes of a model based on relative importance to represent the model. Saul et al. [16] created a chair-design system, in which the orientation of the slots could vary according to the geometry of a



Fig. 1. A desktop model measuring (53 × 38 × 48 cm³) laser cut of masonite. Finger joints were drafted through keyboard entry within CAD software.

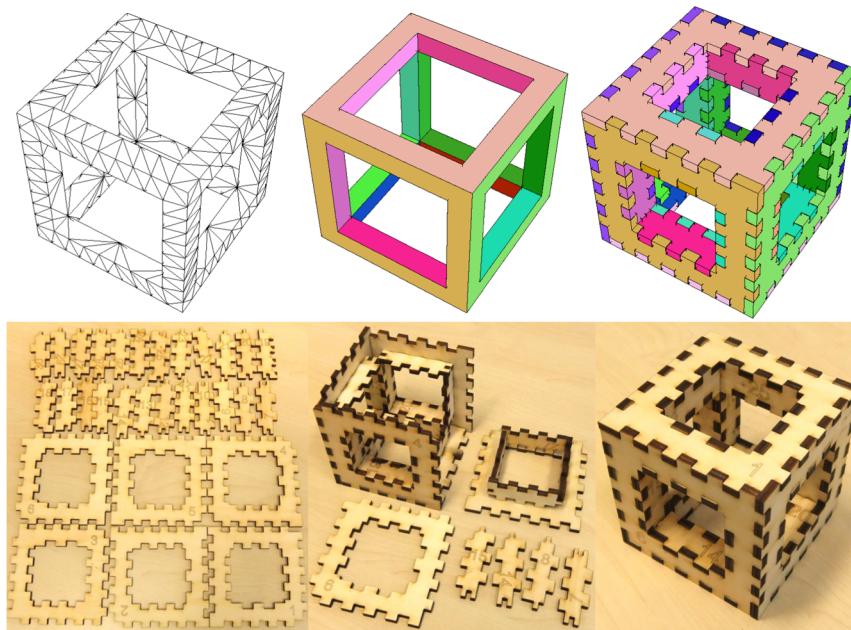


Fig. 2. Planar surfaces complete with finger joints at edges are generated by Fresh Press Modeler from a triangle mesh model.

design as a way to improve chair strength and comfort. Hildebrand et al. [17] proposed an algorithm that contrasts with prior work by demonstrating a method to vary the angle and place planes in locations based on visual goals while maintaining structural goals. Le-Nguyen et al. [18] allowed for unequally spaced slots leading to rationalized models with strong visual representation. Schwartzburg and Pauly [19] described an even more flexible algorithm that generated slots to connect non-perpendicular planar components. Cignoni et al. [20] showed that by loosening the rigidity constraint of a planar component, complicated structures could be assembled from ribbon-like, bendable components. Their models demonstrated a system of interlocking planar contours laser cut from paper as a way to build models of high-quality visual representation. Finally, a commercial example of this method was developed by Autodesk. Their system is named as 123D Make that allows the user to generate a planar structure from a 3D mesh model in several ways. Each of these automated modeling systems produce planar components with interlocking features. The assembled model is an open structure that represents the original digital model.

2.3. Watertight structures

Watertight planar structures are physical models as a complete volume, as opposed to sets of overlapping contours. There is sparse exploration of this method of model production. Recently, Chen et al. [7] developed a multiplanar modeler that subdivided an original model into several planar surfaces. Interior connectors were used to join the planar components. If the interior angles between planes are too sharp (less than 90°), the authors suggested using finger joints for connection. We believe that there are limitations in model fabrication and assembly when using the interior connectors. First, the connectors must be fabricated by a separate process from the planar components. Second, as the connectors are inside the surface volume, assembly will be difficult as the access area to the connectors is continuously reduced while the surface of a model is gradually closed during assembly.

3. Computing press fit mechanism

We address the challenge of assembly design by exploring a generative system for assembly of planar structures. We created a system that generates finger joints where two planar surfaces meet.

Given a 3D triangle mesh as an input, our goal is to generate a constructible set of 2D components with interlocking features ready for fabrication. The end result is a set of planar components complete with finger joints. The system must also generate corresponding 3D components for visual evaluation prior to fabrication. This is done in three steps:

- Planar components are extracted from the mesh ([Section 3.1](#)).
- Connectivity of the planar components is analyzed ([Section 3.2](#)).
- Finger joints are generated for each mutual edge between two planar components ([Section 3.3](#)).

3.1. Planar component extraction

A planar component of a triangle mesh is a polygon that encompasses a group of coplanar adjacent triangles.¹ To extract

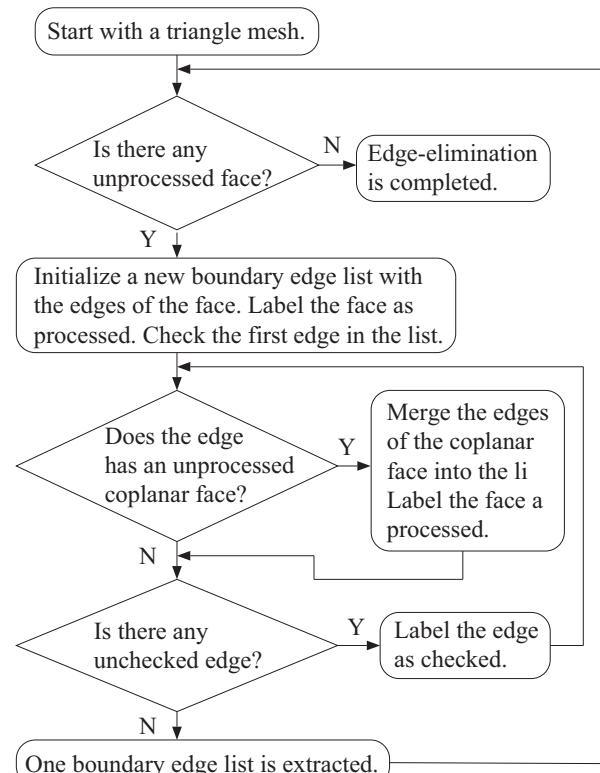
¹ Due to floating point calculation, an implementation-dependent threshold with respect to the dot product of the normal vector of two triangles is needed to determine coplanarity. A dot product of 1 suggests perfect coplanarity. We used a threshold of 0.999.

planar components, the mesh is re-characterized by merging such triangles, which is an inverse operation of polygon triangulation. Methods exist in creating polygons from a set of coplanar points [21]; however, at this stage our goal is redefinition of areas of the mesh to form arbitrary polygons, possibly with holes.

We assume that triangles within the mesh have consistent winding direction and an edge is not shared by more than two triangular faces, i.e. 2D manifold mesh. A vertex list is generated defined by end points of multiple edges; each edge has two end points and belongs to at most two faces; each face is a triangle with three vertices and three edges. Lists for faces and edges are created, each storing the non-duplicated entities, i.e. edges and faces. An edge entity contains information about its end points and the face(s) it belongs to. A face entity contains information about its vertices and edges.

After, edge-elimination is used to extract a list of boundary edges of a group of coplanar adjacent faces. A new boundary edge list is created and initialized with the edges of an arbitrary face. Each entity of the list is checked for coplanar faces. The edges of the coplanar face are merged into the list, i.e. duplicated edges are removed and unique edges are inserted. If there is no such unprocessed face, one list, corresponding to the edges of one polygonal face, has been extracted. [Fig. 3](#) shows a flowchart of the edge-elimination process for extracting the boundary edge lists of all polygonal faces from a triangle mesh.

[Fig. 4\(a\)](#) shows edge-elimination by extracting one boundary edge list. A number within a triangle represents the sequence of processing. Numbers along the edges represent the sequence at the edges of each triangle. As indicated by the arrows, the winding direction of the triangles is anti-clockwise. A boundary edge list is initialized with edges 1.1 → 1.2 → 1.3. Edge 1.1 does not have a second coplanar face, while edge 1.2 has; hence the edges of face 2 are merged into the list. Duplicated edges 1.2 and 2.1 are removed and unique edges 2.2 and 2.3 are inserted. At this step, the list contains edges 1.1 → 2.2 → 2.3 → 1.3. The next edge that has



[Fig. 3](#). Flowchart of an edge-elimination process.

a second coplanar face is 2.3; hence the edges of face 3 are merged into the list. So on and so forth, until there is no unprocessed, coplanar face. The resultant boundary edge list is 1.1 → 2.2 → 5.1 → 6.3.

If a polygon has a hole, as the one shown in Fig. 4(b), the resultant list would be 1.1 → 2.2 → 5.1 → 6.3, 8.1 → 7.3 → 4.3 → 3.2. Edges 6.3 and 8.1 are not connected and edges 8.1 → 7.3 → 4.3 → 3.2 have a reversed winding direction to that of the triangles. The information can be exploited to extract a polygon from the boundary edge list.

Last, based on the connectivity of the edges in the list, one or more sub-lists are generated, each storing a closed loop of edges. For a polygon without holes, only one list is produced. For a polygon with holes, a separate sub-list stores the outer edges and each of the other sub-list stores the edges of a hole. The outer edges can be differentiated from the holes by the winding direction. Planar component extraction produces several arbitrary polygons that may have holes from a triangle mesh. Each polygon is a planar component of the input model.

3.2. Connectivity analysis

The connectivity of the planar components is analyzed also by reading edges and points again generating: a vertex list, an edge list and a face list; each stores non-duplicated entities. A face entity has at least three vertices and three edges because it is a polygon. One detail is special to the polygonal faces, as shown in Fig. 5. Two polygons may have an unequal-length mutual edge. In this case, to facilitate subsequent processing, a vertex is inserted to break the longer edge so that the length of the mutual edges is the same.

Next, the interior angle (θ) between any two adjacent faces is calculated and the information is recorded in their mutual edge entity. The angle is crucial to finger joint generation; it determines the amount of protrusion (d_p) and indent (d_i) of the finger joints with respect to the edge. The relationship between θ , d_p and d_i is

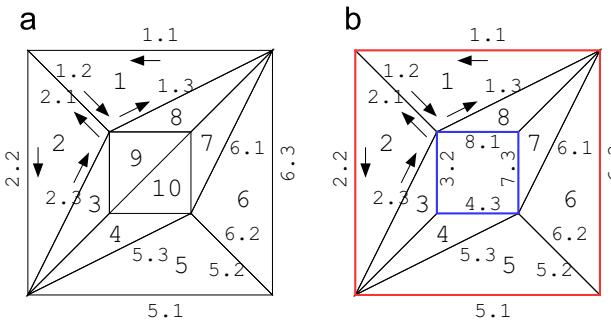


Fig. 4. (a) A polygonal face of ten triangles. (b) A polygonal face with a hole. The red square indicates the outer edges; the blue square indicates the edges of the hole. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

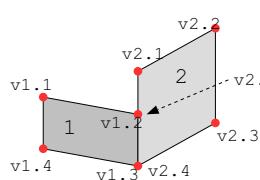


Fig. 5. Polygons 1 and 2 have an unequal-length mutual edge: edge v1.2→v1.3 and edge v2.4→v2.1. A vertex v2.5 is inserted to polygon 2 so that the mutual edges become v1.2→v1.3 and v2.4→v2.5.

depicted in Fig. 6 and is given by

$$d_p = \begin{cases} T/\sin \theta, & \theta \in (0, \pi/2] \\ T/(1 - \cos \theta), & \theta \in (\pi/2, \pi] \\ T \left[\frac{1}{1 - \cos \theta} + \frac{1}{\tan(\theta/2)} \right], & \theta \in (\pi, 3\pi/2] \\ d_i + T, & \theta \in (3\pi/2, 2\pi) \end{cases} \quad (1)$$

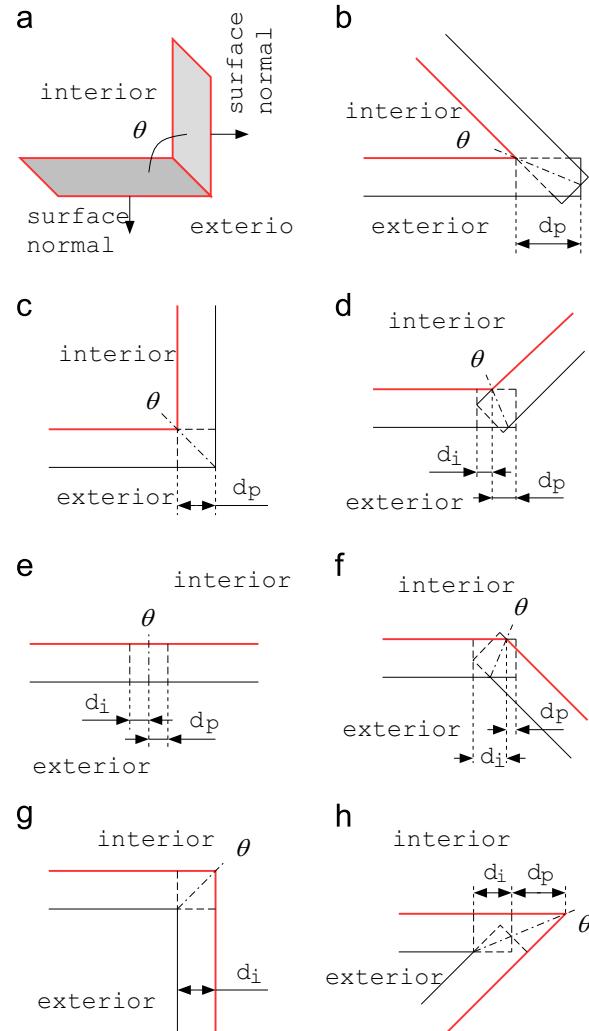


Fig. 6. The relationship between θ , d_p and d_i . (a) The surface normals of the two faces indicate the exterior of a 3D volume. θ is the interior angle between the two faces. (b) $\theta \in (0, \pi/2]$. (c) $\theta = \pi/2$. (d) $\theta \in (\pi/2, \pi]$. (e) $\theta = \pi$. (f) $\theta \in (\pi, 3\pi/2)$. (g) $\theta = 3\pi/2$. (h) $\theta \in (3\pi/2, 2\pi)$. The red line indicates the original boundary of the 3D volume. d_p and d_i are shown for the left, horizontal edge only; those of the other edges are exactly the same based on symmetry about the dot-dashed line. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

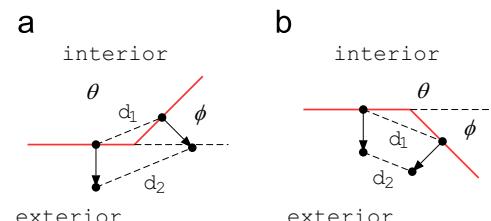


Fig. 7. Determination of θ . (a) $d_1 \leq d_2$. (b) $d_1 > d_2$.

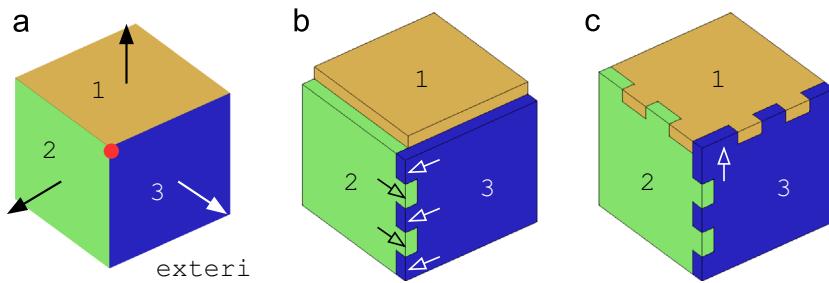


Fig. 8. (a) Three faces have a common vertex indicated by a red dot. The arrows indicate the surface normal and the exterior of a 3D volume. (b) Finger joints are generated between faces 2 and 3. The hollow arrows indicate the direction of protrusion. (c) Finger joints of the other mutual edges are generated. Face 3 occupies the vertex. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

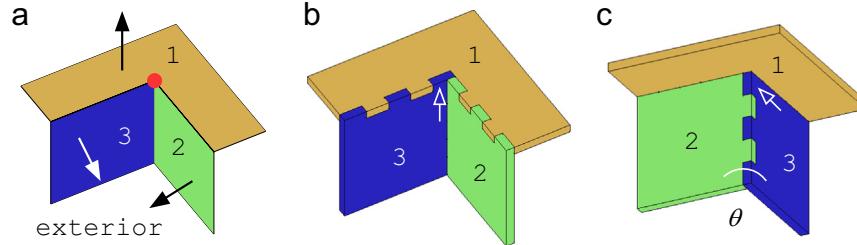


Fig. 9. (a) The interior angle between faces 2 and 3 is $3\pi/2$. Finger joints viewed from (b) exterior and (c) interior of the volume. In (b), the finger joints between faces 2 and 3 are not visible. In (c), the finger joints between faces 1 and 2, 1 and 3 are not visible.

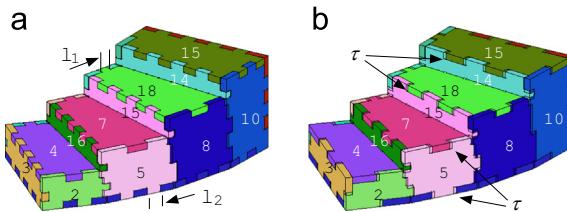


Fig. 10. (a) Length of a finger joint is an indicative parameter. l_1 and l_2 are the actual length of finger joints on two different edges. (b) Tolerance of the finger joints.

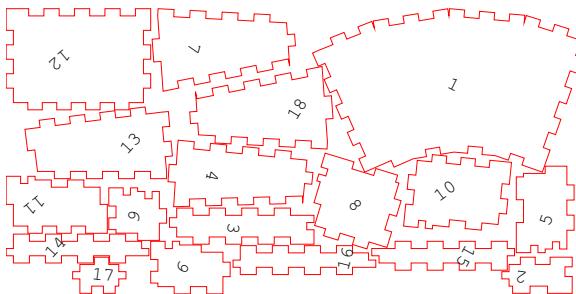


Fig. 11. Output of 2D polygons of the structure shown in Fig. 10.

$$d_i = \begin{cases} 0, & \theta \in (0, \pi/2] \\ d_p \cos \theta, & \theta \in (\pi/2, \pi] \\ \frac{T(1 - \cos \theta)}{\tan(\theta/2)} + d_p \cos \theta, & \theta \in (\pi, 3\pi/2] \\ -T/\tan(\pi - \theta/2), & \theta \in (3\pi/2, 2\pi) \end{cases} \quad (2)$$

where T is the thickness of the planar material. It can be verified that for $\theta \in [\pi/2, 2\pi]$, $d_p - d_i = T$. This is our primary empirical principle for choosing d_p and d_i . For $\theta \in (0, \pi/2)$, $d_p - d_i > T$; the extra length is for increasing the overlap of the joints, while setting a length threshold to guard against a small θ is required in practice.

θ should be determined in a 2π range but by default, the angle between two planes is in between 0 and π , which can be calculated

by

$$\phi = \begin{cases} \arctan \frac{|v_1 \times v_2|}{v_1 \cdot v_2}, & v_1 \cdot v_2 > 0 \\ \pi/2, & v_1 \cdot v_2 = 0 \\ \pi + \arctan \frac{|v_1 \times v_2|}{v_1 \cdot v_2}, & v_1 \cdot v_2 < 0 \end{cases} \quad (3)$$

where v_1 and v_2 are the surface normal vectors of the planes. As $|v_1 \times v_2|$ is non-negative and $v_1 \cdot v_2$ may be positive, negative or zero, the angle ϕ is determined in $[0, \pi]$.

To convert ϕ to θ , additional information is required. Fig. 7 illustrates our strategy. Two arbitrary points on the adjacent faces are selected. d_1 denotes the distance between the points. They are shifted along the respective surface normal direction by an equal amount. d_2 denotes the distance between the shifted points. θ is given by

$$\theta = \begin{cases} \pi - \phi, & d_1 \leq d_2 \\ \pi + \phi, & d_1 > d_2 \end{cases} \quad (4)$$

Thus, θ is determined in between 0 and 2π .

3.3. Finger joint generation

Finger joints are generated sequentially on every mutual edge between two faces. The order of processing, i.e. which edge is processed first, does not matter. What matters is the occupancy of each vertex, i.e. which face occupies a vertex. In an example shown in Fig. 8, there are three mutual edges between faces 1 and 2, 2 and 3, and 1 and 3. A vertex indicated by a red dot in Fig. 8(a) is common to the three faces. The edge between faces 2 and 3 is processed first [Fig. 8(b)]. The amount of protrusion and indent is determined by the angle between the faces. The direction of protrusion, indicated by hollow arrows, is determined by the cross product of the surface normal vector and the edge vector. Because of the finger joints, where one face protrudes, the other face shall indent, vice versa. Face 3 occupies the vertex as it protrudes at the side of the vertex. In subsequent processing, face 3 should

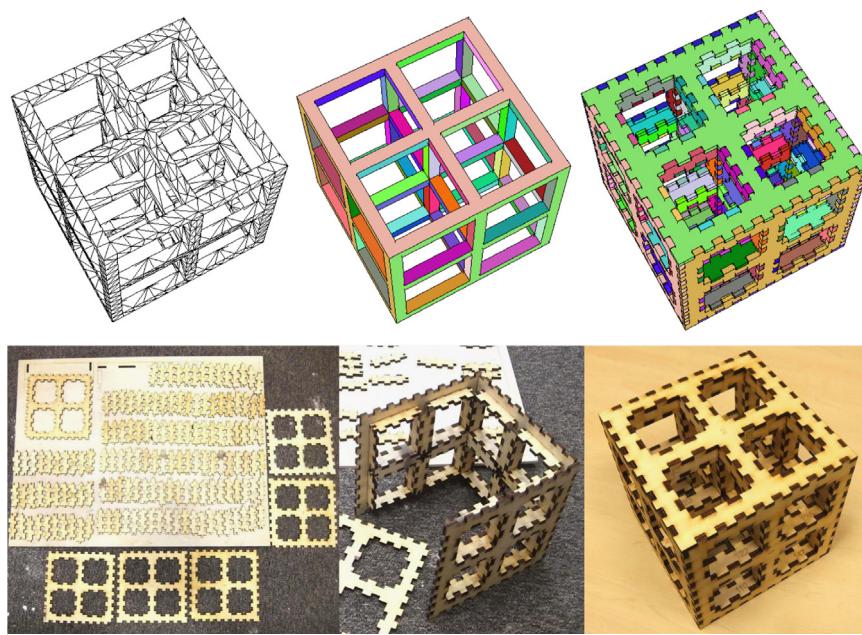


Fig. 12. A cube consisting of 150 components.

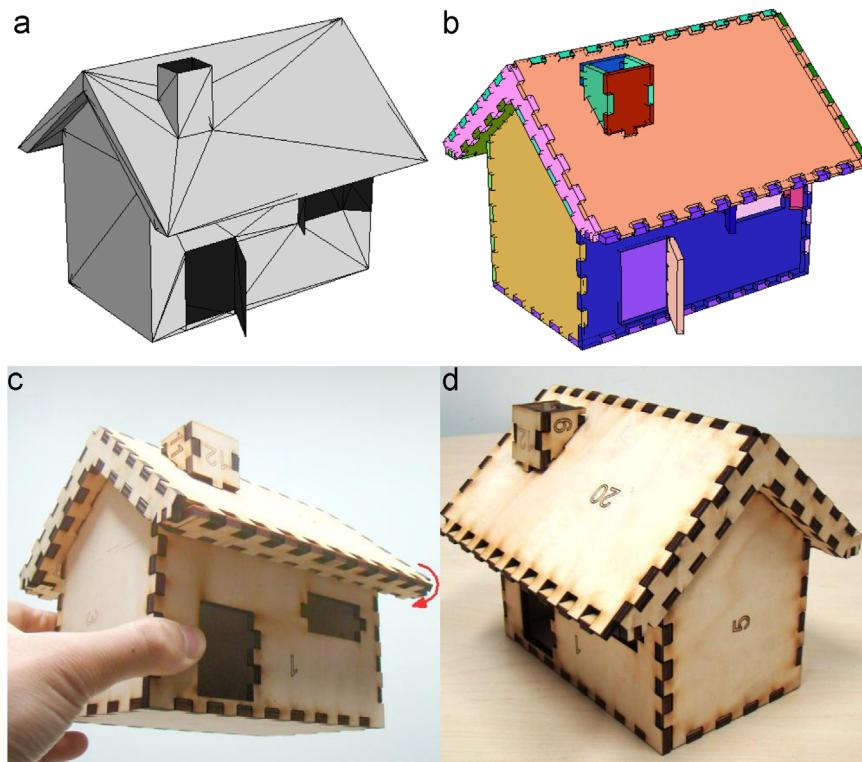


Fig. 13. A house model of 250 mm long, 230 mm wide and 210 mm tall. The red arrow in (c) indicates an edge of high curvature. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

maintain the occupancy of the vertex, as shown by the generated finger joints between faces 1 and 3 in Fig. 8(c).

Vertex occupancy requires each face that has occupied a vertex be recorded. In general, a vertex is visited several times during processing. At the every first visit, the protruding face along the side of the vertex is recorded as the occupying face. At the second visit by that face from an adjacent edge to the previous one, the face should protrude at the side of the vertex again.

This strategy works on situations where the protrusion value is zero or negative. Fig. 9 shows an example, in which face 3 occupies the vertex common to three faces. The angle between faces 2 and 3 is $3\pi/2$, resulting in $d_p=0$ based on Eq. (1). If the edge between faces 2 and 3 is the first visit to the vertex [see Fig. 9(c)], face 3 is recorded as the occupying face even though $d_p=0$. At the second visit by face 3 from the edge between faces 1 and 3, face 3 protrudes again at the side of the vertex [see the arrow in Fig. 9(b)].

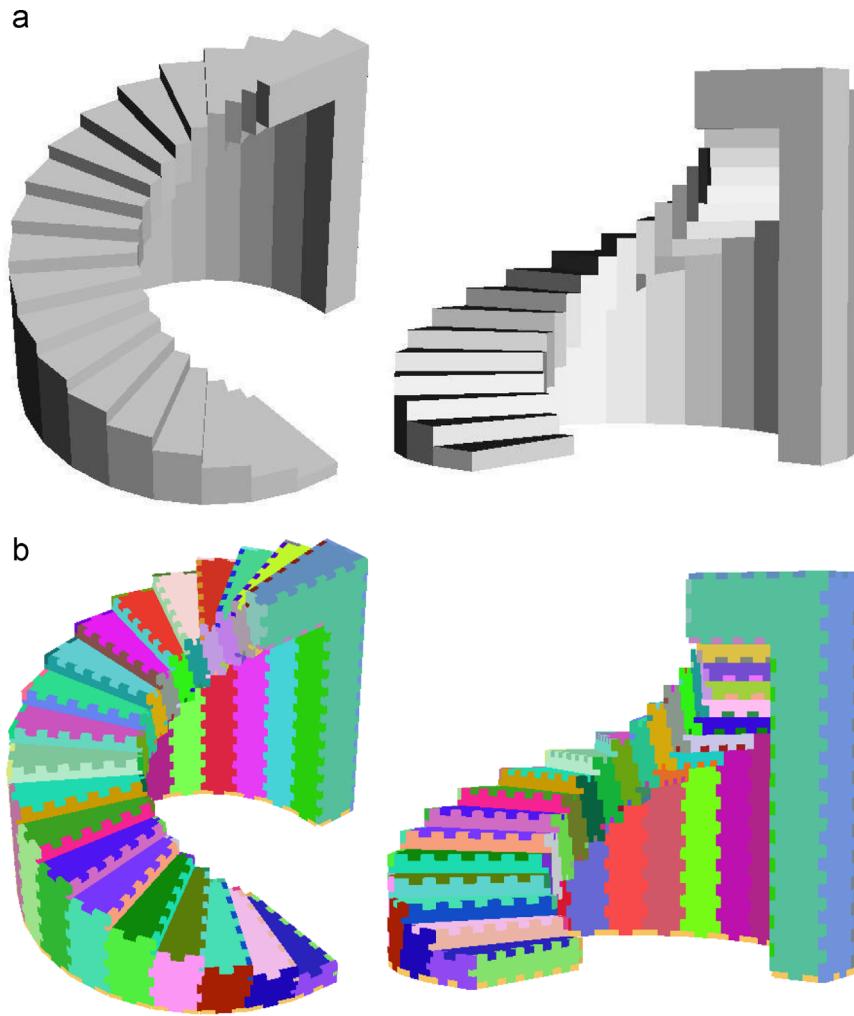


Fig. 14. (a) A spiral staircase model created in Rhinoceros 3D modeling software. (b) Finger-joint structure generated by the proposed algorithm.

Apart from the thickness of the material (T), two other parameters may be specified by a user to control the outcome of the finger joints. First, length of a joint (l) and second, tolerance of a joint (τ). The length marked in Fig. 10(a) is an indicative parameter, meaning that the actual length of a finger joint may not equal the specified length. The length is obtained by evenly partitioning an edge therefore will not be the same along different edges. The user-specified length is a guide for calculating a suitable number of partitions in an edge. The tolerance marked in Fig. 10(b) controls the degree of tightness of the finger joints. A zero tolerance [see Fig. 10(a)] usually leads to loose fit because some material will be removed during the cutting process. A suitable positive tolerance can compensate for this by enlarging the protruding joints and narrowing the indenting joints. The exact value should be tested for different materials and be based on different machine settings.

Finally, 3D data of all faces should be rotated to a plane, e.g. X-Y plane, and the output of the algorithm is 2D polygons with finger joints, as shown in Fig. 11. These data are the tool paths for a digital cutting machine. The planar faces with a thickness shown in Figs. 8–10 are for visualization evaluation.

4. Fabrication: volumes

The *Fresh Press* modeler was used to produce several watertight wooden models from starting solid mesh models. The solid cube in

Fig. 12 shows the process of laser cutting and manual assembly. Adhesives are not needed to sustain assembly joinery. From the starting mesh shape, all components are produced automatically including the internal bars. Cut sheets are generated as a secondary function of which includes a packing algorithm for polygonal items and a numbering system that is etched into each planar component.

Fabrication of a house model (Fig. 13) illustrates the generation of a physical model consisting of non-orthogonal planar components. Success is found in the ability of *Fresh Press* to create flat surfaces from the triangle mesh and associated finger joints at mutual edges, including those of high curvature. The original solid mesh model [Fig. 13(a)] contains a hole in the top of the chimney, the door and the windows. Openings in the original mesh are accounted for by *Fresh Press* as solid edges. Contouring methods described in Section 2 would not support successful evaluation of a shape of this type. Here a watertight solid with holes is possible. This one house demonstrates the range of shapes possible with *Fresh Press*.

Problems expressed in these models are related to the tolerance parameter between the components. Here a general tolerance value between finger joints is given for the model. First, smaller volumes such as the chimney assemble well and with ease. Second, longer surfaces such as the roof and roof underside take longer to assemble due to accumulated friction from the finger joints along the mutual edge. Last, parts with only a single edge,

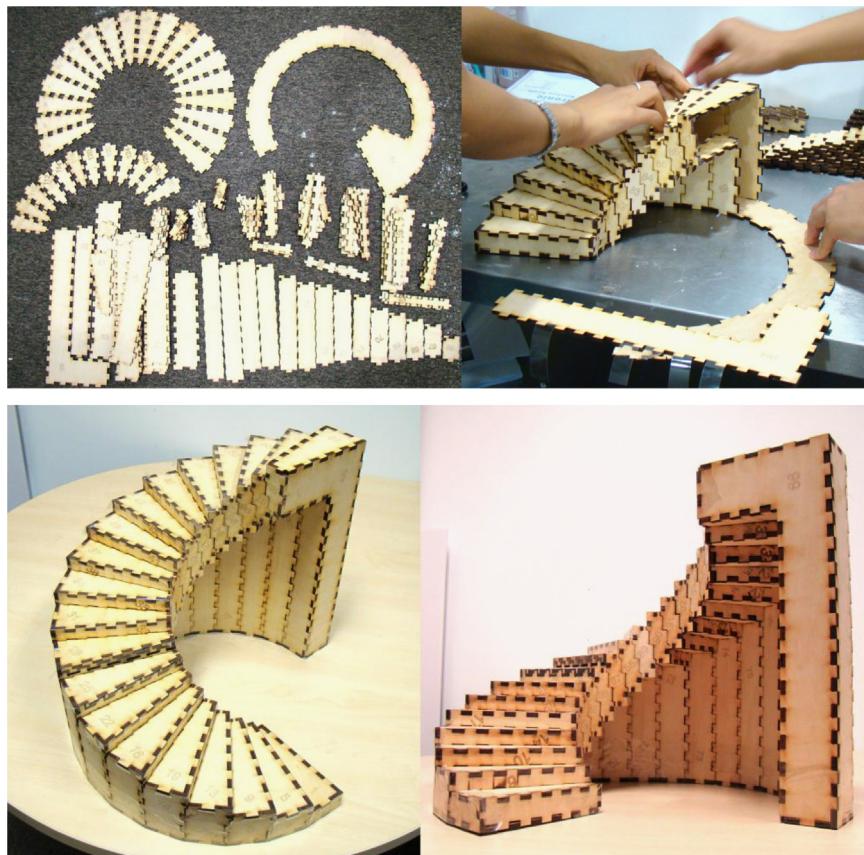


Fig. 15. Assembly of the physical model from planar components laser cut from 5 mm plywood sheets.

such as the door, would not hold from a single edge of friction [Fig. 13(d)]. A similar problem was found with the window attachment.

5. Fabrication: voids

Solid shapes generated by *Fresh Press* can also be used as molds for fabrication of shapes and modeled artifacts. A spiral model is introduced as a way to build an interlocking model as one solid from a curved model [Fig. 14(a)]. Steps were modeled as wedges around a central axis later sculpted into the shape of a spiral staircase. Next, finger joints were generated in 3D with interlocking surfaces along all sides of the model [Fig. 14(b)]. Then, 133 planar components were laser cut from 5 mm plywood sheets and were assembled manually (Fig. 15). It took two students less than 30 min to assemble the structure ($0.4 \times 0.4 \times 0.3 \text{ m}^3$) and they found this task easy to accomplish. The easy of assembly is largely due to the finger joints, which lock the components that are already in place. The students used transparent adhesive tape to further fix the structure during assembly.

The plywood staircase model was used to mold a plaster staircase of the same geometry. The plaster staircase was cast of silicon gel [Fig. 16(a)]. Fortunately, this relatively small model along with low viscosity silicon worked well to fill all voids. Once removed from its mold, the silicon gel provided many levels of model detail in a solid form [Fig. 16(c)] demonstrating high quality molding. Overall, it only took several hours to complete the entire casting process, which included automatic generation of finger joints based on the mesh model, laser cutting of the planar components, assembly of the plywood model, silicon gel casting, and disassembly of the plywood model.

6. Limitations

There are two limitations in our current implementation of the *Fresh Press* system. First, the planar-component-extraction process is rigid (Section 3.1), which only allows perfectly planar triangles to be grouped as a component. In the literature, there are more flexible methods [22,23] that can extract planar components from a dense triangle mesh. The components may be approximation of the mesh, which enables the processing of smoothly varying surfaces. Incorporating one of these methods into our system will significantly broaden the types of mesh that can be prototyped based on the finger joint structure. Second, the number associated with a component (Fig. 11) does not indicate the sequence of assembly. In fact, at this stage we cannot prove in principle that the generated structure can always be assembled, though assembly has not been a problem in all experiments conducted. Automatically generating a feasible assembly sequence will be useful when the number of components of a structure is very large.

7. Conclusion

A novel system on automatic generation of finger joints has been presented as a modeling tool for designers of physical product. The speed of the system demonstrated the potential of *Fresh Press* production of low fidelity physical models in a timely fashion. The input to the system is a 3D model consisting of planar components. The system analyzes the connectivity of the model and generates finger joints on the components for interconnection. The system also offers a series of parameters from which the output model can be modified for example thickness, tolerance and length of finger-joints.

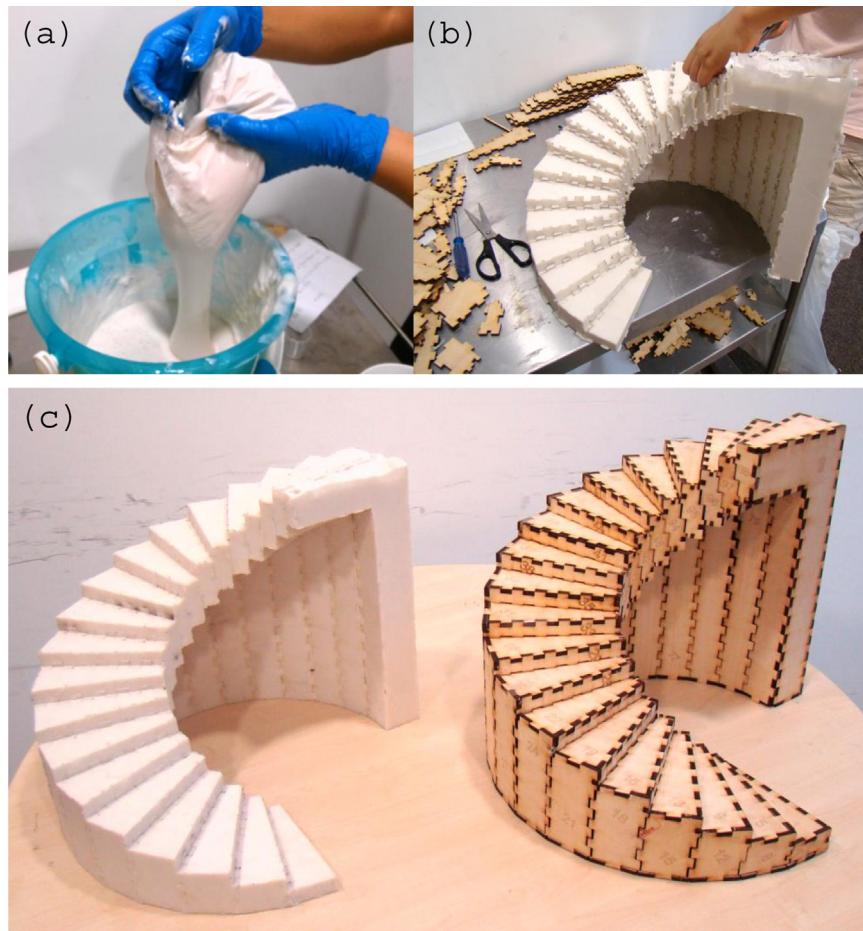


Fig. 16. Casting of a silicon gel model using the finger-joint structure as the mold.

Acknowledgments

This work is funded by the International Design Center at Singapore University of Technology and Design, and Ignition Grant (ING148082-ICT) of Singapore-MIT Alliance in Research and Technology (SMART).

References

- [1] Gerber Elizabeth, Carroll Maureen. The psychological experience of prototyping. *Des Stud* 2012;33(1):64–84.
- [2] Yang Maria C. A study of prototypes, design activity, and design outcome. *Des Stud* 2005;26(6):649–69.
- [3] VoxelJet. Voxeljet, 2014.
- [4] Oh Y, Johnson G, Gross Mark D, Do Ellen Yi-Luen. The Designosaur and the Furniture Factory: Simple software for fast fabrication. In: 2nd international conference on design computing and cognition (DCC06), 2006.
- [5] Mori Yuki, Igarashi Takeo. Plushie: an interactive design system for plush toys. In: ACM SIGGRAPH 2007 papers, SIGGRAPH'07. New York, NY, USA: ACM; 2007.
- [6] Igarashi Yuki, Igarashi Takeo, Mitani Jun. Beady: interactive beadwork design and construction. In: SIGGRAPH Asia 2011 sketches, SA '11. New York, NY, USA: ACM; 2011. p. 17:1–2.
- [7] Chen Desai, Sittithamorn Pitchaya, Lan Justin T, Matusik Wojciech. Computing and fabricating multiplanar models. In: Computer graphics forum, vol. 32. Wiley Online Library, 2013. p. 305–15.
- [8] Hanna Awad S. Concrete formwork systems. CRC Press; 1998.
- [9] Chan Yeung. Classic joints with power tools. Lark Books; 2002.
- [10] Kolarevic Branko. Designing and manufacturing architecture in the digital age. In: Architectural information management: 19th eCAADe conference proceedings, eCAADe: conferences. Helsinki, Finland: Helsinki University of Technology (HUT); , 08/2001 2001. p. 117–23.
- [11] Sass Lawrence. Wood frame grammar. In: Martens Bob, Brown Andre, editors. Computer aided architectural design futures 2005. Netherlands: Springer; 2005. p. 383–92.
- [12] Sass Lawrence. A wood frame grammar: a generative system for digital fabrication. *Int J Archit Comput* 2006;4(1):51–67.
- [13] Lau Manfred, Ohgawara Akira, Mitani Jun, Igarashi Takeo. Converting 3d furniture models to fabricatable parts and connectors. *ACM Trans Graph* 2011;30(July (4)):85:1–6.
- [14] Schulz Adriana, Shamir Ariel, Levin David IW, Sitti-amorn Pitchaya, Matusik Wojciech. Design and fabrication by example. *ACM Trans Graph* 2014;33(July (4)):62:1–62:11.
- [15] McCrae James, Singh Karan, Mitra Niloy J. Slices: a shape-proxy based on planar sections. In: Proceedings of the 2011 SIGGRAPH Asia conference, SA'11. New York, NY, USA: ACM; 2011. p. 168:1–12.
- [16] Saul Greg, Lau Manfred, Mitani Jun, Igarashi Takeo. Sketchchair: an all-in-one chair design system for end users. In: Proceedings of the fifth international conference on tangible, embedded, and embodied interaction, TEI'11. New York, NY, USA: ACM; 2011. p. 73–80.
- [17] Hildebrand Kristian, Bickel Bernd, Alexa Marc. Crdbrd: shape fabrication by sliding planar slices. *Comput Graph Forum* 2012;31(May (2)):583–92.
- [18] Le-Nguyen Tuong-Vu, Low Kok-Lim, Ruiz Conrado, Le Sang N. Automatic paper sliceform design from 3d solid models. *IEEE Trans Vis Comput Graph* 2013;19 (11):1795–807.
- [19] Schwartzburg Yuliy, Pauly Mark. Fabrication-aware design with intersecting planar pieces. *Comput Graph Forum* 2013;32(2):317–26.
- [20] Cignoni Paolo, Pietroni Nico, Malomo Luigi, Scopigno Roberto. Field-aligned mesh joinery. *ACM Trans Graph* 2014;33(February (1)):11:1–12.
- [21] Duckham Matt, Kulik Lars, Worboys Mike, Galton Antony. Efficient generation of simple polygons for characterizing the shape of a set of points in the plane. *Pattern Recognit* 2008;41(October (10)):3224–36.
- [22] Mitani Jun, Suzuki Hiromasa. Making papercraft toys from meshes using strip-based approximate unfolding. *ACM Trans Graph* 2004;23(August (3)):259–63.
- [23] Liu Y-J, Lai Y-K, Hu SM. Stripification of free-form surfaces with global error bounds for developable approximation. *IEEE Trans Autom Sci Eng* 2009;6 (4):700–9.