

FlexTruss: A Computational Threading Method for Multi-material, Multi-form and Multi-use Prototyping

Lingyun Sun
Zhejiang University, China

Jiaji Li
Zhejiang University, China

Yu Chen
Zhejiang University, China

Yue Yang
Zhejiang University, China

Zhi Yu
Zhejiang University, China

Danli Luo
Carnegie Mellon University, USA

Jianzhe Gu
Carnegie Mellon University, USA

Lining Yao
Carnegie Mellon University, USA

Ye Tao
Zhejiang University City College,
China

Guanyun Wang*
Zhejiang University, China

ABSTRACT

3D printing, as a rapid prototyping technique, usually fabricates objects that are difficult to modify physically. This paper presents FlexTruss, a design and construction pipeline based on the assembly of modularized truss-shaped objects fabricated with conventional 3D printers and assembled by threading. To create an end-to-end system, a parametric design tool with an optimal Euler path calculation method is developed, which can support both inverse and forward design workflow and multi-material construction of modular parts. In addition, the assembly of truss modules by threading is evaluated with a series of application cases to demonstrate the affordance of FlexTruss. We believe that FlexTruss extends the design space of 3D printing beyond typically hard and fixed forms, and it will provide new capabilities for designers and researchers to explore the use of such flexible truss structures in human-object interaction.

CCS CONCEPTS

• **Human-centered computing** → Human computer interaction (HCI); Interactive systems and tools.

KEYWORDS

Mesh structures, Shape-changing interfaces, Personal fabrication, 3D printing, Euler path

ACM Reference Format:

Lingyun Sun, Jiaji Li, Yu Chen, Yue Yang, Zhi Yu, Danli Luo, Jianzhe Gu, Lining Yao, Ye Tao, and Guanyun Wang. 2021. FlexTruss: A Computational

*Corresponding Author: guanyun@zju.edu.cn

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

CHI '21, May 08–13, 2021, Yokohama, Japan

© 2021 Association for Computing Machinery.

ACM ISBN 978-1-4503-8096-6/21/05...\$15.00

<https://doi.org/10.1145/3411764.3445311>

Threading Method for Multi-material, Multi-form and Multi-use Prototyping. In *CHI Conference on Human Factors in Computing Systems (CHI '21)*, May 08–13, 2021, Yokohama, Japan. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3411764.3445311>

1 INTRODUCTION

In the Human-Computer Interaction (HCI) community, truss structures are widely used for prototyping since it is stable, lightweight, and saves material [1, 18, 20]. Similarly, more researchers attempt to integrate truss structures with 3D printing to accelerate design iterations while maintaining stability [25, 46]. However, 3D printed models are difficult to modify physically as they are supposed to be a truthful resemblance of the input shape and are not designed for continuous, interactive fabrication (Figure 1). As a result, a growing interest in developing real-time design, fabrication, and modification methods that allow users to receive continuous feedback is revealed by recent works [30, 31, 51].

In this paper, we present FlexTruss, a design and fabrication pipeline that allows on-demand modifiable prototyping with modularized 3D printed components assembled by simple threading to reconstruct the original design. Each individual node on the FlexTruss structure is hollowed to incorporate a threading wire that traverses all nodes. A design tool, powered by optimal Eulerian circuit calculation coupled with multi-node generation and sorting, is developed to aid this process. We also experiment with various structures and multi-material systems to expand the physical properties and malleability of the assembled models for different uses. While our approach requires some manual assembly after the computer-aided design and 3D printing processes, it facilitates intuitive, instructive, and sensible design by allowing users to reflect on and tinker with the final artifact in their hands. We believe that FlexTruss extends the design space of 3D printing beyond typically hard and fixed forms, and it will provide new capabilities for designers and researchers to explore the use of such flexible truss structures in human-object interaction.

This paper makes the following contributions:

1. proposing an end-to-end fabrication method, including a parametric design tool, structural design of joint nodes, and

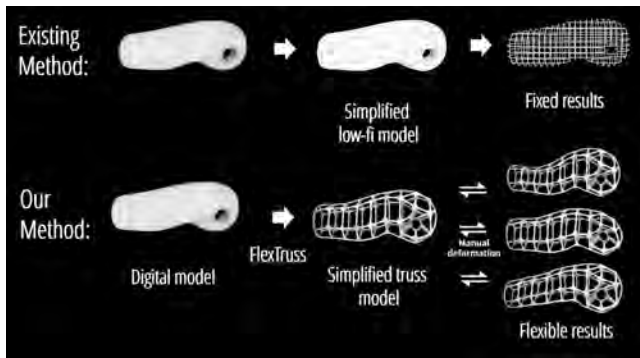


Figure 1: FlexTruss method for modifiable prototyping.

material option, that enables on-demand modifiable prototyping with a simple assembly process.

2. developing a parametric design tool powered by optimal Eulerian circuit algorithm, which supports both inverse and forward design workflows and multi-material construction of modular parts.
3. implementing a series of application cases to demonstrate the affordance of FlexTruss.

2 RELATED WORK

2.1 Interactive Fabrication

Recently, in HCI, researchers became more interested in the interaction between users and fabrication techniques. For example, Interactive Fabrication [45] gives users great flexibility in making adjustments and modifications in the early stage of prototyping. Inspired by traditional crafting which provides real-time physical feedback, researchers have developed multiple fabrication approaches and tools to augment users' interactivity and involvement in the process of sculpturing [51], claying [16], weaving [39], painting [34], wiring [47], and 3D printing [50]. In addition, a number of projects demonstrate that interactive fabrication would facilitate the integration of physical and digital design [44], crafting wearable objects directly on the body [8], designing and fabricating models in parallel [31], enabling 3D printing's tangibility [30, 32, 38], and releasing digital fabrication activity from the traditional fab shop [6]. Like these works, we draw inspiration from these hands-on modeling approaches and develop FlexTruss as we try to engage users in the process of fabrication.

2.2 Low-Fi Fabrication and Truss-Shaped Objects

The development in digital fabrication technology (e.g., 3D printing, laser cutting) has rendered rapid prototyping in small production volume more accessible, yet it still requires dozens of hours to fabricate a single artifact. For example, people usually print with a 3D printer overnight for a palm-sized artifact with reasonable resolution. To save fabrication time and speed up design iterations, researchers proposed a series of low-fidelity (low-fi) techniques such as 3D printing truss to abstract the model [25, 31], coupling 3D printing with Lego bricks [26], and stacking laser-cut plates to

substitute 3D printing [3]. Additionally, printing objects with 2D truss-shaped patterns then morphing to 3D structures has been demonstrated as a more efficient fabrication way in 4D printing related works [37, 40–43]. In the field of computer graphics, more researchers are interested in developing optimized truss structures that are easy to fabricate or functionally versatile [5, 11, 46, 48]. FlexTruss investigates and expands the boundary of low-fi fabrication by proposing a flexible truss-based prototyping method that allows users to modify 3D models after fabrication and accelerate design iterations.

2.3 Prototyping with Computational Line

Lines are the most basic building components existing both in the natural and artificial world. Researchers have taken the advantage of its simplicity from different perspectives. For instance, Nakagaki et al. explored the design space of line-shaped interfaces with skeletal actuation [27, 28]. Gu et al. [9, 10] and Zhao et al. [49] both leveraged linear continuous 3D printing path planning to achieve specific morphing functions or to improve fabrication quality and efficiency. Some researchers computationally designed and made accessories such as jewelry using wire as their main composition [12, 24]. Lira et al. prototyped wire sculptures with minimal overlap using Eulerian Path in a non-planar fashion [22]. Loe et al. also advanced the Eulerian Path algorithm to optimize the printing path for delicate sole fabrication [7]. Compared with these works, FlexTruss applies Eulerian Path to simplify the assembly process with only a single thread while maintaining the structural complexity and flexibility. Also, the design tool suite of FlexTruss lowers the barrier of design and fabrication for novices.

2.4 Flexible Fabrication and Construction Kit

Deconstructing a digital model into smaller kits to form a construction set, such as Lego, is well-developed in the area of digital fabrication. By taking advantage of joint structures, some researchers demonstrated that the construction kit facilitates the prototyping process of functional objects at different scales, such as for product-sized objects [20], furniture-sized artifacts [15], and room-sized installations [17, 19]. Flexibility is also a common concern in the domain of personal fabrication. As such, flexible mechanisms are incorporated and developed with inflatable construction [33, 35], mechanical meta-materials [2, 13, 14], compliant mechanisms [23], and embeddable modification [4]. FlexTruss combines the ease of construction and tunable mechanical properties of truss-like components made by 3D printing, which expands the possibilities of flexible fabrication.

3 DESIGN SPACE OF FLEXTRUSS

Here, we describe the design space of the flexible truss structure enabled by our framework which helps guide the employment of this method.

3.1 Shape Change

The modular nodes of the truss structure can be generated as fixed, semi-fixed, semi-flexible, and fully flexible connections (Table 1). A truss structure modularized into smaller pieces and more nodes has more flexibility, but the assembly is more complex and

time-consuming. Therefore, depending on the requirement of the flexibility of the final assembly, a combination of different types of nodes can be used to resemble fixed and deformable areas as needed.

3.2 Physical-Property Change

Although the truss units are usually rigid and made of hard plastic materials, the global physical properties of a FlexTruss structure can be changed by swapping the threading wire. For example, soft metal wires such as galvanized iron can be used to maintain deformation after manual sculpturing, plastic wires such as fishing line can soften the connections, and elastic wires can make it stretchable.

3.3 Media Embedding

The hollow pipes inside the truss units can be connected with different media to enhance the affordances and capabilities of FlexTruss. Instead of normal wires mentioned above, we consider optical fiber, conductive wire, and soft tubing that can perfuse liquid or gas. They may be used to construct wearable devices, interactive objects, or installation art pieces.

4 COMPUTATIONAL DESIGN

4.1 User Interface

Using Grasshopper, a parametric design plugin for Rhinoceros, we developed a user interface (Figure 2) for FlexTruss to allow users to visualize the threading path and customize node types for flexibility. It also allows users to input models to create case-specific solutions.

FlexTruss is designed with a few rudimental goals:

1. The system is expected to be usable following standard steps in fabrication and assembly.
2. The software, which needs to accommodate different shapes, should provide high customizability.
3. The technique should support as much shape-changing viability as possible. Users can then take full advantage of the flexibility and be more engaged in the design process.
4. FlexTruss should guide easy and rapid assembly.
5. The prototype can be made out of low-cost and accessible materials and tools.

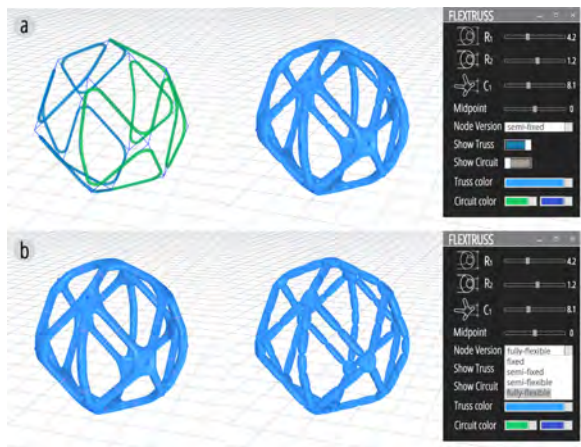


Figure 2: FlexTruss interface.

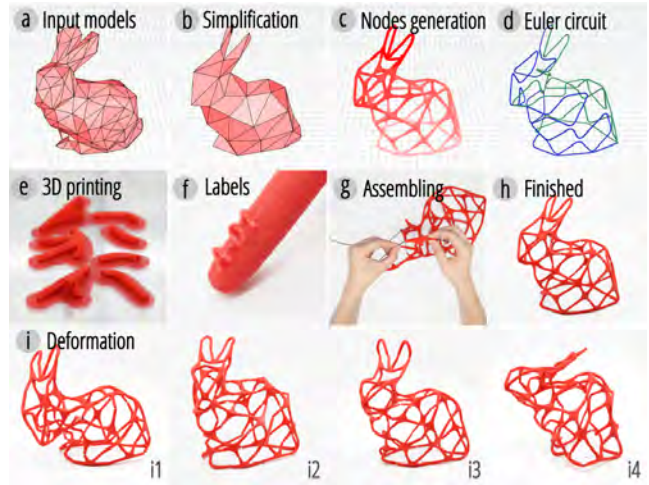


Figure 3: Inverse design workflow.

4.2 Inverse Design

In the inverse design process, our design tool consists of four steps: model simplification, nodes generation, Euler path calculation, and assembly guidance.

Firstly, after the user imports a 3D model (Figure 3a), the system simplifies the model into a truss mesh structure, with the density of the mesh being one of the adjustable parameters (Figure 3b). After previewing the simplified result, the user could tune the size of nodes on demand and check the node generation (Figure 3c). Meanwhile, the system generates a single path of connected pipes following a calculated Euler path that traverses all nodes (Figure 3d). The radius of the pipe varies based on the material of the wire. Eventually, the multi-directional joints are generated and labeled (Figure 3e, f). After making the joints (e.g., by filament deposition modeling), users can assemble the FlexTruss prototype following the labels (Figure 3g). Lastly, the user can manipulate the prototype with countless possibilities of deformation (Figure 3h).

4.3 Forward Design

In the forward design process (Figure 4), instead of importing a 3D model, users will constitute a pattern in the software. Our steps start with selecting nodes from the libraries provided in the design software (Figure 4a, b). A visualizer would help users to arrange and align color-coded nodes on an auxiliary grid (Figure 4c). The simplicity in the 2D pattern allows nodes to be modularized. Next, an Eulerian circuit is generated after the nodes have been set in place; the fabrication, assembly, and manipulation of the final artifact (Figure 4e, f, g) follow the same steps as in the inverse design workflow. Finally, as Figure 4h shows, users can deform the initial design into multiple variations.

4.4 Node Design for Flexible Structure

In the Inverse Design Workflow, we have shown that a semi-closed geometry (the Stanford Bunny) can be abstracted and fabricated with only semi-fixed nodes, a node type that allows orthogonal bending and stretching between adjacent nodes or pipes. However, to reassemble the closed geometry, the position of all the semi-fixed

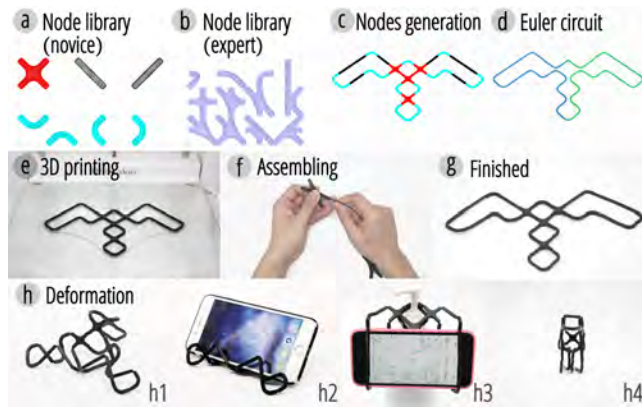


Figure 4: forward design workflow.

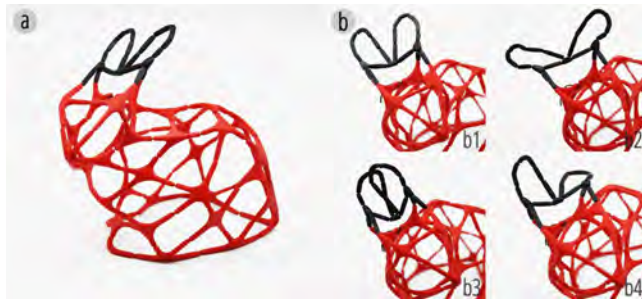


Figure 5: Stanford Bunny fabricated with (a) fully-flexible nodes in black for ears. (b) Fully-flexible nodes enhance the flexibility of ears.

nodes are uncompromising. Thus, in addition to the semi-fixed nodes, three types of nodes are developed. We classify the available node types into four categories - fully-flexible, semi-flexible, semi-fixed, and fixed, as tabulated in Table 1. They are defined by the degree of the node (number of edges that are incident to the node) and the number of links (flexible and open connector). Consequently, we can predict and utilize their shape-changing (bending and stretching) behavior to construct flexible structures. Nevertheless, the superior flexibility on joints reduces the fidelity of FlexTruss objects after deformation, which confounds the resemblance of the original input. A combination of different types of node is desired when users attempt to balance the superior flexibility on joints with the fidelity.

To demonstrate the flexibility enabled by combinations of multiple types of nodes, we modified the Stanford Bunny shown in Figure 3, which is constructed with semi-fixed nodes, and replaced the nodes on ears with fully-flexible nodes (Figure 5). This modification gives the ears a softer touch by adding more joints that allow deformation in more angles.

5 PIPELINE

FlexTruss pipeline is a general design system that deconstructs a simplified 3D geometry into truss-like nodes for 3D printing. Figure 6 shows an overview of the pipeline with an octahedron as an example, which is composed of eight equilateral triangles.

Step 1: Preparation for a Selected Mesh Model

This step adjusts a given geometry for further processing. We use the simplification function in Meshlab [52] as shown in Figure 3b for the Stanford Bunny. To ensure such geometry to be a Eulerian geometry, some edges are deleted uniformly until all vertices are 4- or 2-degree. The output is a special geometry $G = (V, E)$, which contains the set of vertices $V [v_0:(x_1, y_1, z_1), v_1:(x_2, y_2, z_2), \dots, v_t:(x_t, y_t, z_t)]$ and the line (v_1, v_2) representing an edge in the edge set E .

Step 2: Generating Nodes

Users-controlled parametric input for step 3 is $C^{\text{total}} = \{R_1, R_2, C_1\}$, which refers to the external radius, internal radius, and connection size of nodes. As Figure 6b shows that 4 filleted cylinders are generated by controlling R_1 and the length of E . Then, as the user tunes C_1 , the system shortens these cylinders to reserve room for node generation (Figure 6c). According to our experimental results shown in Figure 8, the threading material determines the arc of the connector. As Figure 6d-f shows, the connector generated from the point population can be joined with cylinders smoothly.

Step 3: Eulerian Circuit Algorithm

We define the five principles for a more intuitive threading strategy and user experience:

P1. Eulerian circuit: a single wire will connect all nodes.

P2. Bricklaying assembly: FlexTruss is assembled from bottom to top (Figure 6g, from pink nodes to red nodes).

P3. Midpoint assembling: user threads and fixes the first node in the middle (Figure 6h, red part) of a wire, then threads the rest node-by-node on the back (blue pipe) and front (green pipe).

P4. Non-crossing holing: the wire threading through a node multiple times will not cross itself (Figure 6i)

P5. Midpoint determination: the default midpoint is set on the lowest point, but the user could reset an arbitrary point as the first node (midpoint) or the last node (knot point).

P1 enables a FlexTruss prototype to be assembled tightly and uniformly with only one knot, while P2 minimizes the total distance of the threading wire used. P3 and P5 meet our daily habits of fabrication and make the assembly process more intuitive. P4 avoids the situations where two segments of the wire overlap in one node. We integrate the principles in a Python script (1).

The output is two sets of points, $(V_{R1}, V_{R2} \dots V_{Rt})$ and $(V_{L1}, V_{L2} \dots V_{Lt})$, which are then connected as 2 polylines. They are then filleted by C_1 and hollowed by R_2 .

Step 4: Print File Generation

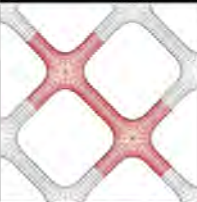

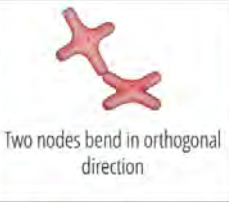

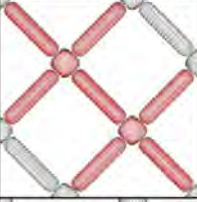
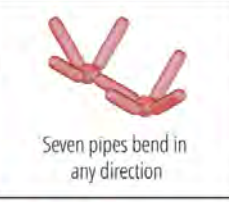
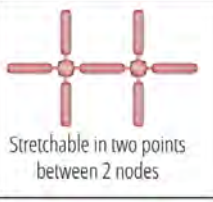
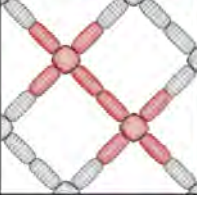

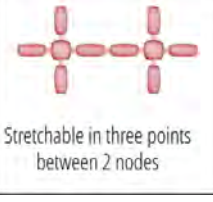
In this stage, the nodes are visualized and almost finished. The user can modify and swap the type of nodes manually for more specific deformation and application.

After that, with pipes inside all the nodes bridged by a Eulerian circuit, the software tool generates holes inside them using *Boolean difference* (Figure 6j). The user now obtains the final low-fi model virtually and exports an STL mesh file for 3D printing. Additionally, the sequence of assembly is also imprinted as labels on each node for assembly guidance (Figure 6k).

6 MULTI-MATERIAL FABRICATION

For the 3D printing setup, we employ an Ultimaker S5 desktop 3D printer with default settings - 0.15 mm layer thickness and 30% infill with triangular pattern - to print the nodes with a standard

Table 1: The four basic structural components of FlexTruss can be chosen from our design tool to generate the truss structure with on-demand flexibility.

	Features	Threading	Basic Structure	Shape-changing Behavior	
				Bending	Stretching
(a) Fixed	<ul style="list-style-type: none"> One 0-degree truss 	None		None	None
(b) Semi-Fixed	<ul style="list-style-type: none"> Two 4-degree nodes One link between two nodes 	4 times		 Two nodes bend in orthogonal direction	 Stretchable in one point between 2 nodes
(b) Semi-Flexible	<ul style="list-style-type: none"> Two 4-degree nodes with seven 2-degree pipes Two links between two nodes 	12 times		 Seven pipes bend in any direction	 Stretchable in two points between 2 nodes
(b) Fully-Flexible	<ul style="list-style-type: none"> Two 4-degree nodes with eight 2-degree pipes Three links between two nodes 	13 times		 Eight pipes bend in any direction	 Stretchable in three points between 2 nodes

thermoplastic material. As our system requires manual assembly, we test multiple threading wire materials for assembly and multiple tubing materials for replacing the printed modular part which can dramatically decrease the fabrication time.

6.1 Threading Line Material

Compared with the work Eulerian wire [22], the materials used for our prototype are not limited to metal wire, which makes this method suitable for a wide range of easy-to-cut materials, such as wires, threads, and fibers, with suppleness, strength, and even elasticity. The suppleness allows bending of FlexTruss objects without cracking. It also requires certain material strength that can support 3D shapes. Besides, certain elasticity would the objects to be stretched and worn as clothing.

More wiring and threading materials can be integrated with FlexTruss. Figure 7 shows three selected materials with different properties. Among these materials, the PVC coated galvanized wire is used in a wide variety of applications including packaging, shaping, and fencing. It is also easily deformable, maintains the shape after manual manipulation, and cheap (less than \$0.02 per

meter). Therefore, in the following experiments, we select it as the main threading material.

6.2 Threading Test

To optimize the assembly process, we conduct some threading experiments in two parts: 1) determining suitable pipe size for different wire diameters; 2) finding the minimum fillet radius that allows the wire to pass through at a specific angle.

In the experiment of determining internal radius (Figure 8), we find that sometimes the holes cannot be threaded though because of the instrumental error of 3D printing. Generally, in the virtual modeling process, reserving a 1.2 mm pipe for the thread material will leave a 0.6-0.8 mm hole after printing and ensure that most of the holes can be threaded through. We also print pipes (1.2 mm reserved) with different fillet radii from 3 mm to 6 mm, with a 0.5 mm increment. In this case, the exposed metal wire is the most difficult to thread because the sharp head may pierce through and get stuck inside the printed model. A fillet radius of 3.5 mm is the minimum for threading metal wires at 90°, while the 0° angle needs a 5 mm fillet radius.

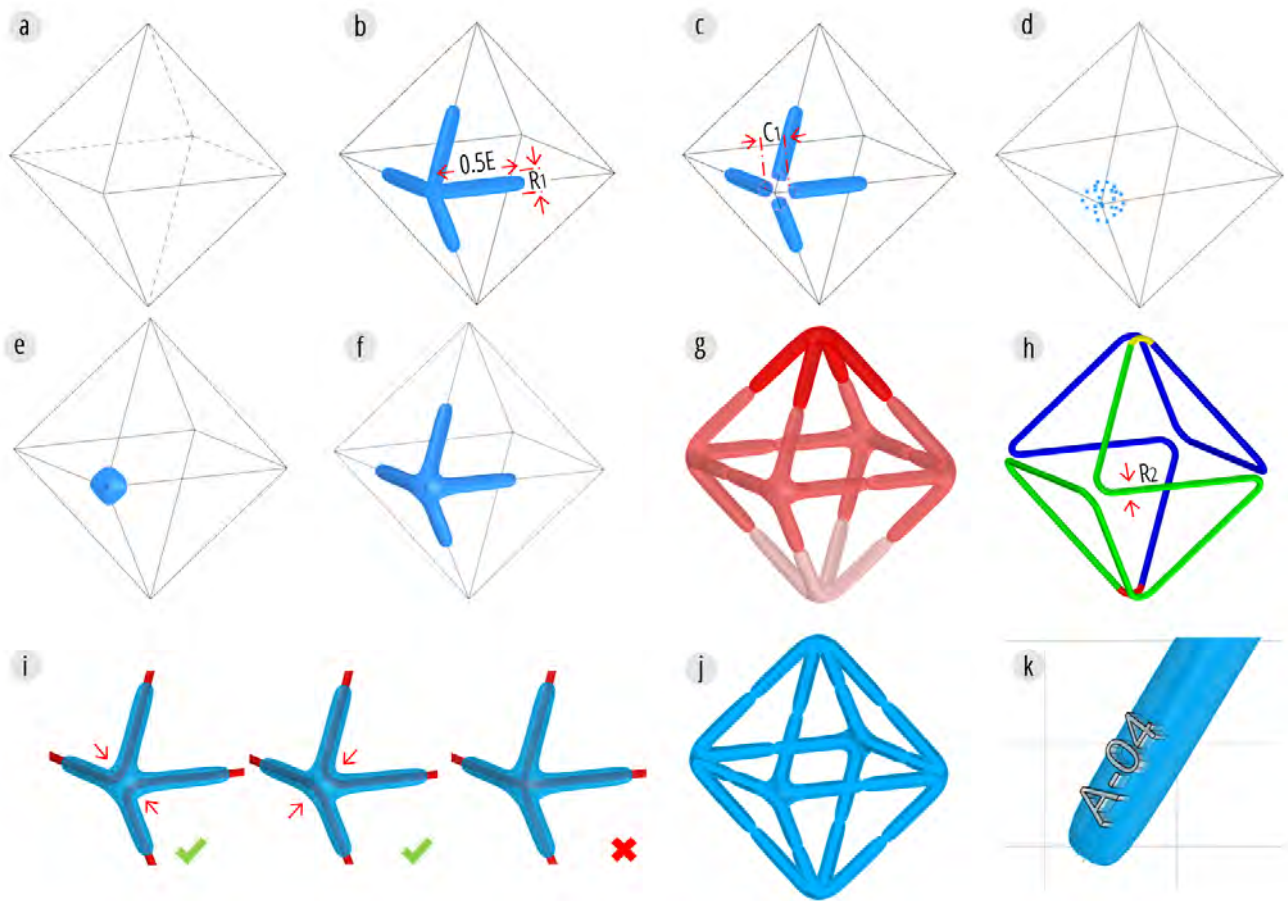


Figure 6: An example of the path generation on an octahedron (a). The octahedron is split into (b) 4 filleted cylinders, (c) shorted cylinders, (d) points population, and (e) connectors. A node is then generated (f), with (g) assembling sequence visualized for users. h) Eulerian circuit; (i) expected and unexpected nodes type; (j) nodes model after Boolean difference (k) final nodes with a label.

6.3 Replaceable Pipe Material

In Table 1, we propose fully-flexible and semi-flexible mechanisms, which separate nodes into smaller joints and multiple straight pipes. While the joint has to be customized by 3D printing, replacing pipes with other materials saves printing time and grants new usability. In Figure 9c, we print the joints and replace the pipes with rubber and thread them with a deformable wire, which greatly reduces the proportion of stiff parts. Such material choice increases the softness of FlexTruss and further brings the possibility for wearable prototypes.



Figure 7: The materials of threading wire: (a) PVC coated wire; (b) elastic rope; (c) optical fiber. (Scale bar: 10 mm)

6.4 Knotting Method

The last step of the assembly process is knotting. We form a knot to connect two ends of the wire after threading through the entire geometry (Figure 10). The length of the knot would not interfere with the whole structure. To hide the knot in the pipes, a split node is printed (Figure 10a) or a pipe of other material could be added to enclose the knot (Figure 10b). In addition, when threading with an elastic band such as TPU, the knot will be left outside of the node for later adjustments (Figure 10d).

7 MULTI-FORM DESIGN

By modular node generation and threading assembly with different types of wires, we can create a wide variety of geometrical shapes that can be embedded with shape-changing properties (Table 2).

7.1 Geometrical Primitives

In theory, FlexTruss can achieve any simplified shape from a digital geometric model. However, due to assembly constraints, the shape

Algorithm 1 Eulerian Circuit Algorithm

```

Input: Edge list E, Point position V, Middle point index m, Point
number pn
Output: Point index list L, Point list R
Global Point index list L, Point index list R, Boolean isfinish
Function DFS
  If all edges have been covered
    isfinish = TRUE
    Return
  EndIf
  Find the last 2 Points a, lasta from L and the last 2 Points b, lastb
  from R
  For all nexta connected point a
    If the line connected nexta and lasta UNCROSS the line
    connected other a's neighbor points
      Save nexta in SL
    EndIf
  EndFor
  Sort SL by z-axis value from minimum to maximum
  For all nextb connected point b
    If the line connected nextb and lastb UNCROSS the line
    connected other b's neighbor points
      Save nextb in SR
    EndIf
  EndFor
  Sort SR by z-axis value from minimum to maximum
  For all nexta in Sa and all nextb in Sb
    Save nexta in L, nextb in R
  Do Function DFS
  If isfinish = TRUE
    Return
  EndIf
  Remove nexta and nextb from L and R
EndFor
EndFunction
Function Main
  Initialize empty list L and R
  Save m both in L and R
  Select 2 points ma and mb connected point m with minimum
  z-axis value
  Save ma in L, mb in R
  Do Function DFS
  If isfinish = TRUE
    Return L, R
  EndIf
End

```

of some nodes may deviate from the input model (explained in *Sections 5 Pipeline* and *6.2 Threading Test*).

Generally, we can reconstruct a 3D shape (Table 2a-d) with 2D surfaces. For 3D shapes, both semi-closed surface (Table 2e) and fully-closed solid (Table 2f, g) can be translated into a truss structure with an inverse design workflow.

For the experiment in Table 2, the 1.2 mm-diameter PVC coated galvanized wire is used as the experimental material, and all the nodes are 3D printed and assembled.

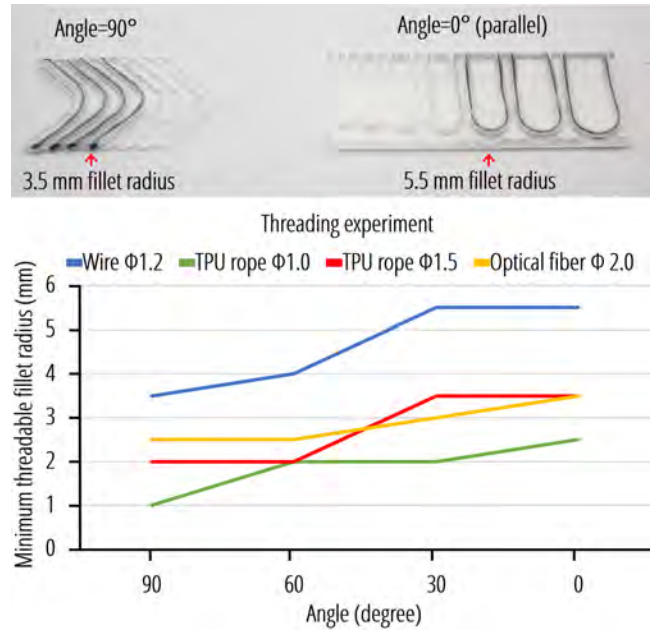


Figure 8: Results of the threading test with different materials.



Figure 9: The pipes are replaced with soft materials such as (a) acrylic; (b) PLA; (c) rubber. (Scale bar: 10 mm)



Figure 10: Knotting method: (a) a split node can hide the knot easily; (b) pipes can be cut for hiding the knot; (c) coiling two ends of the wire together for closing the loop; (d) a TPU knot.

7.2 Shape-changing Primitives






















Nodes and threading provide more possibilities for deformation. Here, we showcase the following primitives:

Foldable primitive (Table 2a, b): based on the bending behavior described in Table 1, origami structure can be easily generated with a forward design flow by adopting semi-flexible nodes.

Rollable primitive (Table 2d): tiling quadrilateral nodes can form a rollable structure with semi-flexible or fully flexible nodes.

Stretchable primitive (Table 2c): semi-flexible nodes arranged in a planar fashion can enable stretchable structure.

Table 2: Experimental primitives.

Name	Model	Prototype	Deformation	Node type	Nodes amount	Volume (L×W×H cm ³)	Assembly time
(a) Cube fold				Semi-flexible	14	16×12	7min
(b) Pyramid fold				Semi-flexible	6	11×9.5	4min
(c) Stretchable grid				Semi-flexible	13	10×13	7min
(d) Plane				Semi-fixed	38	21×14	16min
(e) Cylinder				Semi-fixed	40	10×10×20	20min
(f) Sphere				Semi-fixed	20	10×10×10	9min
(g) Annulus				Semi-fixed	30	12.5×12.5×5	12min

Volume-changing primitive (Table 2f): applying semi-fixed nodes on a sphere can create a low-fi Hoberman Sphere which changes its volume by rotating central symmetrical nodes.

Texture-changing primitive (Table 2g): half-closed FlexTruss object can deform in multiple directions, which will transform a smooth surface into an uneven one.

8 APPLICATION EXAMPLES

To situate FlexTruss in a range of use cases, we implement four application examples in wearable design, modular toy design, fashion design, and architecture scale model.

8.1 Wearable Devices

Since the flexibility of FlexTruss is directly related to the material properties of the threading wire, we thread FlexTruss with an elastic rope to demonstrate a wearable device design. We complete the prototype by reconstructing a wristband with 40 semi-flexible nodes (Figure 11 a-c). The artifact can be worn by users with different wrist sizes and deformed according to the wearer's action. In Figure 11d, and e, we replace all nodes with fully flexible nodes, which maintain the flexibility even when the elbow is fully bent. By embedding an optical fiber connected to a light source and a controller, this

prototype can provide a real-time response to the wearer's motion during exercise (Figure 11f).

8.2 Modular Toy

FlexTruss can be combined with off-the-shelf toys such as Lego. Lego bricks are readily expandable. FlexTruss can be designed to infuse compliant mechanisms to the powerful Lego system and hence increase its interactivity. Figure 12 shows two examples of using FlexTruss as added components to the block-shaped toys. Additionally, the shape, color, and posture of the FlexTruss add-ons can be easily manipulated.

8.3 Fashion Design

The forward design workflow empowers designers to create on-demand fashion pieces. Figure 13 shows an example of using FlexTruss as a handbag. Users can shape the handbag as functionally needed and even add straps to extend its functionality.

8.4 Architecture

FlexTruss is demonstrated as a feasible prototyping method in the field of architecture, which allows users to demo and modify the shape of a scale model of a building to conceptualize their design. A

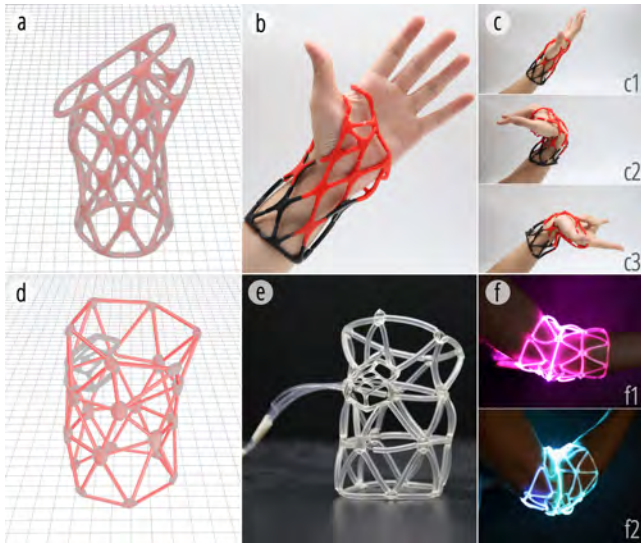


Figure 11: (a, d) Virtually generated nodes for wearable devices: a wristband (b, c) can adapt to the user’s hand gesture, and an armband (e, f) can give color-changing feedback using an optical fiber as the threading wire.

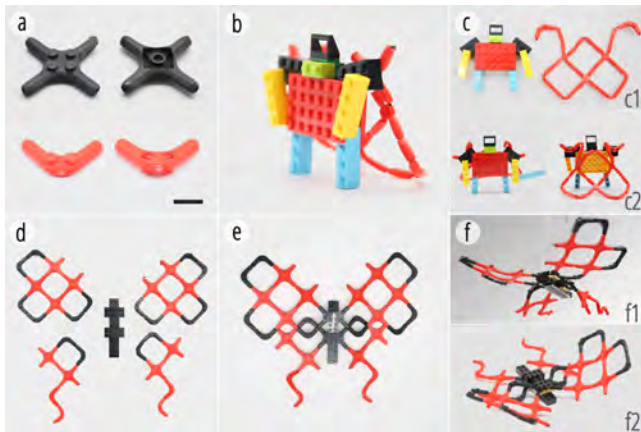


Figure 12: Modular toys made of FlexTruss: (a) block-shaped components; (b-c) a flexible cape is made for a Lego block mannequin; (d-f) flexible and bendable attachment as butterfly wings attached to a Lego block body. (Scale bar: 10 mm)

stadium with 42 nodes and galvanized wire is assembled as shown in Figure 14

9 DISCUSSION, LIMITATION, AND FUTURE IMPROVEMENT

We have showcased the potential of FlexTruss streamlining the fabrication process for users to exercise their design practice. Here we consider some aspects in the current state of FlexTruss that warrant future research efforts.



Figure 13: (a) A flat FlexTruss assembly can be transformed into (b) an envelope-shaped bag or (c) a cup holder. (d-f) adding straps to the same FlexTruss sheet will turn it into a shoulder bag.



Figure 14: Architecture design enabled by FlexTruss: (b) an architect can input a stadium (adapted from [36]) to the design platform, and (b) nodes will be generated; (c) the stadium as fabricated; (d) the architect can manipulate the FlexTruss stadium to illustrate their design to the audience.

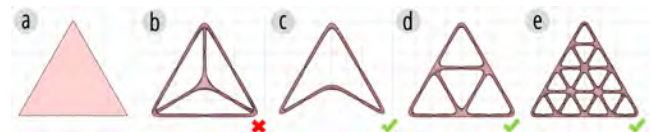


Figure 15: (a) an input triangle; (b) an ideal but unpractical solution for four 3-degree nodes; (c) current simplified result of 4 nodes; (d, e) practical examples by triangular tessellation.

9.1 Shape and Dimension

Shape limitation: Theoretically, our pipeline can simplify any mesh with a filter in MeshLab: quadric edge collapse decimation. However, a small node number may produce an unexpected result. As shown in Figure 15, if a user tries to simplify a triangular surface with only four nodes, an ideal and intuitive result would be a truss connecting three corners and the geometric center (Figure 15b), which cannot be threaded by a Eulerian circuit. The current software will output Figure 15c in such a situation. Though Figures 15d and 15e are the practical solutions for a flexible triangle, FlexTruss cannot indefinitely minimize the number of nodes in a given geometry. In future work, we may implement the Chinese Postman Algorithm [29] to visit every edge with the smallest number of duplicated graph edges. Thus, non-Eulerian geometries could also be fabricated with FlexTruss.

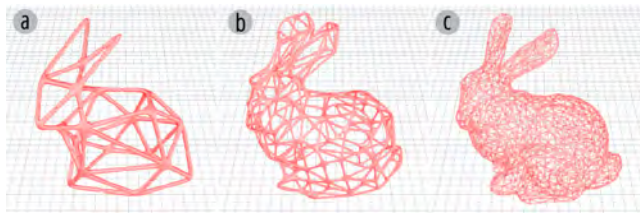


Figure 16: The Stanford Bunny is simplified with different node numbers: (a) 20, (b) 100, and (c) 1000.

General utility: The flexibility and fidelity of a prototype highly depend on the number of nodes, which is one of the most important parameters of our software. Generally, adding one additional 4-degree node (semi-fixed) in an existing prototype will need two more flexible joints, while a semi-flexible node will need four joints, and a fully flexible node will need six joints. In the example of the Stanford Bunny (12x17x23 cm), the fidelity of the model increases with the number of nodes assigned to the input (Figure 16).

Figure 16a and 16b produce outputs that are reasonably easy to assemble. On the other hand, though 1000 nodes (Figure 16c) will generate a close resemblance of the Stanford Bunny, the Eulerian circuit would be too long (24 m) for one artifact and the undersized nodes could not be printed.

Node dimension: As there is a minimum value of fillet radius on nodes for threading (Figure 8), the minimum size of nodes can be calculated as 2 times the fillet radius plus the wall thickness. Empirically, for a semi-fixed node, the minimum size is 13 mm in length, 13 mm in width, and 4 mm in height; for a semi-flexible node, the minimum size is 6 mm in length and width and 4 mm in thickness.

9.2 Fabrication

3D Printing: Although we propose a strategy to replace the pipes in FlexTruss for reducing the printing time, the hollow structure inside of the nodes will inevitably increase the difficulty and failure rate in 3D printing. On the other hand, the printing time needs to be budgeted for before the assembly.

Assembly: The Eulerian circuit imprinted in the design process guides and eases the assembly process. While a large-volume prototype with 120 nodes requires a 7-meter wire to thread through, the increase in wire length will complicate the assembly due to the additional friction between nodes. Thus, large-volume models are expected to be split into multiple circuits, which is a function to be integrated into the system.

Replacement: As the whole model is assembled with only one thread, replacing a broken node follows the dismantling of the entire structure. To solve this problem, we can add a new function in the software to instruct users to add switchable or replaceable nodes for quick and easy disassembly.

9.3 Interactivity

Deformation: Although the four different types of nodes can achieve different degrees of deformation, the deformability of a fully closed object or surface will be reduced by node collision. Thus, more

parametric studies can be planned to control or unleash more deformation behavior, such as tuning the gap distance between two nodes and introducing more structural design of nodes and pipes.

Media embedding: While our application example has shown the possibility of embedding special wires to transmit information inside the pipes (Figure 11f), the pipe space is still under-explored for additional interactivity. We envision, for example, that a liquid-infiltrated plastic pipe could be embedded following the geometry guided by FlexTruss. Yet, the requirement on the internal radius allowing such pipe to pass through must be met.

User interface: The current user interface is built with the grasshopper plugin in Rhinoceros, yet we believe a graphic user interface would make the design process more intuitive and more user-friendly. Furthermore, we plan on conducting a user evaluation including a user test on software rationality, assembly guidance, and assembly time.

10 CONCLUSION

In this work, we have shown that threading compliant and flexible components brings the possibility of integrating multiple functions into one prototype for makers and designers using a consumer-grade 3D printer. While today's 3D printers are capable of fabricating a fixed object, we present 3D printed FlexTruss assembled from modular parts for repeatable modification, partial replacement, and continuous creation in an active way. We developed a computational design tool to create truss structures, which can be directly 3D printed as well as easily assembled. We fabricated several application examples using our tool to explore multi-material, multi-form, and multi-use applications with shape-changing, physical property-changing, and media embedding features. We look forward to extending our method into future sustainable fabrication.

ACKNOWLEDGMENTS

This paper is supported by the National Science and Technology Innovation 2030 Major Project (2018AAA0100703) of the Ministry of Science and Technology of China, National Natural Science Foundation of China (No. 62002321), and the Provincial Key Research and Development Plan of Zhejiang Province (No. 2019C03137).

REFERENCES

- [1] Harshit Agrawal, Udayan Umapathi, Robert Kovacs, Johannes Frohnhofen, Hsiang-Ting Chen, Stefanie Mueller, and Patrick Baudisch. 2015. Prototyper: Physically Sketching Room-Sized Objects at Actual Scale. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15), 427–436. <https://doi.org/10.1145/2807442.2807505>
- [2] Davide Jose Nogueira Amorim, Troy Nachtigall, and Miguel Bruns Alonso. 2019. Exploring mechanical meta-material structures through personalised shoe sole design. In Proceedings of the ACM Symposium on Computational Fabrication, 1–8. <https://doi.org/10.1145/3328939.3329001>
- [3] Dustin Beyer, Serafima Gurevich, Stefanie Mueller, Hsiang-Ting Chen, and Patrick Baudisch. 2015. Platener: Low-Fidelity Fabrication of 3D Objects by Substituting 3D Print with Laser-Cut Plates. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15), 1799–1806. <https://doi.org/10.1145/2702123.2702225>
- [4] Xiang “Anthony” Chen, Stelian Coros, and Scott E. Hudson. 2018. Medley: A Library of Embeddables to Explore Rich Material Properties for 3D Printed Objects. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 1–12. <https://doi.org/10.1145/3173574.3173736>
- [5] Paolo Cignoni, Nico Pietroni, Luigi Malomo, and Roberto Scopigno. 2014. Field-Aligned Mesh Joinery. ACM Trans. Graph. 33, 1. <https://doi.org/10.1145/2537852>

- [6] Laura Devendorf and Kimiko Ryokai. 2015. Being the Machine: Reconfiguring Agency and Control in Hybrid Fabrication. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15), 2477–2486. <https://doi.org/10.1145/2702123.2702547>
- [7] Loe Feijs, Troy Nachtigall, and Oscar Tomico. Sole Maker: Towards Ultra-personalised Shoe Design Using Voronoi Diagrams and 3D Printing. 11.
- [8] Madeline Gannon, Tovi Grossman, and George Fitzmaurice. 2016. ExoSkin: On-Body Fabrication. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), 5996–6007. <https://doi.org/10.1145/2858036.2858576>
- [9] Jianzhe Gu, David E. Breen, Jenny Hu, Lifeng Zhu, Ye Tao, Tyson Van de Zande, Guanyun Wang, Yongjie Jessica Zhang, and Lining Yao. 2019. Geodesy: Self-Rising 2.5D Tiles by Printing along 2D Geodesic Closed Path. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19), 1–10. <https://doi.org/10.1145/3290605.3300267>
- [10] Jianzhe Gu, Vidya Narayanan, Guanyun Wang, Danli Luo, Harshika Jain, Kexin Lu, Fang Qin, Sijia Wang, James McCann, and Lining Yao. 2020. Inverse Design Tool for Asymmetrical Self-Rising Surfaces with Color Texture. In Symposium on Computational Fabrication (SCF '20). <https://doi.org/10.1145/3424630.3425420>
- [11] Yijiang Huang, Juyong Zhang, Xin Hu, Guoxian Song, Zhongyuan Liu, Lei Yu, and Ligang Liu. 2016. FrameFab: Robotic Fabrication of Frame Shapes. ACM Trans. Graph. 35, 6. <https://doi.org/10.1145/2980179.2982401>
- [12] Emmanuel Iarussi, Wilnot Li, and Adrien Bousseau. 2015. WrapIt: Computer-Assisted Crafting of Wire Wrapped Jewelry. ACM Trans. Graph. 34, 6. <https://doi.org/10.1145/2816795.2818118>
- [13] Alexandra Ion, David Lindlbauer, Philipp Herholz, Marc Alexa, and Patrick Baudisch. 2019. Understanding Metamaterial Mechanisms. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19), 1–14. <https://doi.org/10.1145/3290605.3300877>
- [14] Alexandra Ion, Ludwig Wall, Robert Kovacs, and Patrick Baudisch. 2017. Digital Mechanical Metamaterials. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 977–988. <https://doi.org/10.1145/3025453.3025624>
- [15] Alec Jacobson. 2019. RodSteward: A Design-to-Assembly System for Fabrication using 3D-Printed Joints and Precision-Cut Rods.
- [16] Michael D. Jones, Kevin Seppi, and Dan R. Olsen. 2016. What You Sculpt is What You Get: Modeling Physical Interactive Devices with Clay and 3D Printed Widgets. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), 876–886. <https://doi.org/10.1145/2858036.2858493>
- [17] Robert Kovacs, Alexandra Ion, Pedro Lopes, Tim Oesterreich, Johannes Filter, Philipp Otto, Tobias Arndt, Nico Ring, Melvin Witte, Anton Synytsia, and Patrick Baudisch. 2019. TrussFormer: 3D Printing Large Kinetic Structures. In Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19), 1. <https://doi.org/10.1145/3290607.3311766>
- [18] Robert Kovacs, Anna Seufert, Ludwig Wall, Hsiang-Ting Chen, Florian Meinel, Willi Müller, Sijing You, Maximilian Brehm, Jonathan Striebel, Yannis Kommana, Alexander Popiak, Thomas Bläsius, and Patrick Baudisch. 2017. TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 2606–2616. <https://doi.org/10.1145/3025453.3026016>
- [19] Robert Kovacs, Anna Seufert, Ludwig Wall, Hsiang-Ting Chen, Florian Meinel, Willi Müller, Sijing You, Maximilian Brehm, Jonathan Striebel, Yannis Kommana, Alexander Popiak, Thomas Bläsius, and Patrick Baudisch. 2017. TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 2606–2616. <https://doi.org/10.1145/3025453.3026016>
- [20] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17), 471–479. <https://doi.org/10.1145/3126594.3126643>
- [21] Wallace Lira, Chi-Wing Fu, and Hao Zhang. 2019. Fabricable eulerian wires for 3D shape abstraction. ACM Transactions on Graphics 37, 6: 1–13. <https://doi.org/10.1145/3272127.3275049>
- [22] Vittorio Megaro, Jonas Zehnder, Moritz Bäcker, Stelian Coros, Markus Gross, and Bernhard Thomaszewski. 2017. A Computational Design Tool for Compliant Mechanisms. ACM Trans. Graph. 36, 4. <https://doi.org/10.1145/3072959.3073636>
- [23] Eder Miguel, Mathias Lepoutre, and Bernd Bickel. 2016. Computational Design of Stable Planar-Rod Structures. ACM Trans. Graph. 35, 4. <https://doi.org/10.1145/2897824.2925978>
- [24] Stefanie Mueller, Sangha Im, Serafima Gurevich, Alexander Teibrich, Lisa Pfisterer, François Guimbretière, and Patrick Baudisch. 2014. WirePrint: 3D printed previews for fast prototyping. In Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14, 273–280. <https://doi.org/10.1145/2642918.2647359>
- [25] Stefanie Mueller, Tobias Mohr, Kerstin Guenther, Johannes Frohnhofen, and Patrick Baudisch. 2014. faBrickation: fast 3D printing of functional objects by integrating construction kit building blocks. In Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14, 3827–3834. <https://doi.org/10.1145/2556288.2557005>
- [26] Ken Nakagaki, Artem Dementyev, Sean Follmer, Joseph A. Paradiso, and Hiroshi Ishii. 2016. ChainFORM: A Linear Integrated Modular Hardware System for Shape Changing Interfaces. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16), 87–96. <https://doi.org/10.1145/2984511.2984587>
- [27] Ken Nakagaki, Sean Follmer, and Hiroshi Ishii. 2015. LineFORM: Actuated Curve Interfaces for Display, Interaction, and Constraint. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15), 333–339. <https://doi.org/10.1145/2807442.2807452>
- [28] Christos H. Papadimitriou. 1976. On the Complexity of Edge Traversing. J. ACM 23, 3: 544–554. <https://doi.org/10.1145/321958.321974>
- [29] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 2018. RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18), 1–12. <https://doi.org/10.1145/3173574.3174153>
- [30] Huaishu Peng, Rundong Wu, Steve Marschner, and François Guimbretière. 2016. On-The-Fly Print: Incremental Printing While Modelling. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, 887–896. <https://doi.org/10.1145/2858036.2858106>
- [31] Huaishu Peng, Amit Zoran, and François V. Guimbretière. 2015. D-Coil: A Hands-on Approach to Digital 3D Models Design. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15), 1807–1815. <https://doi.org/10.1145/2702123.2702381>
- [32] Hiroki Sato, Young ah Seong, Ryosuke Yamamura, Hiromasa Hayashi, Katsuhiko Hata, Hisato Ogata, Ryuma Niiyama, and Yoshihiro Kawahara. 2020. Soft yet Strong Inflatable Structures for a Foldable and Portable Mobility. In Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20), 1–4. <https://doi.org/10.1145/3334480.3383147>
- [33] Roy Shilkrot, Pattie Maes, Joseph A. Paradiso, and Amit Zoran. 2015. Augmented Airbrush for Computer Aided Painting (CAP). ACM Trans. Graph. 34, 2. <https://doi.org/10.1145/2699649>
- [34] Méline Skouras, Bernhard Thomaszewski, Peter Kaufmann, Akash Garg, Bernd Bickel, Eitan Grinspun, and Markus Gross. 2014. Designing Inflatable Structures. ACM Trans. Graph. 33, 4. <https://doi.org/10.1145/2601097.2601166>
- [35] Sporadical. Dolní Břežany Sports Hall. Retrieved from <https://www.archdaily.com/889816/dolni-brežany-sports-hall-sporadical>
- [36] Lingyun Sun, Jiaji Li, Yu Chen, Yue Yang, Ye Tao, Guanyun Wang, and Lining Yao. 2020. 4DTexture: A Shape-Changing Fabrication Method for 3D Surfaces with Texture. In Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, 1–7. <https://doi.org/10.1145/3334480.3383053>
- [37] Haruki Takahashi and Jeeun Kim. 2019. 3D Pen + 3D Printer: Exploring the Role of Humans and Fabrication Machines in Creative Making. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19), 1–12. <https://doi.org/10.1145/3290605.3300525>
- [38] Ye Tao, Guanyun Wang, Caowei Zhang, Nannan Lu, Xiaolian Zhang, Cheng Yao, and Fangtian Ying. 2017. WeaveMesh: A Low-Fidelity and Low-Cost Prototyping Approach for 3D Models Created by Flexible Assembly. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17), 509–518. <https://doi.org/10.1145/3025453.3025699>
- [39] Guanyun Wang, Tingyu Cheng, Youngwook Do, Humphrey Yang, Ye Tao, Jianzhe Gu, Byoungkwon An, and Lining Yao. 2018. Printed Paper Actuator: A Low-Cost Reversible Actuation and Sensing Method for Shape Changing Interfaces. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18), 1–12. <https://doi.org/10.1145/3173574.3174143>
- [40] Guanyun Wang, Fang Qin, Haolin Liu, Ye Tao, Yang Zhang, Yongjie Jessica Zhang, and Lining Yao. 2020. MorphingCircuit: An Integrated Design, Simulation, and Fabrication Workflow for Self-Morphing Electronics. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4, 4. <https://doi.org/10.1145/3432232>
- [41] Guanyun Wang, Ye Tao, Ozguc Bertug Capunaman, Humphrey Yang, and Lining Yao. 2019. A-Line: 4D Printing Morphing Linear Composite Structures. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19), 1–12. <https://doi.org/10.1145/3290605.3300656>
- [42] Guanyun Wang, Humphrey Yang, Zeyu Yan, Nurcan Geceer Ulu, Ye Tao, Jianzhe Gu, Levent Burak Kara, and Lining Yao. 2018. 4DMesh: 4D Printing Morphing Non-Developable Mesh Surfaces. In The 31st Annual ACM Symposium on User Interface Software and Technology - UIST '18, 623–635. <https://doi.org/10.1145/3242587.3242625>
- [43] Christian Weichel, John Hardy, Jason Alexander, and Hans Gellersen. 2015. ReForm: Integrating Physical and Digital Design through Bidirectional Fabrication. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15, 93–102. <https://doi.org/10.1145/2807442.2807451>
- [44] Karl D.D. Willis, Cheng Xu, Kuan-Ju Wu, Golan Levin, and Mark D. Gross. 2010. Interactive Fabrication: New Interfaces for Digital Fabrication. In Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11), 69–72. <https://doi.org/10.1145/1935701.1935716>

- [45] Rundong Wu, Huaishu Peng, François Guimbretière, and Steve Marschner. 2016. Printing Arbitrary Meshes with a 5DOF Wireframe Printer. *ACM Trans. Graph.* 35, 4. <https://doi.org/10.1145/2897824.2925966>
- [46] Ya-Ting Yue, Xiaolong Zhang, Yongliang Yang, Gang Ren, Yi-King Choi, and Wenping Wang. 2017. WireDraw: 3D Wire Sculpturing Guided with Mixed Reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*, 3693–3704. <https://doi.org/10.1145/3025453.3025792>
- [47] Jonas Zehnder, Stelian Coros, and Bernhard Thomaszewski. 2016. Designing Structurally-Sound Ornamental Curve Networks. *ACM Trans. Graph.* 35, 4. <https://doi.org/10.1145/2897824.2925888>
- [48] Haisen Zhao, Fanglin Gu, Qi-Xing Huang, Jorge Garcia, Yong Chen, Changhe Tu, Bedrich Benes, Hao Zhang, Daniel Cohen-Or, and Baoquan Chen. 2016. Connected Fermat Spirals for Layered Fabrication. *ACM Trans. Graph.* 35, 4. <https://doi.org/10.1145/2897824.2925958>
- [49] Amit Zoran. 2013. Hybrid Basketry: Interweaving Digital Practice within Contemporary Craft. In *ACM SIGGRAPH 2013 Art Gallery (SIGGRAPH '13)*, 324–331. <https://doi.org/10.1145/2503649.2503651>
- [50] Amit Zoran and Joseph A. Paradiso. 2013. FreeD: A Freehand Digital Sculpting Tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*, 2613–2616. <https://doi.org/10.1145/2470654.2481361>
- [51] Meshlab. Retrieved September 18, 2020 from <https://www.meshlab.net/>