Rhapso: Automatically Embedding Fiber Materials into 3D Prints for Enhanced Interactivity

Daniel Ashbrook
dan@di.ku.dk
University of Copenhagen
Copenhagen, Denmark

Wei-Ju Lin
wei-ju.lin@hci.csie.ntu.edu.tw
National Taiwan University
Taipei, Taiwan

Nicholas Bentley
nxb484@student.bham.ac.uk
University of Birmingham
Birmingham, United Kingdom

Diana Soponar
soponar.diana@gmail.com
University of Copenhagen
Copenhagen, Denmark

Valkyrie Savage
vasa@di.ku.dk
University of Copenhagen
Copenhagen, Denmark

Zeyu Yan
zeyuy@umd.edu
University of Maryland
College Park, USA

Lung-Pan Cheng
lung-pan.cheng@csie.ntu.edu.tw
National Taiwan University
Taipei, Taiwan

Huaishu Peng
huaishu@cs.umd.edu
University of Maryland
College Park, USA

Hyunyoung Kim
h.kim.4@bham.ac.uk
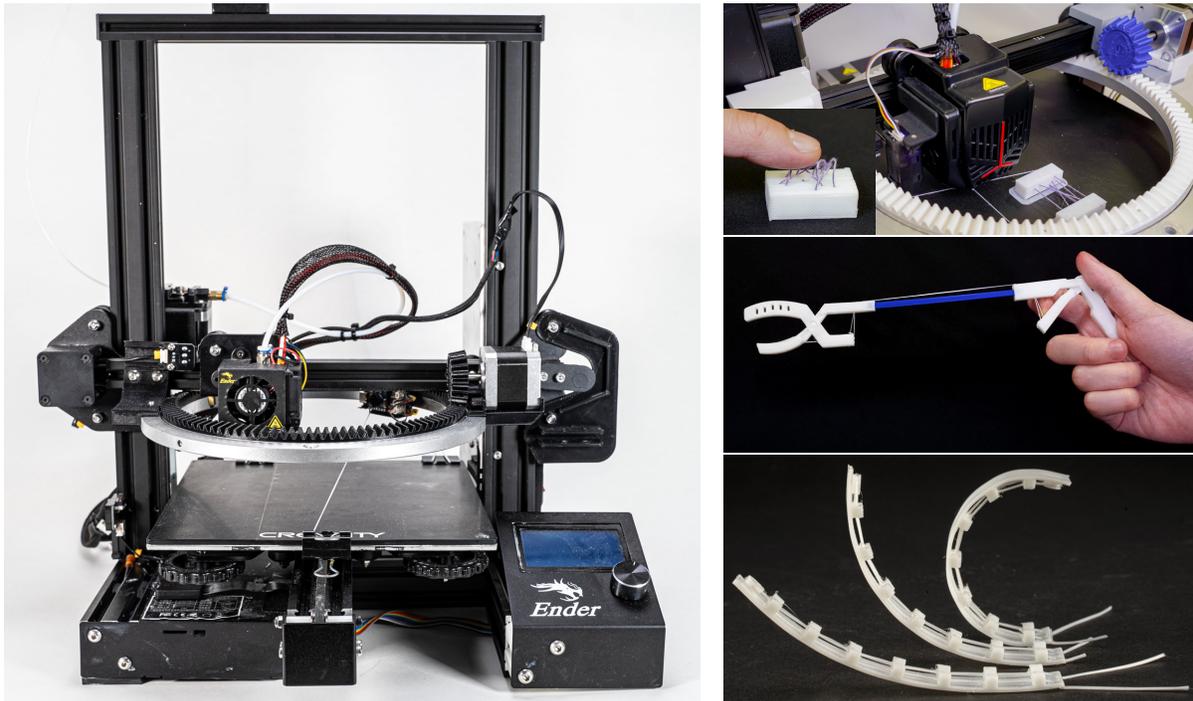University of Birmingham
Birmingham, United Kingdom

Figure 1: Rhapso allows embedding various types of fibers into 3D prints to enable quick fabrication of multifunctional objects. A) Rhapso hardware made by modifying a consumer grade 3D printer. An added fiber spool is moved along the attached ring. B-D) Objects created by Rhapso. B) Hairy object. A cotton fiber is embedded between two cuboids and creates hairy texture. C) Grabber. The white elastic thread transfers users' pinch to the extended grabber. D) Test of tension control system. The same 3D design can result in different shapes by controlling the tension of the elastic fiber during printing.

ABSTRACT

We introduce Rhapso, a 3D printing system designed to embed a diverse range of continuous fiber materials within 3D objects during the printing process. This approach enables integrating properties like tensile strength, force storage and transmission, or aesthetic and tactile characteristics, directly into low-cost thermoplastic 3D prints. These functional objects can have intricate actuation, self-assembly, and sensing capabilities with little to no manual intervention. To achieve this, we modify a low-cost Fused Filament Fabrication (FFF) 3D printer, adding a stepper motor-controlled fiber spool mechanism on a gear ring above the print bed. In addition to hardware, we provide parsing software for precise fiber placement, which generates G-code for printer operation. To illustrate the versatility of our system, we present applications that showcase its extensive design potential. Additionally, we offer comprehensive documentation and open designs, empowering others to replicate our system and explore its possibilities.

CCS CONCEPTS

• **Human-centered computing → Interactive systems and tools**.

KEYWORDS

fabrication, additive manufacturing, fiber, multi-material, G-code

1 INTRODUCTION

Personal fabrication technologies such as 3D printers offer the possibility to create entirely custom objects on demand. While 3D printing has not yet reached the ease of use of inkjet printer technology, it is not unusual to find 3D printers in schools, offices, and private homes. Despite the technology's accessibility, it remains difficult to fabricate objects with functionality beyond their form. Fused-filament fabrication (FFF) printers, today's popular low-cost machines, print by melting and extruding thermoplastic filament, meaning the possible functionalities of the resulting objects are limited to those that can be implemented with thermoplastic.

To enable extra functionality, composite thermoplastics are available with properties such as conductivity [67], magnetism [68], or stretchability [31]. Such composite materials have the advantage that they can be directly used with typical FFF-printer extruders; however, they are also limited by the necessity of being compatible with the FFF melt-and-extrude process.

To overcome this limitation, researchers have begun exploring the possibilities of incorporating *non-extrudable* materials into printed objects. In contrast to embedding discrete *components* [6] such as bolts, magnets, or microcontrollers, these efforts augment the printed plastic with a second homogeneous material such as fabric [64], wire [56], or magnetophoretic liquid [78]. To date, however,

these efforts have been limited by a requirement that the user manually embed the material, the necessity for a heavily modified or entirely custom 3D printer, or support of only a single non-printable material type.

In this paper, we present Rhapso, a system for automatically incorporating non-extrudable fiber-like materials into objects during the 3D printing process. Such materials, while affordable, accessible, and easy to work with, offer an incredibly wide variety of useful properties. Our prototype system works with fibers which possess useful mechanical and aesthetic characteristics: typical threads, strings, and yarns are extremely durable and flexible and come in many materials and colors; elastic fibers can store energy and resist change ; and synthetic fibers like nylon fishing line have high tensile strength. Rhapso enables such materials to be laid out inside 3D printed objects, imparting new functionalities via the fiber's properties and its interaction with the printed structure.

To begin to uncover the potential of embedding fiber in printed objects, we created a design space. We first survey existing aesthetic, functional, and material properties of fibers that exist pre-print and could be of interest for interaction. We describe during-print requirements for attaching printed structures and fibers together or leaving the fiber free, including the need to control position, tension, fixation, and arrangement of the fiber. Finally, we discuss the concept of thread as a force-bearing substrate within printed objects for interactive purposes.

To enable exploration of the design space, we present Rhapso, a 3D-printing system that embeds a continuous fiber into objects as they are printed. Rhapso—named for a Greek goddess associated with organizing the thread of a person's life—consists of a high-level architecture and abstract algorithms for embedding a fiber in a user-specified path through the printed object, as well as two concrete instantiations suitable for consumer-level 3D printers. The first, which automatically embeds fiber, is based on simple modifications to a low-cost 3D printer. We add a geared ring parallel to the print bed and level with the currently printing layer. Fiber stretches in a straight line from a ring-mounted spool to an anchor point on the bed or object. To embed the fiber, the ring rotates to position it across a new anchor point, where the extruder prints over and affixes it. This process can repeat multiple times in a single layer and over multiple layers.

The second implementation requires no printer modification; instead, the user takes the place of the ring to reposition the fiber when prompted, allowing for quick and easy experimentation. The systems use identical algorithms to print the object and route fiber on the user-specified path.

We show the interaction potential of 3D printing with fiber via ten examples illustrating a subset of the design space. We also provide open-source code and 3D design files to enable others to replicate our work.

To summarize, the main contributions of our work are:

- a design space describing the interactive potential of fibers in 3D prints;
- a high-level architecture and abstract algorithms for embedding fiber in a user-specified path through a 3D-printed object;

- an open-source hardware and software system for low-cost 3D printers that enables embedding different types of fiber-like materials; and
- example applications demonstrating the interaction potential of combining fibers with 3D printing.

2 RELATED WORK

Researchers have expanded the functionalities of 3D printed objects by exploring new printing techniques and materials as well as adding functional materials during and after printing process. These functional materials include fiber-like materials, though the design space has not in general been explored nor has a generic automated solution been proposed.

2.1 Fiber-like materials in 3D printing

Fiber-like materials (variously referred to in related work as "line materials", "fibers", "fiber", and "fibrous materials"), as noted, offer a wide variety of desirable characteristics, and Rhapso is far from the first to capitalise on them. Shape-change is a popular application [50], with explorations in using shape-memory alloys (SMAs) [48, 54], and tensionable cables [80]. Variable haptics is another area with multiple explorations, seeing fibers used to create texture [26] and modulate stiffness [53], or generate heat [10]. Naturally, sensing is a third explored area, with conductive fiber being used to connect modules [12, 28], measure stretching [76], and create inductors and similar sensors [17, 58]. PunchPrint designs printed objects for later fiber embedding and EscapeLoom for later weaving, with the goal of supporting crafters [7, 8]. Non-fibrous metal wires have also been explored as components that can be wrapped to create electromagnets [56] or enable poseable objects through manual wire threading [72]. In terms of characterization, the effects of fiber inclusions in 3D printed objects have been measured with respect to their mechanical [2, 61], electrical [58], and thermal [4] properties. These works each explore a single type of fiber-like material, and for the most part do not offer automated solutions. Our work unites their explorations.

2.1.1 Automated solutions for embedding fibers. Automated solutions to embed fiber-like materials have been explored in different 3D printing systems in multiple contexts. For example, some embed wires [11, 25, 30] as electronic components, or carbon fiber [3, 27] as a structural or electrical component, or for direct strength reinforcement [42]. Others embed various materials in printed concrete [37, 44] as a reinforcement method, or develop custom filaments (or filament-creating machines) with fiber-like cores for conductivity [81] or structural stability [60, 79]. While these solutions all include machines capable of depositing the needed fibers alongside other geometry, they each focus on a specific fiber/application context and lack flexibility.

Stalin et al. propose an automated device that can embed fiber-like material into 3D models created with a silicone printer, and explore their influence on shape-change rigidity and local heating, as well as their use as electronics [69, 70]. Sun et al. propose a similar, open source and low-cost solution for embedding fiber by pushing it into hydrogel as it is printed [73]. These works are conceptually similar to our work, but instead of a special-purpose silicone printer or hydrogel printer, we focus on a system for modifying low-cost, generic FFF printers, which present novel challenges, such as not being able to push fibers as with the hydrogel implementation. Stiltner et al.'s method for including fibers as joint actuators in polyjetted models [71] similarly explores possibilities and demonstrates deformable interactive devices, but requires manual fiber embedding during layer pauses and an expensive machine.

2.2 Fabrics and 3D printing

Fabric, a dense mesh of fibers, is also of interest to HCI and other researchers as a desirable embedded material thanks to its flexible properties that complement hard thermoplastics. Rivera et al. [65] and FabriClick [18] embed fabric into 3D printed objects to leverage its rich interactive capabilities, and CurveUps [20] and Koch et al. [32] embed pre-stretched fabric with precise printed components to enact shapechange. Fiber offers fundamentally different opportunities from full fabric sheets; instead of being a plane, it is a line that can be woven through space in varying configurations. Rivera et al. also explored including individual fibers as actuators for their fabric-embedded objects (similar to our robot finger example); Rhapso expands the opportunities of fiber-like material embedding, and automates the bespoke, manual embedding process used for that example object. Others have built custom machines that 3D print out of cut-and-layered fabric sheets [57], felted yarn strands [24], spooled yarn or fiber [35], or hydrogel added into existing fabric materials [62], thus creating fully-soft objects in a 3D printer-like manner. These last works create unique, soft objects, but cannot take advantage of the blending of hard and soft materials that Rhapso explores. Of note, Rivera et al. have built a machine that can blend hard and soft materials by using electrospinning to deposit textiles into designed objects [63]; Rhapso instead explores an adjacent design space with the deposition of full fibers, and could potentially be combined with the electrospinning implementation.

Others examine not just how to integrate fabric into 3D printing processes, but rather how to print fabric-like materials directly from common thermoplastics. Commercially, this was most famously explored by n-e-r-v-o-u-s systems in their Kinematics project [51], which created small, rigid, linked units that flowed like a textile. In research, DefeXtiles uses a single printed material (PLA) to create both flexible and rigid components by fabricating a novel metamaterial structure [13], while Takahashi et al. [74] and Li et al. [36] explore weaving-like techniques.

2.3 Other uses of fabrics and fibers in HCI

Many other explorations of fibers and fabrics in HCI go beyond embedding them in 3D prints, including developing novel fabric– or fiber-like materials and structures that behave like actuators [14, 29, 49], sensors [39, 59], or both [1, 40, 45, 52]. Compilers and design software for knitting and textiles is also a popular area of research [1, 15, 43]. A future version of Rhapso could integrate some of these novel fiber-like materials, but our exploration is largely orthogonal to this body of work.

2.4 Multimaterial processes and non-printable materials

Multimaterial printing processes and processes that allow integrating several machines into the manufacture of a single object [19, 41]
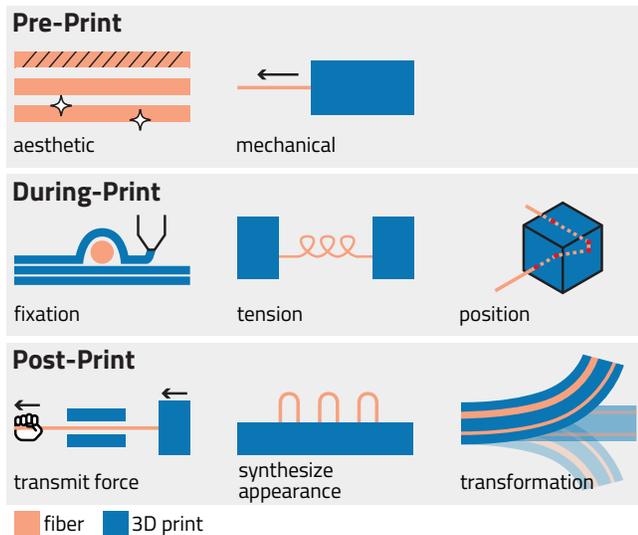
Pre-Print

aesthetic

mechanical

During-Print

fixation

tension

position

Post-Print

transmit force

synthesize appearance

transformation

fiber ▮ 3D print

Figure 2: The design space of working with fiber embedded in printed objects.

offer many opportunities for embedding interactivity. But the fact remains that, while 3D printers and related techniques offer many different material possibilities, there are still many classes of material and objects which cannot be printed—or even "faked" with clever metamaterial designs—by consumer-grade machines. Thus, others have explored embedding non-printable materials during and after 3D printing to achieve interactivity goals. Aside from fabric, discussed above, these investigations in HCI examine embedded electronics like IMUs [23], cameras [66], or as integrated bricks [16, 34]. More generally, everyday objects can be embedded into prints to enhance their capabilities as in Medley [5], or just to make printing faster [77]. Rhapso focuses on fiber-like materials, but we take inspiration from the use cases described in these works and the general concept of adopting existing materials' properties for use in 3D printed objects.

3 DESIGN SPACE OF PRINTING WITH FIBER

Combining non-printable materials with fabricated objects can add desirable characteristics not possible to achieve with printable material alone; Rhapso accomplishes this by interleaving fibers into 3D printed objects. Here, we loosely define *fibers* as continuous fiber-like materials such as, for example, string, wire, plastic cable, etc. To understand and organize the opportunities for fiber embedding, we developed a design space (Figure 2).

As a practical matter, our design space assumes an extrusion-based thermoplastic printing process—i.e., FFF—as this is the printer type we use for our implementation (Section 4). We organize the design space by the three stages of the fabrication process: pre-print *(fiber properties)*, during-print *(fiber/print assembly actions)*, and post-print *(force transmission)*. Design choices made in each stage have cascading effects on later stages; e.g., choice of fiber material in the first stage can impact adhesion between fiber and printed object during the second.

3.1 Pre-Print: fiber properties

The quantity of available fiber-like materials is vast, with thousands of general- and special-purpose fibers, cables, and wires available, and with properties as varied as the fibers themselves. We briefly survey aspects of fiber properties that we explore in this work, and discuss some of their potential uses for interaction.

3.1.1 Aesthetic properties. The inclusion of non-thermoplastic materials offers an opportunity to vary the aesthetics of 3D-printed objects in both texture and color. Common sewing thread is available in hundreds of colors, with textures ranging from smooth to hairy. While unexplored in this paper, metallic materials such as wire, or textile fiber that incorporates metal—such as conductive fiber—can be shiny.

3.1.2 Mechanical Properties. The varying mechanical properties of fibers—including thickness, flexibility, strength, elasticity, slipperiness/friction, and more—suggest various interactive uses. In combination with 3D-printed structures that affix, tension, or constrain fiber relative to the printed object, they can give objects a wide range of capabilities. Inelastic fibers can enable cable-drive actuation [13, 64] or constrain the shapes of inflatable or flexible objects [69], while embedding strong fibers could strengthen prints [79]. Many fibers—e.g., textile-based—are extremely flexible and can withstand an essentially unlimited amount of bending [64].

3.2 During-print: fiber/print assembly actions

To leverage the properties and opportunities of fibers, they must physically interface with the fabricated object. We focus on the design space with respect to FFF 3D printers, discussing fixation of fiber and printed object, tension between them, 3D placement, and fiber topology, or how the fiber is arranged.

3.2.1 Fixation. The fundamental interaction between fiber and printed structure is fixation, or how they are attached. A fiber can be permanently or temporarily affixed, and fixation points can occur at one or multiple locations along the fiber's length and one or more locations within the printed object.

The method of fixation depends on the interaction between the fiber's properties (Section 3.1) and those of the printed material. Such interactions are complex and difficult to predict [21], but experimentally we have observed several potential fixation mechanisms. For some fiber types, *adhesion* is possible, where the thermoplastic seeps into the fiber material (as with loose or "hairy" cotton fiber) or partially melts the fiber (as with TPU fiber), forming a strong bond. When adhesion is poor, multiple lines of plastic laid down perpendicular to the fiber can *clamp* it into place between the object's layers. Some fibers (like nylon fishing line) will not adhere at all; in such cases, *winding* the fiber around a printed internal post followed by clamping can be effective. Printed hooks and clips can also be used, for less-permanent fixation.

3.2.2 Tension. Closely related to the attachment between a fiber and an object is the *tension* of the fiber between attachment points, which can control the post-print behavior of the object. Tension influences other mechanical properties: a fiber under tension can be strummed or more easily cut or snapped, while a loose fiber droops. With elastic fiber materials, a fiber can be extended under

tension beyond its natural length, and its tendency to contract can be used post-print to implement, for example, self-assembly or shape-change.

3.2.3 Position. For full flexibility in adding fiber to printed objects, we must be able to position the fiber at various points in 3D space. This includes *in-plane* and *out-of-plane* positioning. Thanks to the planar nature of the typical FFF 3D printing process, in-plane direction changes can be achieved by printing guiding or affixing paths or points in an object, with the fiber strung through or between them. The fiber can then be sealed into position in the layer.

Out-of-plane fiber pathing can be more difficult for machines of this design. In the easy case, a fiber affixed at layer N_i can be brought upwards for further fixation in layers $N_{i+1}, N_{i+2}, \ldots$. The other direction is more difficult: it requires special design care to attach to layer N_{i-j}. This is conceptually related to either tension (e.g., drooping the thread to send it downwards), wrapping (e.g., using the object itself as a guidepost [35]), or having print-head access to a specific location despite existing structure (e.g., to print thread and anchors in an area of shallow slope [47]).

3.3 Post-print: force transmission and interactions

After printing, an embedded fiber can interact with physical forces to add functionality to the object, depending on the fiber's properties and how it is incorporated in the object. Some types of fibers can *originate* forces. For example, pre-tensioned elastic fibers can store force to enable shape-change (Section 6.1), self-assembly (Section 6.4), or bistability (Section 6.5). Fibers can *transmit* forces from one point in an object to another; for example, a fiber pulled by a user carries that force through the object, including around curves, to where the fiber is fixed (Section 6.7). In conjunction with printed geometry, fibers can also *transform* forces; for example a linear pull can become a rotational curl (Section 6.6).

In addition to manipulating mechanical forces, fibers can also influence the appearance and tactile qualities of objects. Threads and yarns are available in a nearly unlimited variety of colors and textures; by strategically embedding such material at the surface of an object, different visual and tactile effects can be achieved (Section 6.8).

4 RHAPSO

In order to explore the design space of printing with embedded fibers, we created Rhapso. Rhapso's goal is to automatically embed a single continuous fiber into 3D-printed objects during FFF printing, a process which presents multiple challenges.

Fiber cannot be dispensed in the same way as thermoplastic filament. While 3D printers typically grip and push filament with motorized rollers, using this approach with fibers causes tangling and clumping. Additionally, most fiber material is incompatible with the FFF printing process, which relies on the ability of thermoplastic filament to melt, adhere to previously-extruded filament, and then quickly solidify.

These challenges require a non-extrusion-based system for fiber printing. To instantiate the during-print fiber/object assembly actions enumerated in the design space (fixation, tension, and topology; see Section 3.2), a fiber-embedding printer must support three primitive operations:

(1) *Position fiber:* The printer must be able to place the fiber in the desired 3D location and orientation with respect to the object.
(2) *Avoid fiber:* To allow the fiber to freely move in some location, or to print parts of the model not involving fiber, the printer must be able to extrude plastic without intersecting the fiber, to avoid fixing the fiber in place.
(3) *Print over fiber:* To attach the fiber to the model in order to gain the desired functionality, or to allow the fiber to make directional changes within the model, the printer must be able to extrude plastic on top of the fiber to fix it in place.

In the following sections, we present a high-level overview of Rhapso's system architecture and how it uses these primitive operations to solve the challenges of incorporating fiber into the FFF printing process. We then detail two implementations that instantiate Rhapso's architecture in low-cost consumer-level printers.

4.1 Walkthrough

To illustrate the process of designing and printing with Rhapso, consider the robot finger illustrated in Figure 3 and Figure 22. It consists of three 3D-modeled parts with an embedded elastic fiber acting as an actuating tendon. When the loose end of the fiber is pulled, it transmits this force to the end of the finger, causing it to curl up (Figure 22b). The bottom segment of elastic fiber resists this motion by stretching, such that when the first segment is released, the finger opens again.

Creating this object is a four-step process; in brief:

(1) the 3D model is annotated to indicate the fiber path and modified to add features where the fiber will be affixed or slide through;
(2) the 3D model is sliced using standard slicing software;
(3) the Rhapso software modifies the sliced model to embed instructions for the Rhapso-enabled printer to correctly print the model; and
(4) the printer is loaded with elastic fiber and is used to print the object, automatically embedding the fiber during the print process.

To design the mechanism, the finger is first modeled in Autodesk Fusion. The path the fiber will follow is added to the model as a single, continuous 3D sketch line. Channels are required to allow the top segment of the fiber to be pulled for curling, and to allow room for the bottom segment to stretch for opening. To keep the pulling tension for the opening behavior, two "half-pipe" anchoring structures (Section 5.4.4) are added to anchor the segment at the two ends of the bottom channel. To divide the fiber into two independent segments, another half-pipe is added to the top channel at the end of the finger. Based on the 0.6 mm fiber diameter, 0.8 mm channels are added to the finger model (Figure 3(A)).

Next, this modified finger model is loaded into standard software such as Ultimaker Cura and sliced according to typical printing
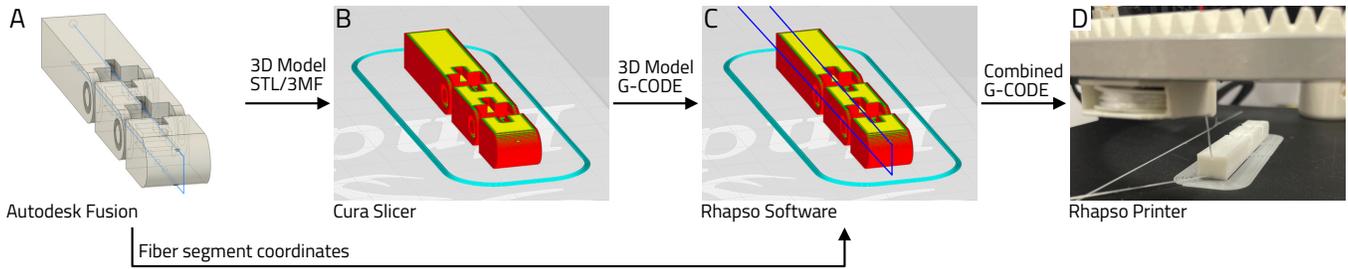
Figure 3: Walkthrough of fabricating robot finger in 6.6. (A) Users draw a 3D (blue) line that will be embedded in the 3D model and export the line and 3D model separately. (B) The 3D model is sliced and converted to G-code. (C) Our custom software generates G-code for line embedding and combines it with 3D model G-code. Note that the image is conceptual; our software visualizes the process on a 2D plane as in Figure 4(a). (D) Our modified 3D printer embeds an elastic fiber in the finger as designed.

parameters for the base printer to be used, producing an output G-code file (Figure 3(B)).

In order to embed fiber during printing, the G-code from the slicer is modified to control the Rhapso hardware. Using a custom script, the coordinates of the fiber path are exported from Fusion and, along with the sliced G-code, provided to the Rhapso software. The software optimizes the print order of the object to accommodate the fiber (Section 4.2.2), includes control commands for the fiber-manipulating mechanism (Section 5.2.3), and adds extra G-code commands to securely fix the elastic fiber (Section 5.4.4). The Rhapso software produces a new G-code file as a result of this process (Figure 3(C)).

Finally, to print the finger model, a spool of elastic fiber is loaded onto the printer's fiber carrier, and one end of the fiber is anchored to the print bed (Section 5.4.4). The printer, loaded with the Rhapso-generated G-code, then produces the finger with no manual intervention (Figure 3(D)).

4.2 Abstract system architecture

To facilitate various future concrete implementations of the Rhapso concept, we first present an abstract overview of basic requirements, representations, and processes involved in embedding fiber into a model during the print process, along with resultant limitations.

4.2.1 Requirements and representations. Our abstract architecture for Rhapso makes a few assumptions about the physical implementation. First, we assume the printing device works via the typical FFF process of creating a 3D object by building it one 2D layer at a time, with each layer containing a set of lines of thermoplastic extruded by a print head that moves in 2D within the layer. Second, we assume a fiber delivery mechanism, or carrier, that can maintain the fiber within the plane of the currently printing layer, in a straight line between the carrier and where the fiber is anchored. For maximum flexibility, we assume the carrier is located outside of the print volume.

To implement the three primitive operations *position*, *avoid*, and *print over*, Rhapso needs to know about the object to be printed, and how the fiber should be incorporated into the object.

In the abstract, Rhapso represents each object layer as an ordered list of 2D line segments, each signifying a move of the print head that extrudes plastic.

The path of the fiber within the object is represented as an ordered list of 3D points, or *anchors*, at which the fiber should be fixed inside the completed object. The arrangement of fiber is subject to the constraint that the print head cannot pass through already-printed material, and thus once a layer is complete and the printer has moved to the next-higher layer, it is not usually possible to return to a previous layer to lay down further plastic. This limitation affects how fibers can be routed through an object: they can be arranged in many ways within a layer, and can transition from a completed layer to a higher one, but they cannot return to a layer printed earlier; thus, anchor points must monotonically increase in the **Z**-axis.

4.2.2 Basic routing. Given object and fiber geometry, Rhapso determines how to route fiber through the object. This operation involves *positioning* the fiber to either be *printed over* by or *avoid* lines of extrusion. Because FFF 3D printers sequentially extrude one line at a time, the routing process involves determining how, when, and where to position the fiber relative to the printed lines.

The routing algorithm, sketched in Algorithm 1, works on one layer of the to-be-printed object at time. As input, it takes two parameters: Layer, an ordered list of 2D line segments comprising the printed geometry of that layer, and Anchors2D, an ordered list of 2D points within the layer through which the fiber should pass. Except for the first anchor in the list—the location at which the fiber is currently fixed, and which may be on the print bed—all points within Anchors2D must lie on at least one of the segments from Layer. This requirement ensures that the fiber path within the printed object matches the path specified by Anchors2D.

The algorithm produces an ordered list of operations Ops, alternating between *fiber rotations* and *print operations*. A *fiber rotation* is a rotation of the fiber around an anchor point, and a *print operation* is a group of one or more line segments representing extruded plastic.

In some cases, the path of the fiber between the current anchor point and the carrier may intersect geometry that should *not* fix the fiber in place. To print these segments, the *avoid* operation is used

to prevent undesired attachment, by rotating the fiber to prevent its being fixed.

Executing the operations Ops on a printer produces the input Layer with the fiber embedded in it according to the locations in Anchors2D.

4.2.3 Pre-routing setup. The basic routing algorithm given above describes the rough procedure, but is missing a number of important details. Those specific to a particular implementation will be discussed in Section 5; here we briefly sketch several implementation-independent procedures which work to prepare a 3D model and fiber path for routing. The 3D model consists of an ordered list of layers in 3D Layers3D = [Layer$_0$, Layer$_1$, ..., Layer$_N$], and the fiber path is specified as an ordered list of anchor points Anchors3D in 3D coordinates.

Multiple layers: While the basic algorithm detailed in Section 4.2.2 routes fiber in 2D, we want to allow fiber to routed through the object in all three dimensions. We want to transform Anchors3D into a list [Anchors2D$_i$, Anchors2D$_{i+1}$, ..., Anchors2D$_M$], where $i ... M$ correspond to the layers through which fiber passes. First, we ensure all anchors are *on* a layer, not between layers, by "snapping" their Z-coordinate to the nearest layer boundary (Figure 5b).

It may be the case that multiple layers separate two subsequent anchors, so that some layers have fiber passing through them. RouteLayer must still be used on these layers in order to prevent the fiber from interfering with the printing process. We thus generate new anchors for these cases by finding every length of fiber between two anchors not on the same layer, and creating new anchors where the fiber intersects the intervening layers (Figure 5c).

Anchor/layer alignment: RouteLayer requires that the points in Anchors2D be aligned to segments of printed geometry represented in Layer. Without this requirement, an anchor not located on a printed segment will effectively "move" to a nearby intersection of fiber with printed plastic, a potentially large distance from the desired anchor point. Rhapso thus "snaps" each anchor to a point on a printed segment that is closest to that anchor point.

Splitting geometry: After adjusting anchor points, it may be the case that a longer segment of printed geometry in Layer contains more than one anchor point. If those anchors are non-sequential, then printing that segment to fix the fiber at the first anchor will prevent fixing at the second anchor. To avoid this problem, we split such segments into parts between the anchors.

4.2.4 Complete routing procedure. For clarity, we have described the processes involved in routing fiber through a model out-of-order. From start to end, the full implementation-independent procedure is as follows:

(1) Input is a 3D model as Layers3D and a fiber path as an ordered list of anchors Anchors3D, where the first anchor A_0 is fixed outside the model (i.e., to the print bed).
(2) Snap each anchor in Anchors3D to a layer boundary.
(3) For each layer through which fiber would pass without a defined anchor, insert a new anchor at the layer boundary.
(4) Snap the anchors on each layer to the nearest printed geometry.
(5) Split printed segments containing multiple anchors.

Algorithm 1 RouteLayer(Layer, Anchors2D)

Input: Layer, a list of 2D line segments representing the geometry that will be printed for a given layer

Anchors2D, an ordered list of points through with the fiber should pass, where each fixation point $A_i \in$ Anchors2D lies on at least one segment of Layer. The fiber should be fixed at A_0.

Output: Ops, an ordered list of fiber moves and line segments to print.

1 Mark all segments in Layer as UNPRINTED
2 Ops ← [ ]   #Initialize Ops to empty

3 Rotate(A_O, A_T)→
4   Append to Ops a command to rotate the fiber about A_O so it crosses A_T

5 Print(G)→
6   Add commands to print segments in G to Ops
7   Mark all segments in G as PRINTED

8 **for each** A_i in Anchors2D:
9   #Fiber is currently fixed at A_i ●;   Fig.4b
       #   now rotate it to cross over the
       #   next anchor A_{i+1} ◎
10   Rotate(A_i, A_{i+1})   Fig.4c
11   G_{fix} ← UNPRINTED segments in Layer
           which intersect A_{i+1}
12   #Print over fiber to fix it at A_{i+1} ◎
13   Print(G_{fix})   Fig.4d

14   #Find geometry that will print over fiber between
       #   previous (A_i ○) and current (A_{i+1} ●)
       #   anchor points; we will print that next.
15     $\overline{S_i}$ ← line segment between A_i and A_{i+1}
16   G_{isec} ← segments from Layer intersecting $\overline{S_i}$

17   #But don't print geometry that will print over
       #   future fiber segments
18   **for each** (A_{i+1}, A_{i+2}) in Anchors2D:
19         $\overline{S_{i+1}}$ ← line segment between A_{i+1} and A_{i+2}
20   G_{future} ← segments from G_{isec} intersecting
               segment $\overline{S_{i+1}}$
21   Remove G_{future} from G_{isec}

22   #Now print segments that cross only $\overline{S_i}$, not any
       #   future fiber segments
23   Print(G_{isec})   Fig.4e

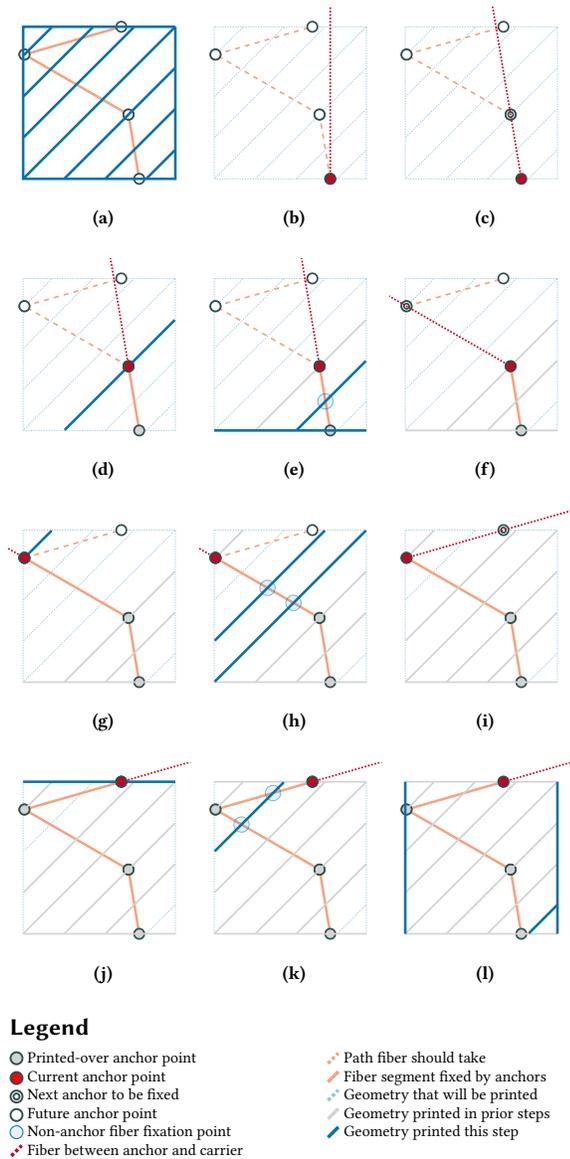24 Print(all UNPRINTED segments in Layer)   Fig.4l

(a)   **(b)**   **(c)**

Figure 5: Layer/anchor snapping process. (a) Cross-sectional view of input anchor points ○ and fiber path ╱, against layers ▨ (layer thickness exaggerated). (b) Anchor points snapped to the nearest layer boundary. (c) New anchor points added where fiber crosses layers without anchor points defined.

(6) Execute RouteLayer(Layer$_i$, Anchors2D$_i$) for each Layer$_i$ that contains anchors Anchors2D$_i$.

5 IMPLEMENTATIONS

Having described the general theory of Rhapso's operation, we now detail two instantiations of the model. Our first implementation, which automatically embeds fiber, consists of a physical printer (Section 5.1) which enables *positioning*. The second implementation (Section 5.3) uses a guided, manual approach to work on an unmodified printer, with the user taking the place of the automated embedding hardware. Both share software based on the Rhapso algorithms (Section 5.2) to implement *avoiding* and *printing over*. The hardware and software are open-source and available at https://github.com/fetlab/rhapso.

5.1 Hardware

We implemented Rhapso by modifying a Creality Ender-3 Pro, a low-cost, small-size, consumer-grade 3D printer. The printer is configured in a typical Cartesian setup where the build plate (220 × 220 mm) moves along the Y axis and the printer head moves within the XZ planes (maximum Z of 250 mm). We replaced the stock motherboard in the printer with a BIGTREETECH SKR V1.4 Turbo control board, which supports five independent motors in contrast to the stock Ender-3's four.

To enable the *position fiber* primitive operation, we mount a fiber carrier to the bottom of a two-part movable ring[1] attached to the printer (Figure 6). The outer portion of the ring is fixed to the X axis, while the inner part of the ring can rotate freely through 360°. A 3D-printed gear mounted to the top of the inner ring meshes with a gear attached to a motor, mapped in firmware (Marlin 2.1.0) to a rotational fourth axis **A**, allowing the printer to control the fiber carrier's position by rotating the ring.

The carrier itself contains a spool of fiber with a spring-based tensioning mechanism. To keep the fiber in the plane of the layer being printed, it exits the carrier level with the opening of the printer's nozzle. In our current design, the spool and tensioning spring are adapted from a standard retractable lanyard; this spool can hold about 10 m of .25 mm-diameter sewing fiber, or about 2 m of .7 mm-diameter elastic fiber, although our current design is constrained by the maximum extension of the spring to about

---

[1]An aluminum turntable bearing, or "Lazy Susan"

(a)   **(b)**   **(c)**

(d)   **(e)**   **(f)**

(g)   **(h)**   **(i)**

(j)   **(k)**   **(l)**

Legend

○ Printed-over anchor point   ⋰ Path fiber should take
● Current anchor point   ╱ Fiber segment fixed by anchors
◉ Next anchor to be fixed   ⋰ Geometry that will be printed
○ Future anchor point   ╱ Geometry printed in prior steps
○ Non-anchor fiber fixation point   ╱ Geometry printed this step
⋰ Fiber between anchor and carrier

Figure 4: A step-by-step example of Rhapso's routing algorithm operating on a single layer of an object. (a) A preview of the finished layer. (b) The initial state of the layer and fiber. (c) The fiber rotates to cross the next anchor point. (d) The fiber is anchored via a *print over* operation. (e) Geometry which does not interfere with the fiber's upcoming paths is printed, including further fixations. (f) The fiber rotates to cross the next anchor point. (g) The fiber is anchored via a *print over* operation. (h) Geometry which does not interfere with the fiber's upcoming paths is printed, including further fixations. Note the line of geometry closest to the anchor point which, if printed at this stage, would prevent the fiber being fixed by that line. (i) The fiber rotates to cross the next anchor point. (j) The fiber is anchored via a *print over* operation. (k) Now the line of geometry can be printed over both segments of fiber it crosses. (l) All remaining unprinted geometry is printed.
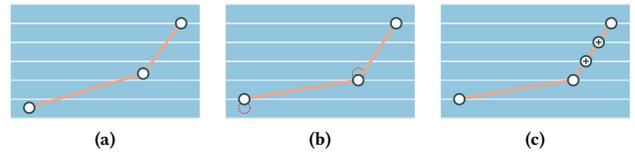
65 cm. In addition to this spring-based spooling mechanism, we also designed an active spool tensioning system, illustrated in Figure 26.

Rhapso's fiber-routing algorithm (Section 4.2.2) assumes the fiber forms a straight line between the current anchor point and the fiber carrier. To form the first anchor, we attach the fiber to the print bed via a clip (Section 5.4.4). Further anchors are created during the printing process via the *print over* operation. By controlling the positions of the ring (**A** axis) and the bed (**Y** axis), the fiber can be made to cross over any point within the printer's build area.

The ring is mounted on the printer so that its diameter (197 mm) along the **X** axis is aligned to allow the print head (57 mm wide) the maximum amount of travel; even so, the usable travel of the print head in **X** is reduced from 220 mm to 110 mm (we must avoid collision of the thread carrier with the print head as well as the belt fixation points on its underside). The **Y** and **Z** axes maintain their full 220/250 mm travel, unaffected by the added hardware.
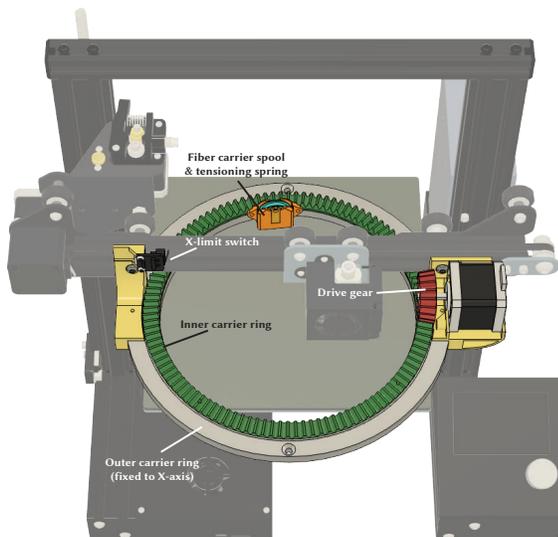

Figure 6: Detail view of Rhapso implementation.

5.2 Software

Using Rhapso to print an object with embedded fiber involves several steps:

(1) **Design:** The user designs the object to be printed, and specifies anchor points where the fiber will be affixed to the object.
(2) **Slicing:** As is standard for FFF printing, the 3D model must be sliced and transformed into G-code instructions for the printer.
(3) **Adding fiber-embedding instructions:** Rhapso's routing implementation post-processes the G-code output from the slicer, adding ring-movement instructions to position the fiber in order to produce the desired output.
(4) **Printing:** The printer uses the processed G-code to simultaneously print the model and embed the fiber.

5.2.1 Design. Rhapso's routing algorithm (Section 4.2.2) requires a list of 3D model-space coordinates indicating where fiber anchor points should be located relative to printed geometry and in what order the fiber should pass through them. We use Autodesk Fusion to specify anchor points. The user designs or imports a model as usual, then models the fiber path as a series of connected line segments in 3D. Rhapso uses the Fusion API to export the path vertices, and the user exports the model itself via the standard Fusion interface. Currently, we do not detect errors or enforce design constraints such as requiring the fiber path to increase monotonically along the **Z** axis. Figure 7 shows a model and fiber path in Fusion.
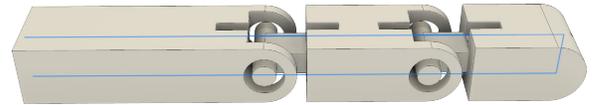

Figure 7: An example of a Rhapso design in Autodesk Fusion, with the fiber path drawn as a 3D sketch line (blue).

5.2.2 Slicing. In FFF 3D printing, slicing is the process of transforming the mathematically defined 3D model into instructions for the 3D printer to execute [38]. We conducted our experiments using Ultimaker Cura[2] version 5.3, but Rhapso is not dependent on a particular slicer.

Slicers enable end-user manipulation of many printing parameters which can affect print speed and quality. Several of these parameters are important to adjust with the Rhapso fiber-embedding process in mind:

- **Bed size:** The slicer needs to know how the bed size is restricted by the fiber-carrier ring to avoid collisions between the print head and carrier.
- **Printing temperature:** Recommended printing temperatures vary dramatically for different filament types and manufacturers; suggested temperatures for polylactic acid filament (PLA) range from 195–230° C, while polycaprolactone (PCL) filaments can be printed at temperatures as low as 60° C. Designers must consider compatability of the chosen fiber material with the print temperature; for example, low-cost optical fibers are typically made of acrylic with a melting point of 70° C.
- **Print speed:** Higher print speeds are desirable as they shorten print time; however, depending on the strength of the fiber and the force exerted by the tensioning mechanism, high acceleration of the print bed for **Y** axis moves could cause the fiber to snap.
- **Infill type and density:** Infill settings influence print speed, amount of material used, and object strength. When a fiber path is designed to pass through the interior of an object, the routing algorithm snaps anchor points to the nearest printed geometry. The maximum possible distance an anchor could be moved is influenced by the type and density of infill.

Once slicing is completed, the generated G-code is then passed to the Rhapso routing program, which post-processes it to generate instructions for embedding the fiber.

---

[2]https://ultimaker.com/software/ultimaker-cura/

5.2.3 Fiber-routing implementation. Our implementation of the Rhapso routing procedure (Section 4.2.2) accepts a G-code file and a fiber path as input, modifies the G-code according to the routing algorithm, and outputs a new G-code file that can be used to print the fiber-embedded model without manual intervention. While our software follows the algorithms laid out in Section 4.2.2, there are a number of implementation-specific details which must be accounted for.

To make calculations of how to move the carrier ring, our system needs to know details of the physical hardware, including the location of the ring relative to the printer gantry, the ring size, and the location of the initial anchor point on the bed.

We transform the G-code output from the slicer into a list of line segments in 3D, allowing us to more easily perform operations such as finding intersections between fiber and print geometry. This transformation also enables manipulating print order: slicing software optimizes in-layer print order for factors like speed or surface appearance; in some cases, we must disrupt this ordering to enable proper fiber routing.

To illustrate why this reordering can be necessary, Figure 8b is a reproduction of Figure 4h with annotations added. The lines labeled 1–3 are printed in sequence by the original G-code from the slicer (Figure 8a). If line 3 is printed, it will fix the fiber at location **P**, but will also cross location **Q**, where the fiber has not yet moved. Thus, we first rotate and fix the fiber before printing line 3 (Figure 8d), which necessitates reordering the G-code (Figure 8c).

Because the center of rotation for each fiber segment will typically be different than the ring's center, additional calculation is required to determine what ring rotation will yield the desired fiber rotation. We find where a ray from the current anchor to the next anchor intersects with the ring, then determine the angle of the intersection relative to the ring center.

5.2.4 Printing. When all layers have been processed, Rhapso generates the final G-code. This step is dependent on the specific printer setup. In particular, for printers with an XZ-head configuration (like our Ender-3-based system, see Section 5.1), where **Y** axis movement occurs via a moving bed, each line of output G-code is augmented with a ring-movement instruction that keeps the fiber angle constant as the bed moves. For every line containing a **Y** move, we calculate the amount the ring must rotate to keep the fiber at the same angle (Figure 9), and modify the G-code to add a corresponding **A** axis rotation.

5.3 Manual printing implementation

Our second, alternate implementation of Rhapso uses the same concepts as the first, but does not require adding a carrier ring to the printer. Instead, we take a *hybrid-fabrication* approach [9], where the user takes the place of the ring, moving and tensioning the fiber when prompted, in contrast to the primary fully automated implementation presented in Section 5.1.

While this fully manual approach lacks the benefits of automation, it offers wider accessibility of Rhapso, as no hardware beyond a FFF 3D printer is required. Leaving fiber manipulation in human hands also provides opportunities to work with fibers that cannot easily be mounted on the carrier spool due to thickness, bend radius,
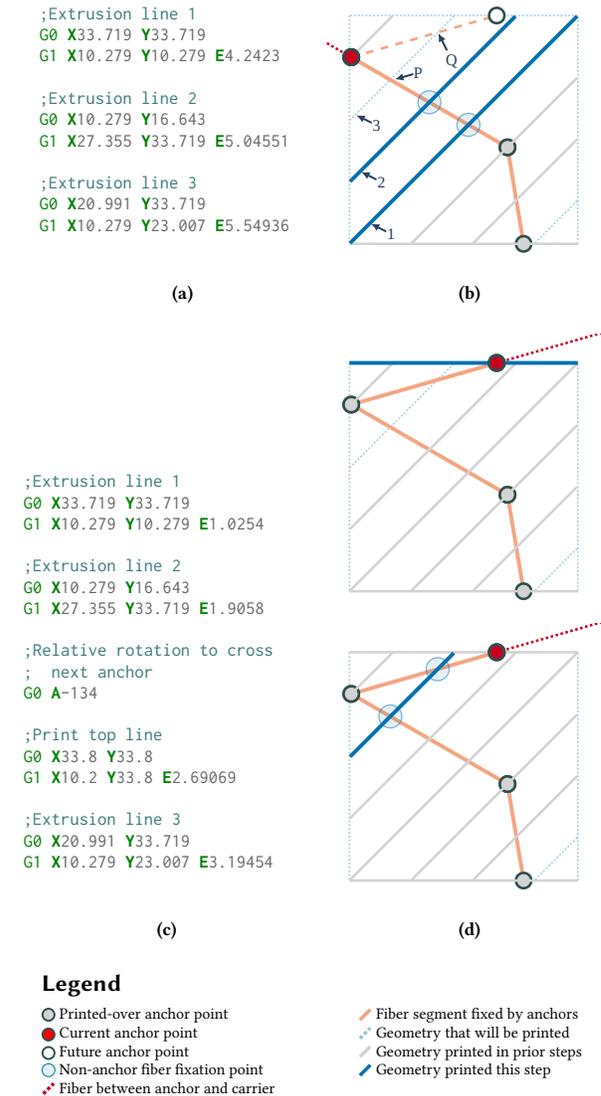


```
;Extrusion line 1
G0 X33.719 Y33.719
G1 X10.279 Y10.279 E4.2423

;Extrusion line 2
G0 X10.279 Y16.643
G1 X27.355 Y33.719 E5.04551

;Extrusion line 3
G0 X20.991 Y33.719
G1 X10.279 Y23.007 E5.54936
```

(a)   **(b)**

```
;Extrusion line 1
G0 X33.719 Y33.719
G1 X10.279 Y10.279 E1.0254

;Extrusion line 2
G0 X10.279 Y16.643
G1 X27.355 Y33.719 E1.9058

;Relative rotation to cross
;  next anchor
G0 A-134

;Print top line
G0 X33.8 Y33.8
G1 X10.2 Y33.8 E2.69069

;Extrusion line 3
G0 X20.991 Y33.719
G1 X10.279 Y23.007 E3.19454
```

(c)   **(d)**

Legend

○ Printed-over anchor point
● Current anchor point
○ Future anchor point
○ Non-anchor fiber fixation point
⋅⋅⋅ Fiber between anchor and carrier

Fiber segment fixed by anchors
Geometry that will be printed
Geometry printed in prior steps
Geometry printed this step

Figure 8: (a) G-code from slicer that would print lines 1–3 of (b) in order. (c) G-code modified by Rhapso to print lines 1 and 2 (b), rotate the fiber and fix it at the anchor (d, top), then print line 3 (d, bottom).

or fragility. The tradeoff is the loss of automation and consequent tedious fiber manipulation work.

The software implementation of the manual printing process uses nearly the same code as the automatic process, but replaces ring movement commands with `M601` pause commands. We also print numbered guide-lines to indicate where the fiber should be moved to at each step. Figure 10 shows an example of the manual embedding process; the total fabrication time, including moving and securing the fiber, was about 10 minutes.
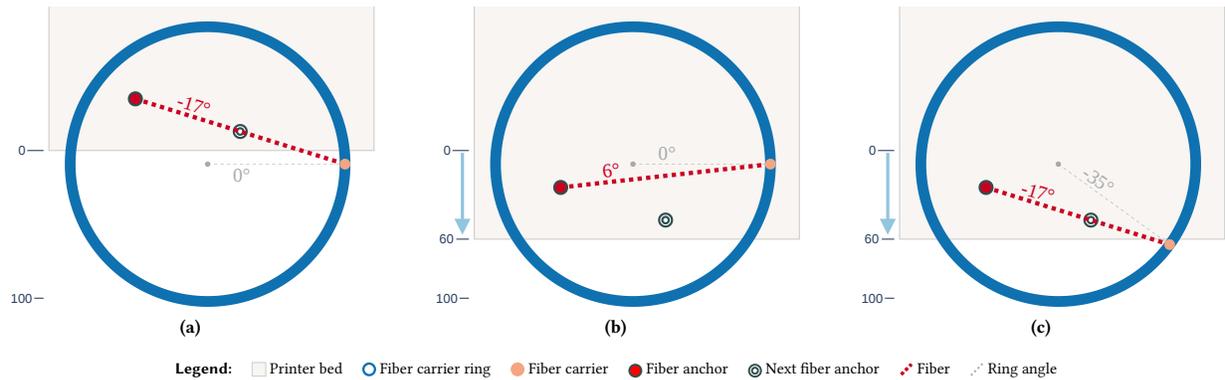
Figure 9: On printers with an XZ-head configuration, the print bed moves relative to the ring, changing the fiber angle. (a) The initial printer state, viewed top-down. (b) If the bed moves and the ring does not, the fiber angle changes. (c) Rhapso synchronizes ring moves to bed moves to keep a constant fiber angle.

5.4 Implementation discussion

In this section we discuss some specific considerations and details of using our Rhapso implementations to embed fiber.

5.4.1 Fabrication-related material considerations. Some fiber properties may necessitate design modifications, extra machine configuration, or special care during and after fabrication. Specifically, some thermal and mechanical properties can interact with the printing process itself: if fiber is not flexible enough, it cannot be bent around an arbitrary corner radius. If fiber is too fragile, it will not sustain tensile forces during fabrication or interaction. Different fibers and their coatings may adhere to different degrees to printed material, or might absorb plastic between their component strands. Unintentional contact with the printer's heated parts may melt or burn vulnerable fibers, and require special methods for affixing them to the printed object without damage. Our software offers configuration parameters for changing printer behavior when the hotend and a fiber cross, but experimentation is required to discover the specific values for each combination of fiber and filament type. As a general rule, configuring Rhapso's hotend/fiber-cross behavior to decrease move speed to 75%, drop extrusion temperature by 5°C,

and increase flow rate by up to 20%, can improve adhesion and prevent the fiber from being dragged by the nozzle.

5.4.2 Fabrication-related print-setting considerations. Standard slicer settings for a given filament type typically require minimal changes to be used with Rhapso. Using a low percentage of infill, or infill patterns with parallel lines, may result in the routing algorithm shifting the fiber path by large amounts in order to ensure anchoring opportunities. Extrusion parameters such as flow rate, temperature, and speed are adjusted by Rhapso during hotend/fiber intersections as detailed above.

5.4.3 Model design considerations. Rhapso's design requires that the fiber between the ring and the anchor point always be under tension; there is currently no way to manipulate free-hanging fiber. Models should be designed with this tension in mind: there is a risk of objects with small footprints being pulled or swept off of the print bed if the fiber path results in large ring rotations.

5.4.4 Fiber-fixing techniques. For "normal" fibers not requiring specific considerations, several approaches can be used to fix the fiber, or prevent its fixation.

First anchor. Given that Rhapso requires that the fiber always be under tension, as a manual first step, the user must secure the loose end of the fiber to the print bed and inform the routing software of its location. We designed a small, reusable anchor that clamps to the edge of the print bed to capture multiple sizes of fiber (Figure 11).

Off-object anchors. Some designs are more easily printed if the fiber is not wholly contained in the object; in these cases, sacrificial objects can be designed to provide anchors external to the main object. Off-object anchors can be *fixing*, where the fiber is captured through the normal *print-over* process, or *wrap-around*, where fiber is tensioned against, but not captured by, the anchor.

The puppets shown in Section 6.1 use *wrap-around* anchor objects to provide extra tension and to allow the fiber to be cut; the hook in Section 6.9 uses a wrap-around anchor to lengthen the fiber between two objects; and the pop-up mechanism in Section 6.5 uses a *fixing* external anchor to provide a fixation point strong enough to withstand the pull of the tensioned elastic fiber.
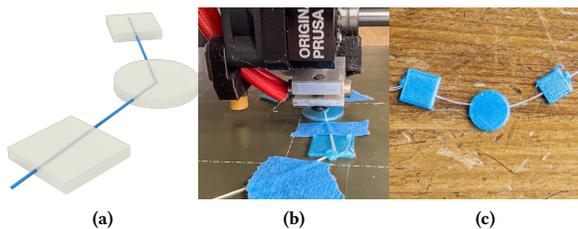

Figure 10: (a) A design for embedding cotton string via the manual printing process. (b) Using tape to secure the string; an anchor has just been printed via the "blob" technique, Section 5.4.4. (c) The resulting beaded string.
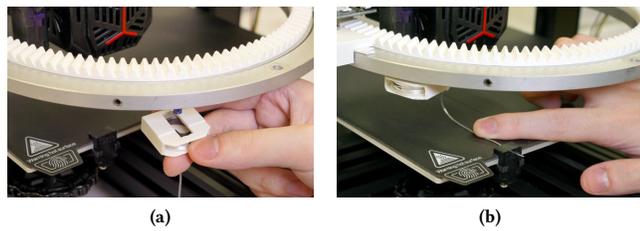
(a)   **(b)**

Figure 11: (a) We attach a fiber spool with a rotational spring inside to the ring. (b) Our bed anchor attaches one end of the fiber to a consistent location on the print bed.

Currently, off-object anchors must be manually created at design time by modeling the anchor structure and setting anchor points. Fixing anchors are specified in exactly the same way as any other internal anchor point, by creating a sketch-line vertex inside the anchor object. Wrapping around anchors is not yet automatically supported by our routing algorithm, and therefore requires manually adding instructions to the generated G-code (Figure 12).
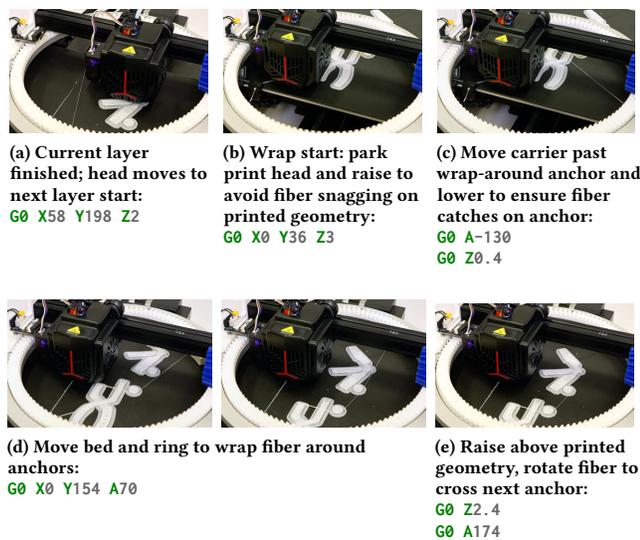
(a) Current layer finished; head moves to next layer start:
`G0 X58 Y198 Z2`

(b) Wrap start: park print head and raise to avoid fiber snagging on printed geometry:
`G0 X0 Y36 Z3`

(c) Move carrier past wrap-around anchor and lower to ensure fiber catches on anchor:
`G0 A-130`
`G0 Z0.4`

(d) Move bed and ring to wrap fiber around anchors:
`G0 X0 Y154 A70`

(e) Raise above printed geometry, rotate fiber to cross next anchor:
`G0 Z2.4`
`G0 A174`

Figure 12: Wrap-around anchoring procedure as used in the "grabber" example (Section 6.7).

In-object anchoring. At times, the *print over* operation by itself does not provide sufficient adhesion to fix a fiber at an anchor point; for example, large fiber rotations at an anchor can overcome the inter-layer adhesion of the plastic and cause the fiber to detach. We developed a technique for printing a **Z**-axis-oriented "blob" to provide extra anchoring strength, illustrated in Figure 13.

5.5 Evaluation

5.5.1 Print-time impact. If the material properties of the fiber and filament do not require reduction in print speed, little additional print time should be expected as a result of automatic fiber embedding. Off-object anchors (Section 5.4.4) are manually designed and
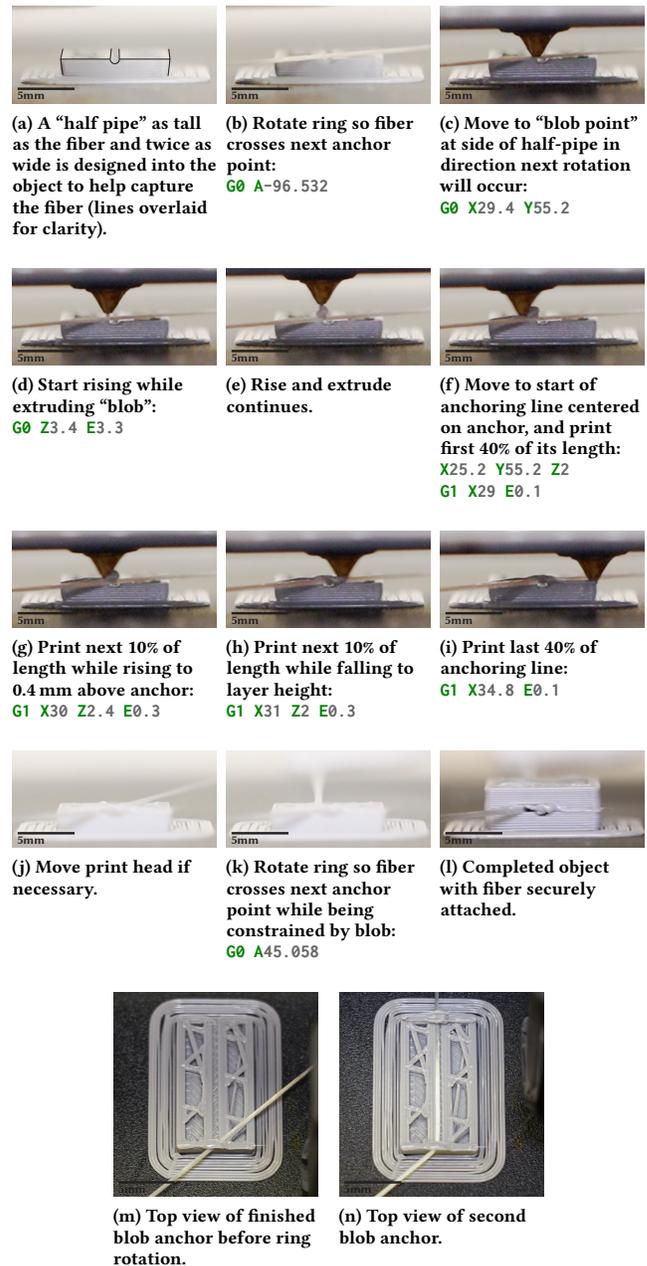
(a) A "half pipe" as tall as the fiber and twice as wide is designed into the object to help capture the fiber (lines overlaid for clarity).

(b) Rotate ring so fiber crosses next anchor point:
`G0 A-96.532`

(c) Move to "blob point" at side of half-pipe in direction next rotation will occur:
`G0 X29.4 Y55.2`

(d) Start rising while extruding "blob":
`G0 Z3.4 E3.3`

(e) Rise and extrude continues.

(f) Move to start of anchoring line centered on anchor, and print first 40% of its length:
`X25.2 Y55.2 Z2`
`G1 X29 E0.1`

(g) Print next 10% of length while rising to 0.4 mm above anchor:
`G1 X30 Z2.4 E0.3`

(h) Print next 10% of length while falling to layer height:
`G1 X31 Z2 E0.3`

(i) Print last 40% of anchoring line:
`G1 X34.8 E0.1`

(j) Move print head if necessary.

(k) Rotate ring so fiber crosses next anchor point while being constrained by blob:
`G0 A45.058`

(l) Completed object with fiber securely attached.

(m) Top view of finished blob anchor before ring rotation.

(n) Top view of second blob anchor.

Figure 13: "Blob" anchoring procedure for stronger anchor/plastic adhesion.

require additional print time according to their geometry. Bed synchronization ring moves (Section 5.2.4) occur simultaneous with bed movement, adding no additional time, while ring moves independent of bed motion—*position* and *avoid*—are infrequent and rapid. As an example, for the grabber illustrated in Figure 23c, non-synchronization ring moves resulted in a print time increase of 1.1%, from 89 to 90 minutes.

5.5.2 Fixing strength evaluation. Although the strength of adhesion between fiber and filament depends on a very wide range of factors [21] including material compositions, printer calibration, fiber location, and model design, we performed some initial tests to convey a rough idea of what to expect from some combinations of fiber material and fixation method.

We designed a simple $10 \times 20 \times 4$ mm cube with the fiber path running horizontally through the cube at a **Z**-height of 2 mm (Figure 14a). We sliced the model with Ultimaker Cura's recommended parameters for generic PLA, with a print temperature of 210℃ and 20% grid infill. After printing each combination of fiber and fixation type, we lightly constrained the cube along the axis of the fiber (to avoid clamping force on the embedded fiber) and pulled the free end with a force gauge (Figure 14b) until the fiber either broke or lost adhesion and slipped out of the test cube.
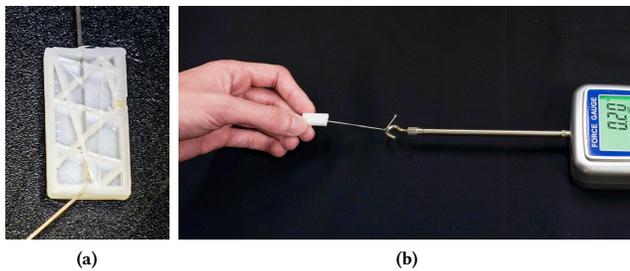

(a) **(b)**

Figure 14: Adhesion test object, shown with spandex fiber. (a) Print paused one layer above fiber-embedding layer, showing layer lines acting as additional anchors. (b) Testing setup.

We tested three fiber types: 0.4 mm nylon fishing line, 0.6 mm spandex thread, and 1.0 mm cotton string. We combined these fibers with two fixation types: normal print-over anchoring with the fiber being captured by six 45° infill lines and four 90° wall lines, and in-object "blob" anchoring as described in Section 5.4.4. The blob-anchoring half-pipe radius for each material was set, based on experience, to 0.4 mm for the fishing line, 0.7 mm for the elastic string, and 0.95 mm for the cotton thread. Table 1 illustrates the results of our testing.

Material	$\varnothing$(mm)	Fixation	Max force (N)	Result
Fishing line	0.4	Normal	1.47	slip
		Blob	14.41	slip/break*
Spandex	0.6	Normal	—	no adhesion
		Blob	3.00	break
Cotton	1.0	Normal	6.56	slip
		Blob	15.73	slip

Table 1: Results of fixation tests. *In some tests, the fishing line broke at the anchor point due to melting.

Our results show that the "blob" fixation method generally produces stronger adhesion between the fiber and the object. We suspect this strength is due to the greater volume of plastic laid on top

of the fiber. However, this method requires modifications to both the model and Rhapso configuration and can result in unaesthetic surface features on the object (Figure 13l). Both the fishing line and spandex thread have very smooth finishes which may make adhesion more difficult, while the cotton string (shown in Figure 24a) is twisted and fuzzy, providing more opportunities for the plastic layers to conform to its shape and capture parts of it.

The factors discussed in this section all contribute to the final properties of any object created with Rhapso. Due to the large number of variables at play it is impossible to provide a single solution to fit all use cases, but the ability to modify the vast majority of them should allow designers to develop objects with the properties they desire.

6 APPLICATIONS

We now present a selection of applications built using our automated implementation of Rhapso (Section 5.1), illustrating the breadth of possibility offered by fiber embedding, as well as further validating the technique's ability to work with a variety of fiber types and configurations. All of the examples shown here were automatically fabricated on our prototype printers.

6.1 Articulated puppets

We created two articulated puppets. The first, modeled after a popular cartoon character (Figure 15), demonstrates transforming translational forces into rotary motion. Pulling the embedded cotton thread causes the puppet to lift its arms, while releasing the thread allows gravity to return the arms to a lowered position. We used an off-object wrap-around anchor to enable us to clip the thread (Figure 15c); routing the thread directly between the two arms in the object's interior would have prevented their moving.

The second puppet (Figure 16) also uses elastic thread as a force transmitter. To ensure the right amount of stretch, the elastic fiber was embedded under full tension during printing using an extra, off-object wrap-around anchor (Figure 16c). When removed from the print bed, the extra anchor was discarded and the fiber relaxed. When the lever is pressed, the robot stands straight and the fiber is tensed; when released, the fiber and robot relax.

This application demonstrates that Rhapso enables novel articulation within produced objects using fibers.
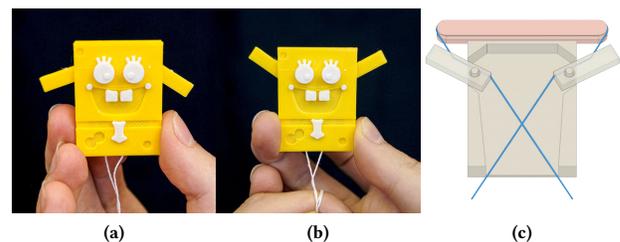

(a) **(b)** **(c)**

Figure 15: Articulated cartoon puppet with embedded cotton thread (a) slack and (b) pulled. (c) Interior thread route.
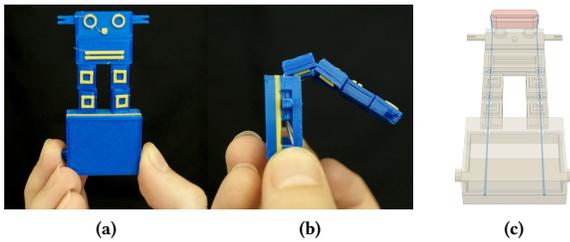
Figure 16: Robot puppet with embedded pre-tensioned elastic fiber. (a) Base pulled down to keep fiber tensed. (b) Base released to allow fiber to relax. (c) Fiber route through robot.

6.2 Abacus

We created a simple abacus inspired by the Japanese Soroban[3] (Figure 17) that illustrates how forces can be decoupled from fibers. The fiber is fixed to the frame at both ends, and the beads are printed around it with 2mm tubes; they can slide freely while the fiber and frame remain in-place.

This application demonstrates that Rhapso enables fibers as central structural elements of produced objects.
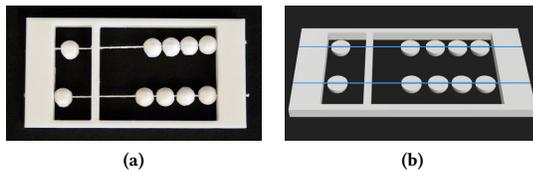
Figure 17: (a) Abacus printed with free-sliding beads. (b) Cutaway view of embedded fiber design.

6.3 Pull-to-assemble box

Conversely, this decoupling can enable the fiber to move relative to the object: we used this technique in combination with fiber hinges that transmit forces around corners to create a box which is printed flat but can be quickly assembled into 3D. The box (Figure 18) uses interlocking tabs for alignment; cutaways at the edges enable fiber tension to pull each pair of sides to fold to 90°.

This application demonstrates that Rhapso enables unique post-print assembly, shape change, and novel geometries.

6.4 Self-assembling boxes

Embedded, force-originating fiber presents opportunities for objects which can self-assemble. Similar to the pull-to-assemble box, we created several boxes using elastic fiber printed under tension which fold up into closed shapes when removed from the printer (Figure 19). To ensure sufficient tension, we used an off-object anchor to keep the thread pulled to its full extent (Figure 20).

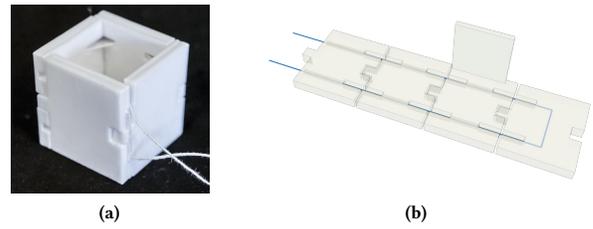This application demonstrates that Rhapso enables the use of fiber tension as a structural element.

---

[3]https://en.wikipedia.org/wiki/Soroban

Figure 18: (a) Box printed flat, assembled by pulling threads. (b) Transparent view showing internal thread routing.
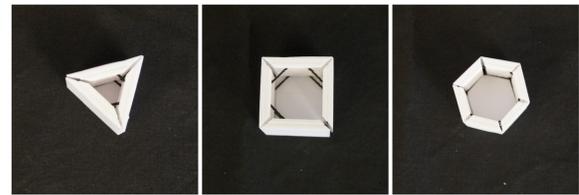

Figure 19: Self-assembling boxes with elastic fiber at various angles 30 degrees (left), 45 degrees (middle) and 60 degrees (right)
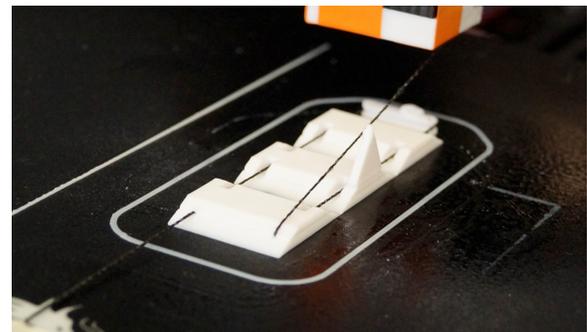

Figure 20: Self-assembling box on the build plate, with an off-object anchor visible just behind the object.

6.5 Tensioned Pop-Up Mechanism

We printed a two-part scissor mechanism—similar to that used in some keyboard switches—which includes tensioned elastic thread as a force originator (Figure 21). When the device is removed from the print bed, it pops up to a 3D configuration and can be repeatedly pressed while popping back up.

This application demonstrates that Rhapso enables the use of fiber tension as a mechanical element.

6.6 Robot finger

We designed a robot finger with a single fiber divided into two independent segments. The upper fiber segment transmits a pulling force to the tip of the finger, causing the finger to curl; integrated channels allow the fiber to slide between the parts of the finger. The lower segment, fixed at both ends of the finger, originates a spring force, straightening the finger when the pulling force is released.
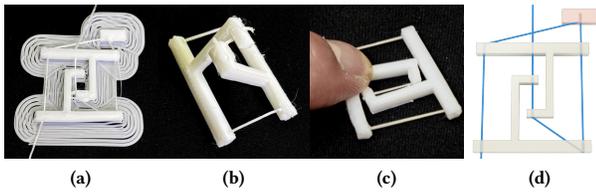
Figure 21: (a) A pop-up mechanism printed flat while the thread is in tension. (b) The mechanism contracts and stands on its own after removing the brim. (c) The structure works like the scissor mechanism in keyboards. (d) The design of the fiber path; note the off-object anchor (pink).

In this way, this application demonstrates the ability of Rhapso to blend forces, as its position is a blend of user input and printed tension.
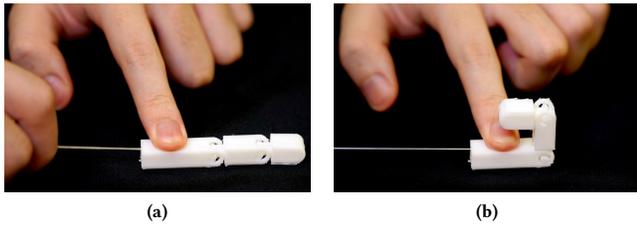

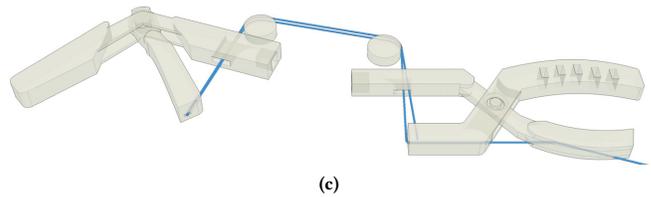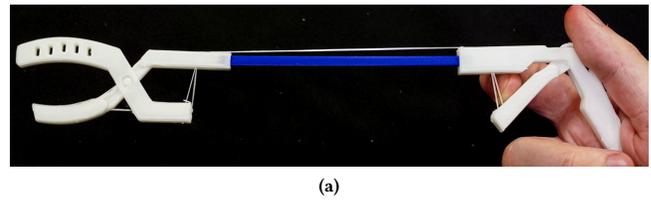**Figure 22: (a) A printed robot finger relaxing and (b) contracting.**

Figure 23: Our grabber device was printed as two plastic parts connected by thread and which, after printing, are rigidly connected using an inserted stick.

6.7 Grabber

We printed a reach-extending "grabber" device which uses elastic fiber to transmit force from a user's hand to a distant pincer mechanism. This object was printed with two separated plastic parts connected by the fiber, such that the fiber also acts as a measurement tool to help the user insert a stick of the correct length during final assembly.

This application demonstrates that Rhapso enables the creation of objects that leverage transfer of force along the path of the fiber.

6.8 Fiber "hair"

Fiber can enter and exit a single object multiple times, in a single plane or across multiple planes. We used this technique with cotton thread to generate loose, multi-plane loops which give a hair-like texture to the printed object (compare to 3D-Printed Hair [33]). These loops can also be cut post-print for a "furrier" texture.

This application demonstrates that Rhapso enables integrating novel aesthetic and textural properties on created objects.

6.9 Extended hook

Fiber is not bound to print bed size in the same manner as extruded thermoplastic, as explored previously by Rivera et al. [65]. Here, we fabricated a two-part over-door hook with connecting fibers that
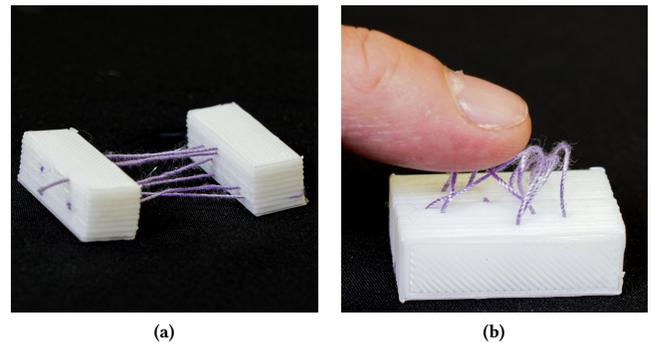

Figure 24: (a) The fiber enters and exits an object multiple times and across multiple layers. (b) The object can fold to create a hair-like texture.

stretch longer than the print-bed (Figure 25) by adding a sacrificial spool for thread winding during the print.

This application demonstrates that Rhapso enables the creation of objects whose use spans larger volumes than the print bed, as well as demonstrates the robustness of fixed fiber adhesion strength when placed under strain.
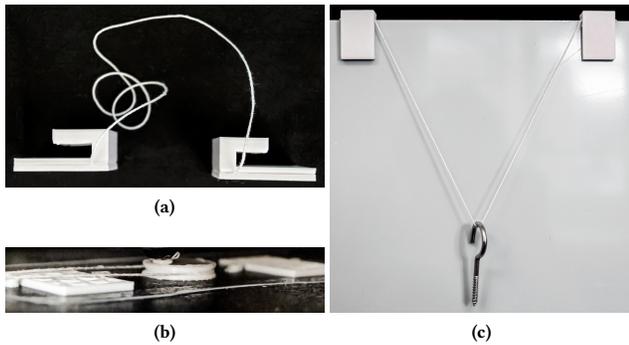
Figure 25: Embedded fiber can enable fabricating objects larger than the printer' build volume. (a) A hook printed to hang from a door (c) can be created by wrapping extra fiber around a temporary structure on the print bed (b).

6.10 Design patterns

During the course of exploring the applications presented above, we recognized a number of recurring patterns in our fiber-embedded designs. Here we briefly discuss these patterns and how they might be generalized for use in further designs.

6.10.1 Slide-guides. Fiber selectively left free to move allows printing objects with tubes through which fiber freely passes, creating a "guide" along which an object slides like a bead on a string. Conversely, guiding structures can be printed around non-fixed fibers to enable the *fiber* to slide. This pattern appears in the *abacus* (Section 6.2) and *pull-to-assemble box* (Section 6.3) applications.

6.10.2 Mechanical motion transfer. Push- and pull-cable mechanisms are commonly used to transfer power between locations. Examples include Bowden cables used in bicycle braking systems and robots [75]. Using guiding tubes, fiber can be pulled from one location to create actuated motion in another part of the object. Our *grabber* and *robot finger* applications (Sections 6.6 and 6.7) illustrate this pattern.

The motion-transfer effect can be multiplied using *fiber arrangement* techniques to achieve *blending of forces*; by attaching a single user-facing fiber to multiple object-interfacing threads, motion could be transferred to several locations, directions, or planes.

6.10.3 Hinges. In contrast to common printed thermoplastics, many common fibers have a nearly unlimited ability to bend without weakening. One or more strands of fiber affixed between two parts of an object can form strong and flexible hinges, allowing movement or assembly by concentrating bending force in the most flexible part of an object. The *pull-to-assemble* box (Section 6.3) uses fiber hinges, allowing it to be assembled and disassembled many times.

6.10.4 Elastic "springs". Spring-like behavior can be created by using elastic fiber with inelastic thermoplastic. Tensioning the fiber before fixing it in place creates stored energy that can be released for self-assembly, while un-tensioned elastic fiber can resist displacement of a moving part and return it to its original position.

Several of our applications use pre-tensioned elastic fiber: the *self-assembling box*, the *pop-up mechanism*, the *robot finger*, and the *grabber* (Sections 6.4–6.7).

6.10.5 Extending objects. Due to printer size constraints, some larger objects can be difficult to fabricate via typical methods. In some cases, fiber can replace portions of models that do not need to be composed of rigid plastic, enabling the creation of objects larger than the print bed. Our *extended hook* example (Section 6.9) used fiber to lengthen the reach of a hook; while the *grabber* (Section 6.7) needs a solid strut in between its two components, the connecting fiber must be continuous.

7 DISCUSSION AND FUTURE WORK

Rhapso opens the door to general-purpose exploration of fiber embedding on commodity printers. In this context, we reflect on opportunities for our current system implementation and its possible utility to future research.

7.1 System generalizability

Our implementation of Rhapso is based on modifications to a specific low-cost FFF printer. Adapting our design for similar printer types is relatively straightforward, primarily requiring adjustments to the carrier-ring mounts. Printers with non-moving beds would require a different mounting mechanism, but are already supported by our software.

Beyond porting Rhapso to other FFF printers, adapting the concept to different fabrication technologies such as stereolithography (SLA) 3D printing or subtractive devices like laser cutters, while challenging, would open up interesting new material/fiber interaction possibilities.

7.2 Limitations of the Rhapso architecture

The Rhapso architecture concept, as presented in this work, does have some specific limitations that notably restrict the design space of objects created using it. The primary limitation is the fact that only a single fiber can be embedded at a time, which cannot be cut and reattached, or switched with another type of fiber mid-print. This limitation also imposes certain restrictions on the geometries that can be produced, as the fiber can only monotonically increase its height along with the printed object layer height, and complex geometries such as spirals can prove difficult. Another limitation is that the embedded fiber length cannot exceed that which is initially loaded into the spool - although this was not an impediment in any of our tests, it limits the size and complexity of design.

7.3 Ideas to extend the current hardware

Using a larger-volume printer could address some of the limitations above by allowing space for a larger spool and longer fibers. The spool design could be customized for fibers with larger bending radii. Beyond direct improvements to the existing system, multi-axis 3D printers [55], or even the application of robot arms for fiber manipulation [22] could significantly expand the capabilities of the system such as allowing fiber embedding in non-monotonic directions, or switching between different fiber spools.

Future work could potentially enable more sophisticated fiber tensioning mechanisms to be placed in the extra space, if accompanied by appropriate software expansion. As an early-stage attempt to add additional control in the fiber embedding manipulator, we replaced the fiber carrier with a motorized spool that can control the tension of embedded elastic fiber between any two anchor points (Figure 26). We built three flexible samples with the same fiber embedding geometry, tensioning the embedded elastic thread to different extents and creating different final curvatures (Figure 1d).
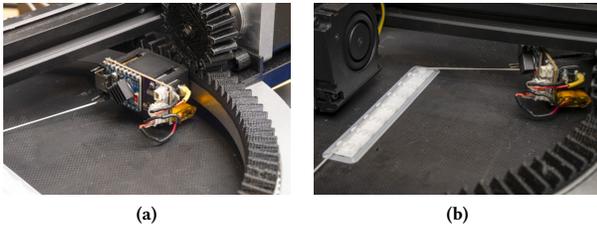

(a) (b)

Figure 26: Automatic tensioning feeder: (a) the automatic tensioning feeder controlled via Wi-Fi through an ESP32 module, (b) the feeder applying tension between anchor points. Figure 1d shows three flexible samples printed with the same geometry embedded with elastic threads elongated at different ratios achieved with controlled tensioning.

7.4 Other and multiple types of fibers

We are also interested in the possibilities offered by embedding other types of fibers. For example, preliminary tests showed that optical fibers suffered from poor adhesion and a tendency to melt when contacted by the hotend. Embedding conductive thread or thin wire offers many sensing and output possibilities, but such fibers may prove prone to tangling. Future work could also investigate McKibben muscle actuators [29] as another fiber material. The relative inflexibility of such fibers means that depositing them would require additional care and space, as they cannot be spooled as tightly. The optical, electrical, thermal, and hybrid properties of these materials could be added to the design space of Rhapso.

Our current implementation only allows adding a single continuous strand of a single fiber type into an object. Adding more fiber carriers would open up new design opportunities, such as paired elastic and inelastic fibers that respond like muscles and tendons, wires carrying multiple electrical signals, or multi-colored fiber art. At the same time, additional fibers would necessitate new hardware designs and algorithms to avoid undesired fiber tangling.

7.5 Fiber-friendly design tools

Autodesk Fusion was adequate to create our application designs, but a fiber-aware design tool would simplify the process and enable a wider variety of designs. Being able to simulate fiber properties such as adhesion, bending radius, elongation, and so forth (Section 3.1) would reduce the time spent prototyping. Additionally, our current process relies on the designer to predict the impact of fiber diameter on the object and printing process; a fiber-aware

tool could automatically adapt the design to accommodate different thicknesses.

8 CONCLUSION

We presented Rhapso, a software and hardware system to automatically embed fiber materials in FFF prints for enabling richer interactions and capabilities in low-cost 3D objects. Rhapso and its open source implementation invite HCI researchers and hobbyist makers to explore the advantages of multi-material fabrication and go beyond purely thermoplastic-based FFF prints [46]. Rhapso solves several algorithmic challenges in fiber-based printing, and provides a platform from which we explored several points in the design space of fiber-embedding: taking into consideration pre-print fiber properties, during-print fiber-object structures, and post-print interactive capabilities. We also shared ten example objects which highlight both novel interactions made possible by automated fiber embedding and our system's ability to replicate previously handmade interactive objects.

ACKNOWLEDGMENTS

REFERENCES

[1] Lea Albaugh, James McCann, Scott E. Hudson, and Lining Yao. 2021. Engineering Multifunctional Spacer Fabrics Through Machine Knitting. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3411764.3445564

[2] Jeffery W Baur, Andrew C Abbott, Philip R Barnett, Gyaneshwar P Tandon, Jevan Furmanski, Nathan A Stranberg, and Tyler B Alvarado. 2023. Mechanical Properties of Additively Printed, UV Cured, Continuous Fiber Unidirectional Composites for Multifunctional Applications. *Journal of Composite Materials* 57, 4 (Feb. 2023), 865–882. https://doi.org/10.1177/00219983221146264

[3] Kazi Md Masum Billah, Jose L. Coronel, Luis Chavez, Yirong Lin, and David Espalin. 2021. Additive Manufacturing of Multimaterial and Multifunctional Structures via Ultrasonic Embedding of Continuous Carbon Fiber. *Composites Part C: Open Access* 5 (July 2021), 100149. https://doi.org/10.1016/j.jcomc.2021.100149

[4] Kazi Md Masum Billah, Jose L. Coronel, Michael C. Halbig, Ryan B. Wicker, and David Espalin. 2019. Electrical and Thermal Characterization of 3D Printed Thermoplastic Parts With Embedded Wires for High Current-Carrying Applications. *IEEE Access* 7 (2019), 18799–18810. https://doi.org/10.1109/ACCESS.2019.2895620

[5] Xiang 'Anthony' Chen, Stelian Coros, and Scott E. Hudson. 2018. *Medley: A Library of Embeddables to Explore Rich Material Properties for 3D Printed Objects.* Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3173736

[6] Xiang 'Anthony' Chen, Stelian Coros, Jennifer Mankoff, and Scott E Hudson. 2015. Encore: 3D Printed Augmentation of Everyday Objects with Printed-Over, Affixed and Interlocked Attachments. In *UIST '15: Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology* (New York, New York, USA). ACM Press, Charlotte, NC, USA, 73–82.

[7] Ashley Del Valle, Mert Toka, Alejandro Aponte, and Jennifer Jacobs. 2023. Punch-Print: Creating Composite Fiber-Filament Craft Artifacts by Integrating Punch Needle Embroidery and 3D Printing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3544548.3581298

[8] Himani Deshpande, Haruki Takahashi, and Jeeeun Kim. 2021. EscapeLoom: Fabricating New Affordances for Hand Weaving. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 630, 13 pages. https://doi.org/10.1145/3411764.3445600

[9] Laura Devendorf and Kimiko Ryokai. 2015. Being the Machine: Reconfiguring Agency and Control in Hybrid Fabrication. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (New York,

New York, USA, 2015). ACM Press, Seoul, Korea, 2477–2486. https://doi.org/10.1145/2702123.2702547

[10] Shreyosi Endow, Mohammad Abu Nasir Rakib, Anvay Srivastava, Sara Rastegarpouyani, and Cesar Torres. 2022. Embr: A Creative Framework for Hand Embroidered Liquid Crystal Textile Displays. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3491102.3502117

[11] David Espalin, Danny W. Muse, Eric MacDonald, and Ryan B. Wicker. 2014. 3D Printing Multifunctionality: Structures with Electronics. *The International Journal of Advanced Manufacturing Technology* 72, 5 (May 2014), 963–978. https://doi.org/10.1007/s00170-014-5717-7

[12] Aluna Everitt, Alexander Keith Eady, and Audrey Girouard. 2022. Enabling Multi-Material 3D Printing for Designing and Rapid Prototyping of Deformable and Interactive Wearables. In *Proceedings of the 20th International Conference on Mobile and Ubiquitous Multimedia (MUM '21)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3490632.3490635

[13] Jack Forman, Mustafa Doga Dogan, Hamilton Forsythe, and Hiroshi Ishii. 2020. DefeXtiles: 3D Printing Quasi-Woven Fabric via Under-Extrusion. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 1222–1233. https://doi.org/10.1145/3379337.3415876

[14] Jack Forman, Taylor Tabb, Youngwook Do, Meng-Han Yeh, Adrian Galvin, and Lining Yao. 2019. ModiFiber: Two-Way Morphing Soft Thread Actuators for Tangible Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3290605.3300890

[15] Mikhaila Friske, Shanel Wu, and Laura Devendorf. 2019. AdaCAD: Crafting Software For Smart Textiles Design. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300575

[16] Wei Gao, Yunbo Zhang, Diogo C. Nazzetta, Karthik Ramani, and Raymond J. Cipra. 2015. RevoMaker: Enabling Multi-Directional and Functionally-Embedded 3D Printing Using a Rotational Cuboidal Platform. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 437–446. https://doi.org/10.1145/2807442.2807476

[17] Jun Gong, Yu Wu, Lei Yan, Teddy Seyed, and Xing-Dong Yang. 2019. Tessutivo: Contextual Interactions on Interactive Fabrics with Inductive Sensing. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. ACM, New York, NY, USA, 29–41. https://doi.org/10.1145/3332165.3347897

[18] Maas Goudswaard, Abel Abraham, Bruna Goveia da Rocha, Kristina Andersen, and Rong-Hao Liang. 2020. FabriClick: Interweaving Pushbuttons into Fabrics Using 3D Printing and Digital Embroidery. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference* (Eindhoven, Netherlands) *(DIS '20)*. Association for Computing Machinery, New York, NY, USA, 379–393. https://doi.org/10.1145/3357236.3395569

[19] Bruna Goveia da Rocha, Johannes M. L. van der Kolk, and Kristina Andersen. 2021. Exquisite Fabrication: Exploring Turn-taking between Designers and Digital Fabrication Machines. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–9. https://doi.org/10.1145/3411764.3445236

[20] Ruslan Guseinov, Eder Miguel, and Bernd Bickel. 2017. CurveUps: Shaping Objects from Flat Plates with Tension-Actuated Curvature. *ACM Trans. Graph.* 36, 4, Article 64 (July 2017), 12 pages. https://doi.org/10.1145/3072959.3073709

[21] Razieh Hashemi Sanatgar, Christine Campagne, and Vincent Nierstrasz. 2017. Investigation of the Adhesion Properties of Direct 3D Printing of Polymers and Nanocomposites on Textiles: Effect of FDM Printing Process Parameters. *Applied Surface Science* 403 (2017), 551–563. https://doi.org/10.1016/j.apsusc.2017.01.112

[22] Harrison Hildebrandt, Mengxi He, Peng-An Chen, Rebeca Duque Estrada, Christoph Zechmeister, and Achim Menges. 2024. Slack Pack: Fabrication System for the Dual Robotic Winding of Spatial Fiber Structures. In *The International Conference on Computational Design and Robotic Fabrication (Computational Design and Robotic Fabrication)*, Chao Yan, Hua Chai, Tongyue Sun, and Philip F. Yuan (Eds.). Springer Nature, Singapore, 476–491. https://doi.org/10.1007/978-981-99-8405-3_40

[23] Jonathan Hook, Thomas Nappey, Steve Hodges, Peter Wright, and Patrick Olivier. 2014. Making 3D Printed Objects Interactive Using Wireless Accelerometers. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1435–1440. https://doi.org/10.1145/2559206.2581137

[24] Scott E Hudson. 2014. Printing Teddy Bears: A Technique for 3D Printing of Soft Interactive Objects. In *CHI'14: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM Press, New York, New York, USA, 459–468. https://doi.org/10.1145/2556288.2557338

[25] Yehia Ibrahim, Garrett W. Melenka, and Roger Kempers. 2018. Additive Manufacturing of Continuous Wire Polymer Composites. *Manufacturing Letters* 16 (April 2018), 49–51. https://doi.org/10.1016/j.mfglet.2018.04.001

[26] Alexandra Ion, Robert Kovacs, Oliver S Schneider, Pedro Lopes, and Patrick Baudisch. 2018. Metamaterial Textures. In *CHI '18: Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems*. ACM, New York, New York, USA, 336–12. https://doi.org/10.1145/3173574.3173910

[27] M. N. Jahangir, K. M. M. Billah, Y. Lin, D. A. Roberson, R. B. Wicker, and D. Espalin. 2019. Reinforcement of Material Extrusion 3D Printed Polycarbonate Using Continuous Carbon Fiber. *Additive Manufacturing* 28 (Aug. 2019), 354–364. https://doi.org/10.1016/j.addma.2019.05.019

[28] Snehal Jain, Thileepan Stalin, Elgar Kanhere, and Pablo Valdivia y Alvarado. 2020. Flexible Fiber Interconnects for Soft Mechatronics. *IEEE Robotics and Automation Letters* 5, 3 (July 2020), 3907–3914. https://doi.org/10.1109/LRA.2020.2982367

[29] Ozgun Kilic Afsar, Ali Shtarbanov, Hila Mor, Ken Nakagaki, Jack Forman, Karen Modrei, Seung Hee Jeong, Klas Hjort, Kristina Höök, and Hiroshi Ishii. 2021. OmniFiber: Integrated Fluidic Fiber Actuators for Weaving Movement Based Interactions into the 'Fabric of Everyday Life'. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. Association for Computing Machinery, New York, NY, USA, 1010–1026. https://doi.org/10.1145/3472749.3474802

[30] Chiyen Kim, David Espalin, Alejandro Cuaron, Mireya A. Perez, Mincheol Lee, Eric MacDonald, and Ryan B. Wicker. 2015. Cooperative Tool Path Planning for Wire Embedding on Additively Manufactured Curved Surfaces Using Robot Kinematics. *Journal of Mechanisms and Robotics* 7, 2 (May 2015), 021003. https://doi.org/10.1115/1.4029473

[31] Hyunyoung Kim, Aluna Everitt, Carlos Tejada, Mengyu Zhong, and Daniel Ashbrook. 2021. MorpheesPlug: A Toolkit for Prototyping Shape-Changing Interfaces. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2021-05-06). Association for Computing Machinery, Online Virtual Conference, 1–13. https://doi.org/10.1145/3411764.3445786

[32] Hannah Carlotta Koch, David Schmelzeisen, and Thomas Gries. 2021. 4D Textiles Made by Additive Manufacturing on Pre-Stressed Textiles—An Overview. *Actuators* 10, 2 (Feb. 2021), 31. https://doi.org/10.3390/act10020031

[33] Gierad Laput, Xiang 'Anthony' Chen, and Chris Harrison. 2015. 3D Printed Hair: Fused Deposition Modeling of Soft Strands, Fibers, and Bristles. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 593–597. https://doi.org/10.1145/2807442.2807484

[34] David Ledo, Fraser Anderson, Ryan Schmidt, Lora Oehlberg, Saul Greenberg, and Tovi Grossman. 2017. Pineal: Bringing Passive Objects to Life with Embedded Mobile Devices. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2583–2593. https://doi.org/10.1145/3025453.3025652

[35] Joanne Leong, Jose Martinez, Florian Perteneder, Ken Nakagaki, and Hiroshi Ishii. 2020. WraPr: Spool-Based Fabrication for Object Creation and Modification. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '20)*. Association for Computing Machinery, Sydney NSW, Australia, 581–588. https://doi.org/10.1145/3374920.3374990

[36] Yue Li, Juan Montes, Bernhard Thomaszewski, and Stelian Coros. 2022. Programmable Digital Weaves. *IEEE Robotics and Automation Letters* 7, 2 (April 2022), 2891–2896. https://doi.org/10.1109/LRA.2022.3145948

[37] Zhijian Li, Li Wang, Guowei Ma, Jay Sanjayan, and Duo Feng. 2020. Strength and Ductility Enhancement of 3D Printing Structure Reinforced by Embedding Continuous Micro-Cables. *Construction and Building Materials* 264 (Dec. 2020), 120196. https://doi.org/10.1016/j.conbuildmat.2020.120196

[38] Marco Livesu, Stefano Ellero, Jonàs Martínez, Sylvain Lefebvre, and Marco Attene. 2017. From 3D Models to 3D Prints: An Overview of the Processing Pipeline. *Computer Graphics Forum* 36, 2 (May 2017), 537–564. https://doi.org/10.1111/cgf.13147

[39] Yiyue Luo, Kui Wu, Tomás Palacios, and Wojciech Matusik. 2021. KnitUI: Fabricating Interactive and Sensing Textiles with Machine Knitting. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3411764.3445780

[40] Yiyue Luo, Kui Wu, Andrew Spielberg, Michael Foshey, Daniela Rus, Tomás Palacios, and Wojciech Matusik. 2022. Digital Fabrication of Pneumatic Actuators with Integrated Sensing by Machine Knitting. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3491102.3517577

[41] Eric MacDonald and Ryan Wicker. 2016. Multiprocess 3D Printing for Increasing Component Functionality. *Science* 353, 6307 (Sept. 2016), 10. https://doi.org/10.1126/science.aaf2093

[42] Markforged. 2016. Carbon Fiber Composite 3D Printer: Markforged Mark Two. https://markforged.com/3d-printers/mark-two.

[43] James McCann, Lea Albaugh, Vidya Narayanan, April Grow, Wojciech Matusik, Jennifer Mankoff, and Jessica Hodgins. 2016. A Compiler for 3D Machine Knitting. *ACM Transactions on Graphics* 35, 4 (July 2016), 49:1–49:11. https://doi.org/10.1145/2897824.2925940

[44] Viktor Mechtcherine, Richard Buswell, Harald Kloft, Freek P. Bos, Norman Hack, Rob Wolfs, Jay Sanjayan, Behzad Nematollahi, Egor Ivaniuk, and Tobias Neef. 2021. Integrating Reinforcement in Digital Fabrication with Concrete: A Review and Classification Framework. *Cement and Concrete Composites* 119 (May 2021), 103964. https://doi.org/10.1016/j.cemconcomp.2021.103964

[45] Annika Muehlbradt, Gregory Whiting, Shaun Kane, and Laura Devendorf. 2022. Knitting Access: Exploring Stateful Textiles with People with Disabilities. In *Proceedings of the 2022 ACM Designing Interactive Systems Conference (DIS '22)*. Association for Computing Machinery, New York, NY, USA, 1058–1070. https://doi.org/10.1145/3532106.3533551

[46] Stefanie Mueller. 2018. Interacting with Personal Fabrication Devices. *IT - Information Technology* 60, 2 (April 2018), 113–117. https://doi.org/10.1515/itit-2017-0041

[47] 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)*. Association for Computing Machinery, New York, NY, USA, 273–280. https://doi.org/10.1145/2642918.2647359

[48] Sachith Muthukumarana, Moritz Alexander Messerschmidt, Denys J.C. Matthies, Jürgen Steimle, Philipp M. Scholl, and Suranga Nanayakkara. 2021. ClothTiles: A Prototyping Platform to Fabricate Customized Actuators on Clothing Using 3D Printing and Shape-Memory Alloys. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3411764.3445613

[49] Sara Nabil, Jan Kučera, Nikoletta Karastathi, David S. Kirk, and Peter Wright. 2019. Seamless Seams: Crafting Techniques for Embedding Fabrics with Interactive Actuation. In *Proceedings of the 2019 on Designing Interactive Systems Conference (DIS '19)*. Association for Computing Machinery, New York, NY, USA, 987–999. https://doi.org/10.1145/3322276.3322369

[50] Ken Nakagaki, Sean Follmer, Artem Dementyev, Joseph A. Paradiso, and Hiroshi Ishii. 2017. Designing Line-Based Shape-Changing Interfaces. *IEEE Pervasive Computing* 16, 4 (Oct. 2017), 36–46. https://doi.org/10.1109/MPRV.2017.3971127

[51] nervous. kinematics. Nervous System | Kinematics. https://n-e-r-v-o-u-s.com/shop/line.php?code=15.

[52] Alex Olwal, Jon Moeller, Greg Priest-Dorman, Thad Starner, and Ben Carroll. 2018. I/O Braid: Scalable Touch-Sensitive Lighted Cords Using Spiraling, Repeating Sensing Textiles and Fiber Optics. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. Association for Computing Machinery, New York, NY, USA, 485–497. https://doi.org/10.1145/3242587.3242638

[53] Jefferson Pardomuan, Nobuhiro Takahashi, and Hideki Koike. 2022. ASTRE: Prototyping Technique for Modular Soft Robots With Variable Stiffness. *IEEE Access* 10 (2022), 80495–80504. https://doi.org/10.1109/ACCESS.2022.3194887

[54] Young-Woo Park, Joohee Park, and Tek-Jin Nam. 2015. The Trial of Bendi in a Coffeehouse: Use of a Shape-Changing Device for a Tactile-Visual Phone Conversation. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 2181–2190.

[55] 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* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3174153

[56] Huaishu Peng, François Guimbretière, James McCann, and Scott Hudson. 2016. A 3D Printer for Interactive Electromagnetic Devices. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. Association for Computing Machinery, New York, NY, USA, 553–562. https://doi.org/10.1145/2984511.2984523

[57] Huaishu Peng, Jennifer Mankoff, Scott E Hudson, and James McCann. 2015. A Layered Fabric 3D Printer for Soft Interactive Objects. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM Press, New York, New York, USA, 1789–1798. https://doi.org/10.1145/2702123.2702327

[58] Aby Raj Plamootil Mathai, Thileepan Stalin, and Pablo Valvivia y Alvarado. 2022. Flexible Fabric Inductive Coils for Soft Robots and Wearable Devices. *IEEE Robotics and Automation Letters* 7, 2 (April 2022), 5711–5718. https://doi.org/10.1109/LRA.2022.3159864

[59] Andreas Pointner, Thomas Preindl, Sara Mlakar, Roland Aigner, Mira Alida Haberfellner, and Michael Haller. 2022. Knitted Force Sensors. In *Adjunct Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 1–3. https://doi.org/10.1145/3526114.3558656

[60] Jing Qiao, Yingrui Li, and Longqiu Li. 2019. Ultrasound-Assisted 3D Printing of Continuous Fiber-Reinforced Thermoplastic (FRTP) Composites. *Additive Manufacturing* 30 (Dec. 2019), 100926. https://doi.org/10.1016/j.addma.2019.100926

[61] Christoph Richter, Stefan Schmülling, Andrea Ehrmann, and Karin Finsterbusch. 2015. FDM Printing of 3D Forms with Embedded Fibrous Materials. In *Design,*

[62] Michael L. Rivera, Jack Forman, Scott E. Hudson, and Lining Yao. 2020. Hydrogel-Textile Composites: Actuators for Shape-Changing Interfaces. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–9. https://doi.org/10.1145/3334480.3382788

[63] Michael L. Rivera and Scott E. Hudson. 2019. Desktop Electrospinning: A Single Extruder 3D Printer for Producing Rigid Plastic and Electrospun Textiles. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300434

[64] Michael L Rivera, Melissa Moukperian, Daniel Ashbrook, Jennifer Mankoff, and Scott E Hudson. 2017. Stretching the Bounds of 3D Printing with Embedded Textiles. In *CHI '17: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 497–508. https://doi.org/10.1145/3025453.3025460

[65] Michael L. Rivera, Melissa Moukperian, Daniel Ashbrook, Jennifer Mankoff, and Scott E. Hudson. 2017. *Stretching the Bounds of 3D Printing with Embedded Textiles*. Association for Computing Machinery, New York, NY, USA, 497–508. https://doi.org/10.1145/3025453.3025460

[66] Valkyrie Savage, Colin Chang, and Björn Hartmann. 2013. Sauron: Embedded Single-Camera Sensing of Printed Physical User Interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 447–456. https://doi.org/10.1145/2501988.2501992

[67] Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. 2015. Capricate: A Fabrication Pipeline to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 253–258. https://doi.org/10.1145/2807442.2807503

[68] Martin Schmitz, Jan Riemann, Florian Müller, Steffen Kreis, and Max Mühlhäuser. 2021. Oh, Snap! A Fabrication Pipeline to Magnetically Connect Conventional and 3D-Printed Electronics. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2021-05-07) *(CHI '21)*. Association for Computing Machinery, Online Virtual Conference, 1–11. https://doi.org/10.1145/3411764.3445641

[69] T. Stalin, S. Jain, N. K. Thanigaivel, J. E. M. Teoh, P. M. Aby Raj, and P. Valdivia Y Alvarado. 2021. Automated Fiber Embedding for Soft Mechatronic Components. *IEEE Robotics and Automation Letters* 6, 2 (April 2021), 4071–4078. https://doi.org/10.1109/LRA.2021.3067244

[70] T. Stalin, N.K. Thanigaivel, V.S. Joseph, and P. Valdiviay Alvarado. 2019. Automated Fiber Embedding for Tailoring Mechanical and Functional Properties of Soft Robot Components. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, Seoul, South Korea, 762–767. https://doi.org/10.1109/ROBOSOFT.2019.8722752

[71] L. Justin Stiltner, Amelia M. Elliott, and Christopher B. Williams. 2011. A Method for Creating Actuated Joints via Fiber Embedding in a Polyjet 3D Printing Process. In *22nd Annual International Solid Freeform Fabrication Symposium*. University of Texas at Austin, Austin, Texas, USA, 583–592.

[72] 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 Threading Method for Multi-material, Multi-form and Multi-use Prototyping. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–12.

[73] Wenhuan Sun, Adam Feinberg, and Victoria Webster-Wood. 2022. Continuous Fiber Extruder for Desktop 3D Printers toward Long Fiber Embedded Hydrogel 3D Printing. *HardwareX* 11 (April 2022), e00297. https://doi.org/10.1016/j.ohx.2022.e00297

[74] Haruki Takahashi and Jeeeun Kim. 2019. 3D Printed Fabric: Techniques for Design and 3D Weaving Programmable Textiles. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. Association for Computing Machinery, New York, NY, USA, 43–51. https://doi.org/10.1145/3332165.3347896

[75] J. F. Veneman, R. Ekkelenkamp, R. Kruidhof, F. C.T. van der Helm, and H. van der Kooij. 2006. A Series Elastic- and Bowden-Cable-Based Actuation System for Use as Torque Actuator in Exoskeleton-Type Robots. *The International Journal of Robotics Research* 25, 3 (March 2006), 261–281. https://doi.org/10.1177/0278364906063829

[76] Anita Vogl, Patrick Parzer, Teo Babic, Joanne Leong, Alex Olwal, and Michael Haller. 2017. StretchEBand: Enabling Fabric-based Interactions through Rapid Fabrication of Textile Stretch Sensors. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2617–2627. https://doi.org/10.1145/3025453.3025938

[77] Ludwig Wilhelm Wall, Alec Jacobson, Daniel Vogel, and Oliver Schneider. 2021. Scrappy: Using Scrap Material as Infill to Make Fabrication More Sustainable. In

Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21). Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3411764.3445187

[78] Zeyu Yan, Hsuanling Lee, Liang He, and Huaishu Peng. 2023. 3D Printing Magnetophoretic Displays. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 54, 12 pages. https://doi.org/10.1145/3586183.3606804

[79] Chuncheng Yang, Xiaoyong Tian, Tengfei Liu, Yi Cao, and Dichen Li. 2017. 3D Printing for Continuous Fiber Reinforced Thermoplastic Composites: Mechanism and Performance. *Rapid Prototyping Journal* 23, 1 (Jan. 2017), 209–215. https://doi.org/10.1108/RPJ-08-2015-0098

[80] Humphrey Yang, Tate Johnson, Ke Zhong, Dinesh Patel, Gina Olson, Carmel Majidi, Mohammad Islam, and Lining Yao. 2022. ReCompFig: Designing Dynamically Reconfigurable Kinematic Devices Using Compliant Mechanisms and Tensioning Cables. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3491102.3502065

[81] Fabian Ziervogel, Lukas Boxberger, André Bucht, and Welf-Guntram Drossel. 2021. Expansion of the Fused Filament Fabrication (FFF) Process Through Wire Embedding, Automated Cutting, and Electrical Contacting. *IEEE Access* 9 (2021), 43036–43049. https://doi.org/10.1109/ACCESS.2021.3065873