

Sculptable Mesh Structures for Large-Scale Form-Finding

Jesse T. Gonzalez

Carnegie Mellon University
Pittsburgh, USA
jtgonzal@cs.cmu.edu

Yanzhen Zhang

Carnegie Mellon University
Pittsburgh, USA
18388402982@163.com

Dian Zhu

Carnegie Mellon University
Pittsburgh, USA
dianz@andrew.cmu.edu

Alice Yu

Carnegie Mellon University
Pittsburgh, USA
xinranyu@andrew.cmu.edu

Sapna Tayal

Carnegie Mellon University
Pittsburgh, USA
sapnat@andrew.cmu.edu

Nazm Furniturewala

Carnegie Mellon University
Pittsburgh, USA
nfurnitu@andrew.cmu.edu

Ziying Qi

Carnegie Mellon University
Pittsburgh, USA
ziyingq@andrew.cmu.edu

Somin Ella Moon

Carnegie Mellon University
Pittsburgh, USA
sominm@andrew.cmu.edu

Leyi Han

Carnegie Mellon University
Pittsburgh, USA
leyih@andrew.cmu.edu

Alexandra Ion

Carnegie Mellon University
Pittsburgh, USA
alexandraion@cmu.edu

Scott E. Hudson

Carnegie Mellon University
Pittsburgh, USA
scott.hudson@cs.cmu.edu

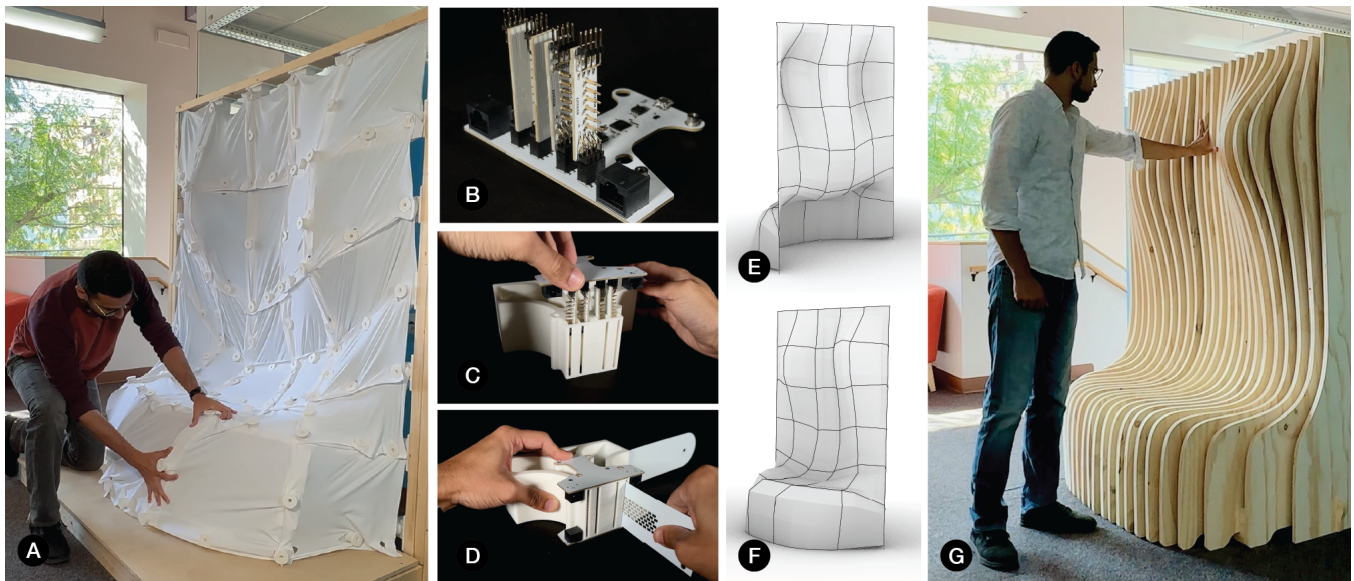


Figure 1: A large-scale mesh structure (a) composed of user-adjustable, flexible beams. A person can manipulate the structure by hand in order to sculpt the mesh into different forms. Each mesh edge is equipped with a length sensor (b-d), and the modules can transmit their configuration to an external computer, where the geometry can be reconstructed (e-f). The now-digitized geometry can serve as input to computer-controlled manufacturing processes, resulting in permanent, room-scale installations that have been "sculpted" by non-CAD users (g).

Abstract

It can be hard to design a physical structure entirely within the confines of a computer monitor. To better capture the interplay between real-world objects and a designer's work-in-progress, practitioners will often go through a sequence of low-fidelity prototypes

(paper, clay, foam) before arriving at a form that satisfies both functional and aesthetic concerns. While necessary, this model-making process can be quite time-consuming, particularly at larger scales, and the resulting geometry can be difficult to translate into a CAD environment, where it will be further refined.

This paper introduces a user-adjustable, room-scale, "shape-aware" mesh structure for low-fidelity prototyping. A user physically manipulates the mesh by lengthening and shortening the edges, altering the overall curvature and sculpting coarse forms. The edges are equipped with resistive length sensors, and transmit their configuration to a central computer. The structure can later



This work is licensed under a Creative Commons Attribution 4.0 International License.
UIST '25, Busan, Republic of Korea
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2037-6/25/09
<https://doi.org/10.1145/3746059.3747771>

be reproduced in software, connecting this prototyping stage to the larger computational design pipeline.

CCS Concepts

• **Human-centered computing** → **Human computer interaction (HCI)**.

Keywords

Digital Twin, Tangible Interface, Reconfigurable Architecture

ACM Reference Format:

Jesse T. Gonzalez, Yanzhen Zhang, Dian Zhu, Alice Yu, Sapna Tayal, Nazm Furniturewala, Ziyang Qi, Somin Ella Moon, Leyi Han, Alexandra Ion, and Scott E. Hudson. 2025. Sculptable Mesh Structures for Large-Scale Form-Finding. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*, September 28–October 01, 2025, Busan, Republic of Korea. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3746059.3747771>

1 Introduction

From architecture to engineering, computers allow us to construct physical forms that define our environments and captivate the eye. In the modern world, digital tools touch nearly every aspect of this process: from parametric modeling, to global file-sharing, to toolpath generation for computer-controlled fabrication. But the initial steps of a physical design — form-finding in particular — often benefit from a tactile approach that software cannot always provide. Designers need to assess how real-world objects will integrate with their creations with respect to both usability (“Is the luggage rack too high?” “Do I have enough leg room?”) and aesthetic quality. For this reason, practitioners will often make low-fidelity prototypes — many of them — out of foam, paper, or clay [19]. These mock-ups deliver practical insights in real-time, and in some cases (e.g. clay) allow the designer to make in-situ adjustments [38].

This hands-on step, however, is still quite siloed from the rest of the computational design pipeline. To bridge this gap, we introduce a room-scale, “shape-aware” mesh structure that can be hand-sculpted into a variety of forms — and can transmit its configuration to an accompanying software system in real-time. The structure itself is a flexible gridshell with adjustable-length members, capable of composing both single- and double-curved surfaces. It serves as a tangible interface to computer-based tools.

Interacting with our system is straightforward: a user walks up to the mesh and physically manipulates the edges and vertices, until they are satisfied with the resulting shape. Because the members are flexible (constructed from thin fiberglass panels), our mesh is well-suited for modeling curved, “organic” surfaces (Figure 2). Resistive length sensors, embedded into each member, are used to capture this new geometry, smoothly connecting it to the full power of a computational modeling pipeline.

This type of direct interaction, outside of a computer, has a number of benefits. For user researchers, it allows for rapid iteration of tangible prototypes, combined with the state-saving advantages of a digital tool. For non-expert CAD users, it opens access to surface modeling, facilitating the co-design of architectural spaces.

Concretely, our contributions are:

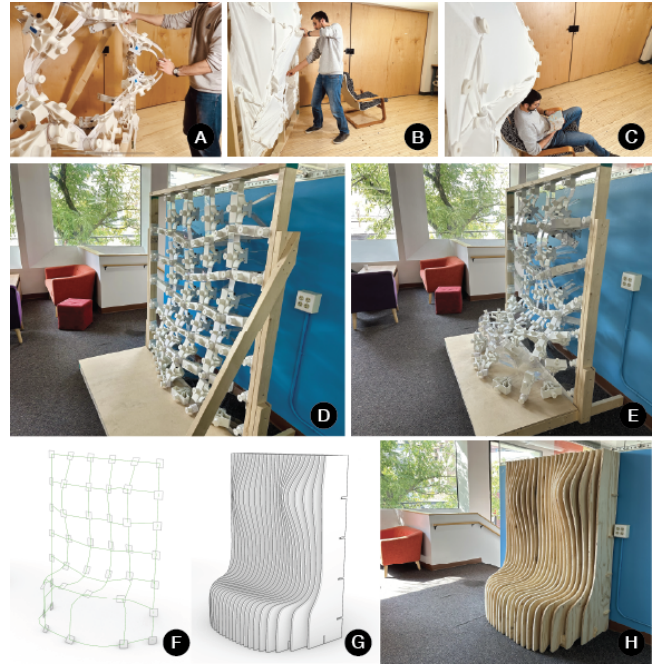


Figure 2: A user can shape our flexible gridshell structure by hand (a-c), as real-time sensor data is captured. For instance, a planar mesh (d) can be sculpted into an organic bench form (e). Since the members of this mesh are equipped with length sensors, the geometry can be reconstructed digitally (f,g) and fabricated (h).

- (1) An extensible, room-scale mesh surface composed of user-adjustable, “length-aware” members.
- (2) An energy-minimization routine for modeling bending-active structures composed of variable-length, flexible beams.
- (3) An end-to-end demonstration of the workflow that our system enables, in the context of an architectural design and construction task.

We particularly intend for our system to serve as a form-finding tool for architects and designers. A typical studio will have many tools for converting digital models into physical artifacts (laser cutters, 3D printers, etc); but the physical-to-digital workflow is less well-established. Our system adds another tool to the studio’s arsenal: allowing designers to sculpt tangible forms, evaluate the prototypes on-location, and then use the sensed geometry to construct permanent installations.

More broadly, HCI researchers have been increasingly interested in advancing tangible, shape-aware interfaces that provide direct manipulation capabilities and real-time feedback during the design process [24, 43, 45]. We advance this space by introducing a system that combines multiple deformation capabilities (bending, length adjustment, and rotation) within a single unified interface.¹

¹Implementation available at <https://github.com/jtgonz/SculptableMesh>

2 Related Work

Our approach bridges multiple research domains: design processes, tangible interfaces, and digital twins. To properly situate our work, we first examine the broader context of physical prototyping in design before addressing the specific gap our system targets.

2.1 Physical Prototyping in Design Processes

Design processes typically progress through phases of increasing fidelity and specificity [31]. Within this progression, physical prototyping serves diverse purposes that vary by design stage and objective. Early explorations may use physical artifacts to inspire ideation, while subsequent stages might employ them to test functionality [10]. These low-fidelity iterations are important for understanding spatial relationships, ergonomics, and user interactions in ways that digital models alone cannot convey [11, 25]. While architectural designers do often build smaller scale models of their designs, the complexities of human body-scale interactions with architecture [27] can be much more directly investigated with full-scale models [1].

Crucially, designers receive feedback from the properties and constraints of physical media. This material “backtalk” [35] often guides design decisions in ways difficult to replicate in purely digital environments. (For instance, when an architect manually bends a thin sheet of metal to explore a curved roof form, the physical resistance suggests natural resting states that inform the final design.) This type of embodied knowledge acquisition has been recognized as fundamental to architectural thinking [8].

We do not position our system as a replacement for all physical prototyping methods, many of which serve important purposes outside our scope. Instead, we focus specifically on the transition point between form-finding and digital capture — a persistent challenge in design workflows that has been identified as a significant bottleneck in the creative process [14].

2.2 Digital Twins and Shape-Aware Materials

Digital twins pair physical objects with virtual representations [21] — in our case, embedded sensors capture the mesh’s geometry for software reconstruction. Traditionally, digital twins are used in the latter stages of the product life cycle, to track usage patterns [22] or mechanical wear-and-tear [16]. Our work instead leverages this concept earlier in the design process, bringing real-time digital capture into the initial prototyping phase.

In the HCI community, researchers have explored similar ideas through the development of “shape-aware” materials and structures [46]. These are physical objects that detect user manipulation (i.e., extension, compression), and can stream this data to a connected computer. This can be accomplished with directly-embedded sensors [29], or by leveraging proxy objects [15] that work in conjunction with external cameras to track deformation [37].

These structures have proven valuable as creative input devices [12]. In puppetry applications [45], for example, users can control digital characters by manipulating sensor-equipped “Tinkertoy-like” modules. Advances in piezoresistive materials [5] have further expanded these possibilities, enabling sensors to detect subtle physical interactions like squeezing or bending, and update digital models

in real-time. Subtractive design approaches [47] explore how digital models can update in response to physical cutting or carving.

Particularly relevant to this work are devices designed to measure curves. ShapeTape [6, 18] is an early example: a handheld rubber strip with fiber-optic bend sensors, which a user can manipulate in order to model and capture curves in 3D space. More recent methods have leveraged flexible circuit boards with onboard IMUs [13] (to capture twist), and capacitive sensing of shifting, offset electrodes [36].

This has also led to the development of physical tools that also have a degree of spatial awareness. HandSCAPE [23] is an orientation-aware digital tape measure that captures both distance and direction, enabling users to digitize field measurements. Similarly, the SPATA tools [44] provide bidirectional transfer between physical and virtual measurements through actuated calipers and protractors.

Further leveraging this physical-digital link, researchers also have created tabletop construction kits that augment hands-on prototyping at the centimeter-scale. Systems like StrutModeling [24] and NurbsForms [43] allow users to assemble deformable physical components, which generate corresponding digital models.

Architectural design, however, often benefits from whole-body interaction and collaborative shaping. Our work scales up shape-aware principles to room-sized surfaces — addressing challenges in sensing and mechanical design to enable multiple users to physically sculpt and evaluate their creations at room-scale. This allows for the kind of intuitive, hands-on exploration that is crucial for processes such as participatory design.

2.3 Room-Scale Physical Interfaces

Within HCI, several approaches have emerged for quickly prototyping large physical forms. Using the handheld Prototyper extruder [4], for instance, a user can sketch physical wireframes of furniture-scale objects in 3D space. At larger scales, similar truss-like structures can be automatically deployed, from inflatable tubes that are either heat-sealed [42] or tied in real-time [30]. Researchers have also explored more modular approaches — assemblies of electronically integrated structures using both rigid pipes [48] and polyhedral voxels [39].

Another direction focuses on reconfigurable architectural surfaces that can dynamically assume different functional forms [17, 20, 41]. Though not explicitly designed to support direct manipulation by users, these systems demonstrate compelling methods of “rendering” low-fidelity forms in physical space. If coupled with an approachable user interface, such systems could be powerful tools for in-situ reshaping of environments.

2.4 Participatory Design in Architecture

Participatory design actively involves stakeholders in the design process [32]. Instead of designing *for* users, architects using participatory techniques will design *with* users, acknowledging that communities have unique insight into their own needs and daily patterns of life [26].

When spaces are designed this way, communities often feel greater ownership over the final result [33]. This collaborative approach can strengthen the connection between people and place.

It has been shown to be an especially effective way of designing schools [33], housing developments [2], and public areas [34].

The success of participatory design hinges on clear communication between architects and community participants. Here, physical prototypes serve as crucial tools. While architectural drawings and digital renderings can be challenging for non-professionals to interpret, physical prototypes provide an intuitive medium for spatial exploration. These tools bridge the expertise gap between architects and community members [9]—helping to focus discussions and test design ideas [31].

Particularly significant are “experiential prototypes”—physical models that allow people to directly interact with and experience spatial concepts [40]. Full-scale mock-ups enable users to directly experience spatial relationships, helping them evaluate factors like ceiling heights, circulation paths, and furniture arrangements that might be unclear in traditional drawings [32].

By providing a tangible interface for spatial design while simultaneously capturing digital information, our system addresses a specific challenge in participatory design: how to involve community members meaningfully while maintaining a connection to the computational design pipeline that dominates modern architectural practice. This integration of physical exploration with digital design tools has the potential to enhance the participatory design process, making it more accessible and effective for everyone involved.

3 Room-Scale Form-Finding with an Interactive Mesh

Our large-scale, interactive gridshell is shown in Figure 2. This is a mesh-like structure composed of bendable fiberglass members, whose lengths can be manually adjusted by a human operator. By manipulating these members, users can “sculpt” the mesh, transforming it from an initial configuration (i.e. a flat plane) to a new one, analogous to how an artisan molds a block of clay. Though not load-bearing, the resulting shapes are stable, serving as a physical wireframe that designers can use to evaluate their forms. Onboard sensors detect the lengths of these wireframe edges, allowing a paired software system to capture and reconstruct the sculpted geometry for later use.

We envision an interaction workflow that follows the general pattern below:

- (1) First, the user chooses an initial topology. In our examples, this is a quad mesh, bounded on three sides by a rigid frame.
- (2) Then, the user physically adjusts edges and vertices, while the accompanying software periodically saves the intermediate forms.
- (3) Finally, the user leverages the digitized form to augment their design process. They may choose to review “snapshots” of their sculpted geometries, or import the digital mesh into a CAD tool for further processing.

We detail this workflow in the following section, where we explore how a group of architects might use our tool to complete a small design and construction task.

3.1 From Physical, to Digital, to Physical Again

Consider the following example: a local library is undergoing a large renovation, and the architects involved would like to engage



Figure 3: A person, inspired by the sunlight from a nearby window, sculpts a couple shelves to hold plants. The mesh holds its form and can support lightweight objects (such as the plants in b and d) without additional reinforcements.



Figure 4: One person approaches the mesh wall and sculpts a child-sized alcove (a,b). A second person sits inside and modifies the alcove. She pulls the mesh around her so as to cover herself more fully (c), in an attempt to create a cozier environment (d).

the community in the design process. They bring a few participants into the building, and direct them to an area with a large mesh wall. Here, the participants are instructed to shape the space into a structure that suits their needs, experimenting with different forms.

Looking towards a nearby window, one participant notes that this site receives a large amount of sunlight. They think it would be great spot to keep some plants, and begin shaping the mesh accordingly (Figure 3). First, they pull out a section of the mesh and flatten it, turning it into a shelf. Then, inspired by the manner in which the plant leaves hang, they sculpt a second, lower shelf beside it. They choose to have this shelf taper off, so as not to obstruct the nearby stairway. Because this low-fidelity prototyping is taking place on-site, the participants can use features of surrounding environment (e.g. windows, sunlight, stairway) to stimulate and inform their design.



Figure 5: In its default state, our sculptable mesh surface is a flat plane (a). A person can pull on the edges and vertices to create forms such as a bench (b,c). They can use physical items, such as a stool (f) to interact with their creations. In this example, a person notices that their sculpted bench is uncomfortable (d), and so they make a modification (e).

Another participant decides to sculpt a small alcove, reasoning that children might enjoy hiding within the nook (Figure 4). After an initial "draft", other participants sit inside, testing the fit. One participant feels that the nook is too shallow, so they drag a piece of the mesh outward, enveloping themselves for a cozier atmosphere (Figure 4c). The architects take note of this and save the configuration digitally.

The site is near the entrance of the library. Some participants point out that they often see people standing in this area, waiting for friends or family to arrive. They suggest that some additional seating could be useful here, and begin to sculpt a bench (Figure 5). The participants also add a wave-like pattern to the wall, which they think compliments the shape of the seat.

The mesh itself is not meant to be load-bearing, so to test the bench, one participant grabs a nearby stool and places it inside of the mesh structure (Figure 5f). When they sit down, they discover that the sculpted wave bulges out too far, and is uncomfortable. Though it appeared fine at first, physical testing made this design flaw apparent. The participant pushes the bulge inward, creating a cavity that they can lean back against (Figure 5e). They note that this would be a nice spot to sit and read a book. Ultimately, the participants conclude that — this being a library — the reading bench is the best choice for the space.

3.2 End-to-End Workflow

With the community participants having finalized the bench design through physical sculpting and testing, the architects now proceed to transform this temporary prototype into a permanent installation. This process demonstrates how our system bridges the gap between intuitive physical prototyping and precise digital fabrication.

Since each edge of the sculpted mesh is equipped with length sensors, the architects can capture the exact geometry through a software interface connected to the mesh. Length readings for all modules are continuously streamed to Rhino CAD software via this interface. Within seconds, the physical adjustments that the participants made (the height of the bench, the waves on the wall, the sculpted cavity) are preserved in a digital wireframe model.

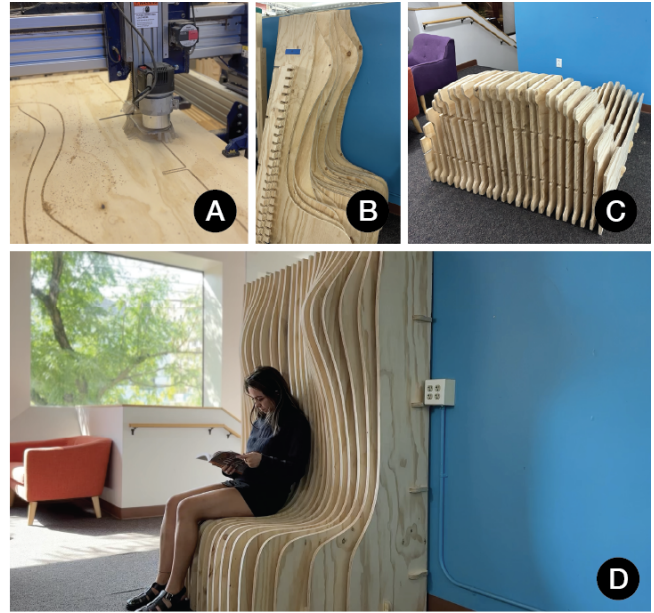


Figure 6: Using data from the length sensors, the sculpted bench is reconstructed in software, and can be connected to the rest of the digital manufacturing pipeline. In this example, the digital mesh is used to create CNC toolpaths (a), which then results in a permanent installation (d).

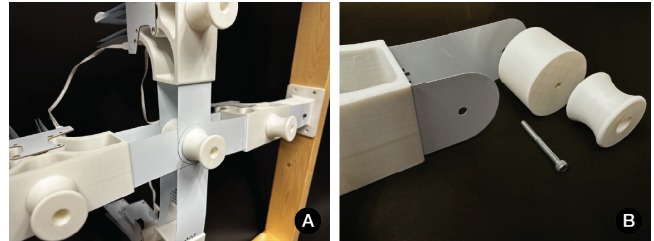


Figure 7: Modules can be joined together at the vertices (a) by using passive mechanical fasteners (b).

The reconstruction algorithm (detailed in Section 6) processes this wireframe, first by applying energy minimization to establish the correct form, then by fitting cubic splines along each member and generating smooth Coons patches between them. This results in a continuous NURBS surface that captures the essential ergonomic features while creating a clean, fabrication-ready model (Figure 2g).

For fabrication, the architects decide to slice the bench into panels that can be cut from standard plywood sheets — a common fabrication strategy for complex curved forms. Using contour tools in Rhino, they generate evenly spaced vertical sections that, when stacked, will recreate the 3D form. These contours are arranged to maximize material efficiency and include alignment features to ensure accurate assembly.

The prepared profiles are exported as vector files and imported into CAM software for toolpath generation (Figure 6a). After the

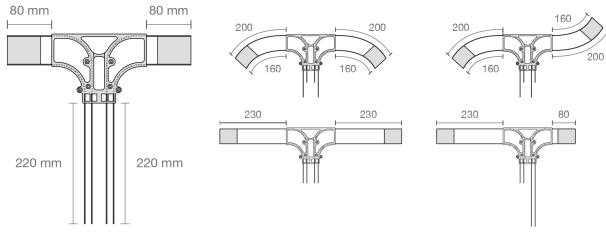


Figure 8: The “T-Module” is the unit cell of our sculptable mesh structure, acting as a variable-length edge. It consists of flexible bands routed through a rigid guide. When fully contracted, the module measures 22 cm from vertex to vertex. When fully expanded, it reaches 56 cm. Excess material is directed towards the interior of the mesh, away from the user.

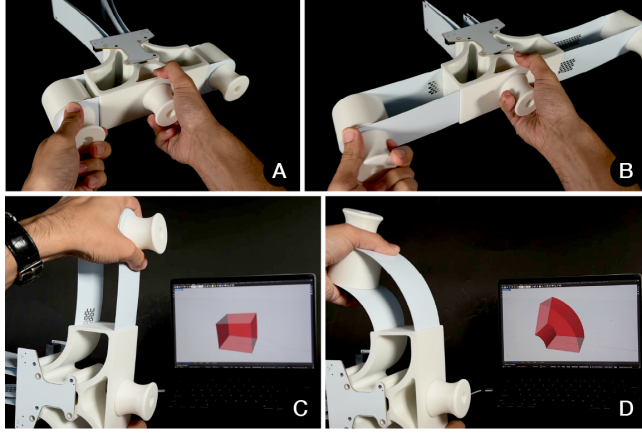


Figure 9: A user can alter the length and curvature of the T-Module by pulling on its handles (a,b). Each side of the module can be adjusted individually. Embedded length sensors capture the module geometry (c,d). By composing many of these members together, we can create sculptable forms that have an inherent digital twin.

panels are cut, they are sanded, assembled according to their designed sequence, and press-fit together (Figure 6b,c).

The completed bench is installed in the exact location where the initial prototyping took place (Figure 6d). The library now has a permanent installation that was collaboratively sculpted by stakeholders in the community — a direct physical manifestation of their design input.

4 Overview and Mechanical Design

The T-shaped structure shown in Figure 8 is the “unit cell” of our sculptable mesh, and we refer to it as a “T-Module”. It is an adjustable-length beam, made of four flexible FR4 (fiberglass) bands which can slide freely through a plastic guide. A user can pull on these bands to expand a particular edge (Figure 9b), or slide them inward to shrink the edge, tucking the excess material away (Figure 9a).

These modules can be linked together to form a mesh, as shown in Figure 7. Mechanically, the T-Modules are connected with simple fasteners (plastic cylinders, with a metal bolt as an axle). Electrically, they are networked via flexible patch cables in a winged-edge configuration (see Section 5).

Individually, the FR4 bands of the T-Module are flexible enough to pass through the plastic guide without much resistance, but together (in a “two-tier” configuration), they form a prismatic structure with enough rigidity to keep the parent mesh from collapsing. Note that for the curved forms in Figure 8, there is a length differential between the inner and outer bands, which allows us to detect the direction of curvature (Figure 9c,d).

The four FR4 strips are circuit boards², 0.8 mm thick. By adding these boards, we transform this passive module into a “length-aware” mesh edge. Electrically, each circuit operates much like a slide potentiometer. The flexible boards are patterned with a resistive code, which, when read by a scanning head at the “neck” of the module, can be mapped to a physical displacement. A microcontroller, connected to this scanning head, records this length data and transmits it to an external computer. (This is done via neighbor-to-neighbor communication within the mesh — the message is passed from module to module until it reaches a pre-selected relay unit). The strips contain no onboard power source — instead, the scanning head sends a small current through the strip’s resistor ladder, and takes a reading.

The modules are joined mechanically through passive vertices — plastic cylinders with a metal axle. They are joined electrically through flexible, Cat6 patch cables, in a winged edge configuration. Any individual module can serve as a relay device, collecting length data from the parent mesh and transmitting it to our software. (In practice, we choose a module near the outer perimeter to perform this function).

4.1 Curvature and Connection Scheme

Our mesh structures mimic a biaxial weave (Figure 10), with four edges meeting at every interior vertex. Compared to other patterns, this configuration gives us a good balance between user manipulability and structural stability. We found that denser structures, such as a valence-6 triangular mesh, were stiff near the vertices and difficult to physically adjust. Sparser structures, such as a valence-3 hexagonal mesh, too often fell into configurations that were underconstrained. A valence-4 configuration gave us a good balance between these alternatives.

Our mesh structure is able to transition smoothly between single and double curvature (Figure 11). By adjusting the length of the flexible members, the surface can “stretch” in ways that would normally cause kinks or distortions in conventional materials. This allows us to adapt to both single-curved and double-curved surfaces, without compromising the continuity of the form.

Though quad meshes are theoretically susceptible to shearing, we did not experience noticeable issues in our prototypes. This assumes that the boundaries of the mesh are anchored to fixed points, which provides sufficient constraint to prevent unwanted deformations.

²Fabricated externally, via JLCPCB.

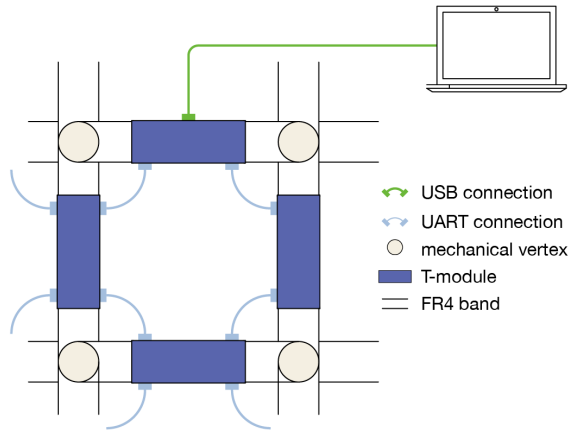


Figure 10: Connection scheme for our sculptable mesh surface. T-Modules are joined to their neighbors using Cat6 cables, and exchange messages via a UART peripheral. One T-Module is connected to a computer, where it transmits the mesh edge lengths to our reconstruction software.

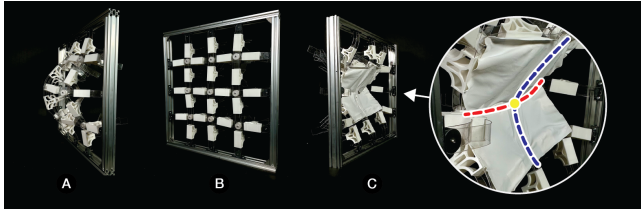


Figure 11: By expanding, contracting, and bending modules, our mesh can transition from a state of zero curvature (b) to a state of either positive curvature (a) or negative curvature (c). We have attached a piece of fabric to the modules in (c) to make the saddle point more apparent.

4.2 Materials and Fabrication

At the core of each module is a 3D-printed, PLA plastic guide. It contains four curved slots, each of which constrain a thin strip of FR4 at a bend radius of 50 mm. This radius was chosen to mitigate plastic deformation — at smaller radii, this deformation becomes noticeable by eye, and introduces additional stresses into the system (making it difficult to model in software). The strips themselves are 60 mm wide and 0.8 mm thick. The inner and outer strips are spaced 50 mm apart.

An early engineering challenge was optimizing the stiffness of the T-Modules. Overly stiff beams limit our ability to form curved surfaces; but if the beams are too flexible, the sculpted mesh won't hold its shape. Stiffness also impacts the extension ratio of the modules — as the strips pass through the rigid guide, stiffer strips must do so at a larger bend radius. A larger bend radius requires a longer plastic guide, which ultimately reduces the overall extension ratio.

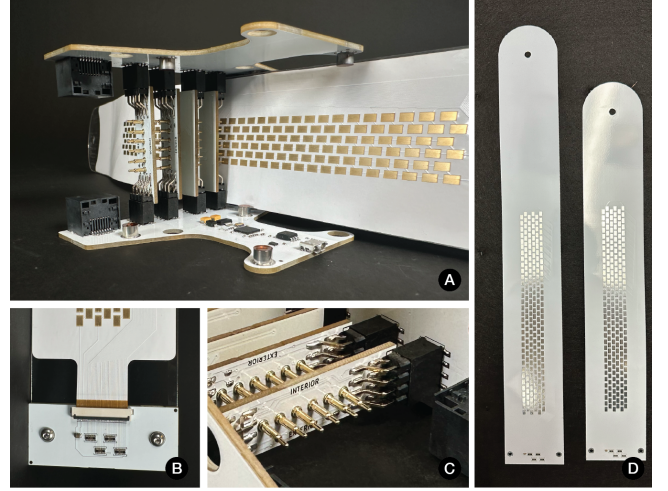


Figure 12: Interior of T-Module (a), which uses flexible FR4 strips (d) as length sensors. The resistive code is read by a scanning head (c). Earlier prototypes used flexible printed circuit boards, attached to a rigid resistor bank (b).

The "two-tier" prismatic structure helps solve this problem: instead of directly optimizing the flexibility of the beam, we can optimize the distance between the outer and inner strips. This allows us to experiment with different stiffness levels while using strips of a constant thickness — a larger distance between strips corresponds to a stiffer beam.

Additionally, using two thin bands instead of a single thick band permits a smaller bend radius at the rigid guide, which in turn increases the dynamic range of the adjustable edge.

For ease of manipulation, we attach plastic knobs to the both the T-modules and mechanical vertices. Once a user adjusts the length of the beam, static friction keeps it in place. (We experimented with mechanical locking mechanisms for the beams, but found friction sufficient for furniture-scale modeling.)

4.3 Topology and Anchors

In our current prototype, the user must provide a starting topology — in the form of an adjacency matrix — before we can reconstruct the mesh. The user also specifies which (if any) vertices should act as "anchors", along with their starting positions. For instance, in the mesh shown in Section 2, the user must specify that the 16 vertices along the top and sides of the frame are anchor points. In the current prototype, the user supplies this information via a C# script. In future versions it should not be hard to do this with a simple interface.

5 Electronics and Firmware

Every active module in our mesh structure is equipped with a small microcontroller³, which allows for local length sensing and neighbor-to-neighbor communication. Adjacent edges are connected by flexible cables, which transmit both power and data,

³RP2040, by Raspberry Pi Ltd.

allowing us to interface with the system via a single external connection.

Figure 12 shows an interior view of the module. Within, there are four components of interest:

- (1) A faceplate, housing the microcontroller, sensing circuitry, a USB interface, and two power/communication ports.
- (2) A backplate, providing two additional power/communication ports.
- (3) Two pairs of flexible circuits with a resistive ladder, each on an 0.8mm thick FR4 panel.
- (4) Two pairs of scanning heads, which read a resistive code from the flexible strips. These also facilitate power and data transmission between the faceplate and backplate.

Note that while only one pair of strips (and scanning heads) are required for measuring the module length, an additional pair helps us determine the direction of curvature.

5.1 Length Sensing

When our software system first connects to the physical mesh, the length of each edge is unknown. This situation is similar to many machines found in an architecture studio, such as laser cutters or 3D printers, which address the issue by performing a homing sequence upon startup. However, unlike those machines, we cannot expect the user to manually "zero" each variable member before beginning the sculpting process — this would be far too tedious. As a result, absolute positioning becomes a critical requirement.

We need to sense changes in length along each strip, detecting variations of up to 200 mm. Commercial sensors that can achieve this over such a range are either difficult to find or prohibitively expensive.

To meet this need, we implement a resistive ladder on each FR4 strip. The ladder itself is a chain of discrete 100-ohm resistors⁴, connected in series at the bottom edge of the strip. Each connection point between resistors is then routed via a copper trace to an exposed pad positioned along the length of the strip.

A scanning head with spring-loaded probes maintains contact with these copper pads. It operates much much like a slide potentiometer, measuring the voltage at whichever copper pad it currently contacts (Figure 14a). Since each pad corresponds to a specific position along the strip, the measured voltage directly indicates the module's length. This offers an inexpensive method for absolute length sensing, as the resistor ladder can be patterned on a flexible substrate (e.g., thin FR4 or polyimide) over significant distances.

However, using only a single probe can cause issues when the scanning head crosses between pads (Figure 14b), as the probe is left floating. This ambiguity may be acceptable for incremental sensors, but it is insufficient for applications requiring absolute positioning, since the probe may start in an undefined state.

We resolve this by introducing a second set of conductive pads, slightly offset from the first (Figure 14b). In this staggered configuration, at least one probe is always in contact with a pad. This has the added benefit of increasing our resolution by a factor of four. With pad lengths of 6 mm, our resolution improves to 2 mm across a 200 mm active area.

⁴We found that this gave us a good balance between noise resistance and current draw.

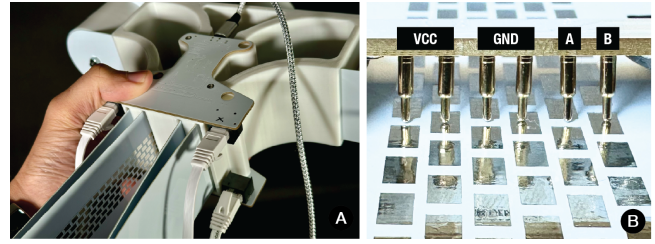


Figure 13: We detect module curvature by sensing the length of resistive strips (b). Having two probes on the scanning head allows us to avoid dead zones on the resistive ladder (see Figure 13a).

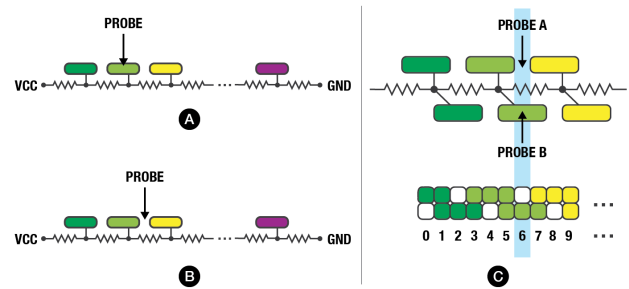


Figure 14: On the left (a,b) we show a standard resistive ladder; on the right (c) we show our configuration. Each node on our resistive ladder is connected to two copper pads, which are offset. This configuration ensures that there are no “dead zones” on the strip — areas between pads where the pins of the scanning head are not in contact with any part of the resistive ladder.

5.2 Curvature Detection

With sensors on only the outer strips, we can detect edge length, but not the direction of curvature. For some scenarios, this is sufficient (for instance, if we know that the mesh is against a wall, we can assume that most edges will bend outward when lengthened, and then bias the simulation accordingly). However, in more underconstrained scenarios, we need to use sensors on both the top and bottom strips — which our system does indeed support.

With two probes per strip, we require a total of eight ADC channels to sense the length on all four strips. This actually exceeds the amount available on the RP2040 (which is limited to four), so we use an analog switch (Vishay DG2788A) to de-multiplex the inputs. By comparing the inner and outer strip lengths, we can determine which way the module bends.

5.3 Power and Communication

On the underside of each T-Module, there are four physical ports (Figure 12a) for transmitting power and data between adjacent units. These connections are made through flexible Cat6 cables — although we use UART, rather than Ethernet, to communicate. Each unit passes messages directly to its neighbors, which allows us to easily extend and reconfigure the mesh network.

The RP2040 only features two hardware UARTs, but up to four additional interfaces can be implemented by using the onboard PIO state machines. We leverage all of them, and create four full duplex UART peripherals, so that each T-Module can communicate with its immediate neighbors. These four ports are sufficient, since no module will have more than four neighbors (as we explain in the following section).

Power can be injected at any point in the structure, or at multiple points. This approach helps distribute the load in larger configurations, reducing the current required through any individual cable and minimizing potential voltage drops across the system.

5.4 Mesh Representation and Traversal

In our implementation, neighboring modules are defined as mesh edges that (1) share a vertex and (2) belong to the same planar face. This definition allows us to efficiently represent our mesh using a winged-edge data structure [7]. In this structure, each edge stores references not only to its two endpoints, but also to the two adjacent faces and the four edges that are incident to them (Figure 15a).

In contrast to other mesh representations (e.g. face-vertex or vertex-vertex), this representation provides several advantages for our system. First, it facilitates traversal of the mesh using only information encoded in the edges — important because the edges are the only active elements, capable of sending and receiving messages. Second, this structure supports efficient updates to the mesh topology, which is crucial as edges are adjusted during the sculpting process, enabling real-time interaction with the physical mesh.

One module sends a message to another by attaching an address — a list of hops — to a data packet and passing it to a neighboring module. Each module along the path tracks the number of hops and forwards the message to the next neighbor specified in the address list (Figure 15b). As the message passes through, each module also notes the port through which it received the message and appends this information to the packet, effectively creating a return address. When the target module receives the message, it can reply to the original sender by following the return address generated during the message’s journey.

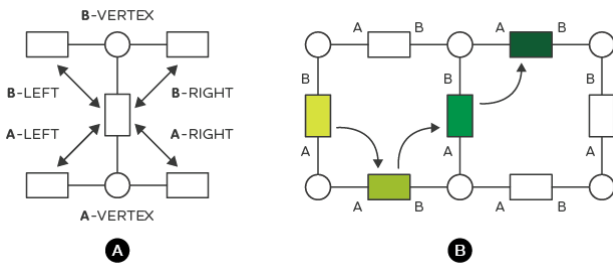


Figure 15: Our mesh can be represented by a winged edge data structure (a). Modules can send messages to each other by passing data from neighbor to neighbor (b). Along this path, modules construct a return address, so the receiver module can send a message back to the sender .

5.5 Physical-to-Digital Link

To ensure smooth interaction between the physical mesh and its digital twin, we must establish a direct connection between the two systems. This process involves linking each physical module to the corresponding edge in the software model, allowing real-time communication and synchronization between the virtual and physical representations of the structure.

First, the user defines the mesh topology in software using a C# script. Next, we need to align the physical mesh with the software-defined topology, meaning each physical module must be mapped to a corresponding edge in the software

For all of our applications, we choose one module to serve as the “relay unit”, and connect it to a central computer via USB. With this module as the root, we first perform a breadth-first traversal of the mesh, obtaining the IDs and addresses (routes relative to the root) of all connected units. After this initialization, the central computer begins polling each module, requesting sensor data. Each polled module responds by sending its ID and length data to the relay unit.

With the physical and digital meshes now synchronized, the user can manipulate the length of each edge in the physical structure, and these changes will be reflected within the digital model.

6 Digital Reconstruction

Once the lengths of each mesh member are known, we can reconstruct the overall geometry to create a digital twin of the physical structure. We approach this as a dynamic relaxation problem, using the Kangaroo physics engine [28] within Rhino 8⁵ to calculate and minimize the residual energy in our mesh structure.

For the mesh wall demonstrated in Section 2 (consisting of 54 T-Modules), our simulation creates approximately 300 particles. These particles correspond to physical points on the sculpted structure, and their positions are optimized through energy minimization.

At each iteration of our solver, a weighted sum of energies acting on each particle is calculated, a correction vector (pointing towards the lowest energy state) is constructed, and the particles are moved accordingly. This process continues until the system reaches a stable configuration that best matches the sensed physical state.

6.1 Complete Reconstruction Pipeline

Our digital reconstruction follows a multi-stage process:

- (1) First, we initialize the mesh topology with known anchor positions (e.g. the wooden frame in which the physical mesh is suspended).
- (2) We then perform a coarse solving pass using only length constraints between vertices, establishing the basic shape.
- (3) Once this coarse mesh is established, we interpolate intermediate points along each member to represent the full T-Module geometry.
- (4) Next, we add additional energy terms to refine the curvature and mechanical behavior of the mesh.
- (5) Finally, we iteratively solve until the mesh represents a physically plausible configuration that satisfies our measured constraints.

⁵Rhino 8 CAD software available at <https://www.rhino3d.com/>

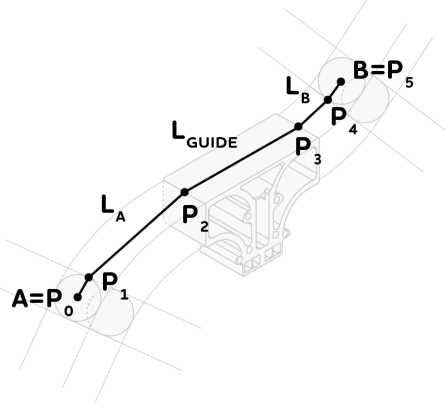


Figure 16: In our simulation, each T-Module is represented by a polyline consisting of six points. The points correspond to particles in our solver.

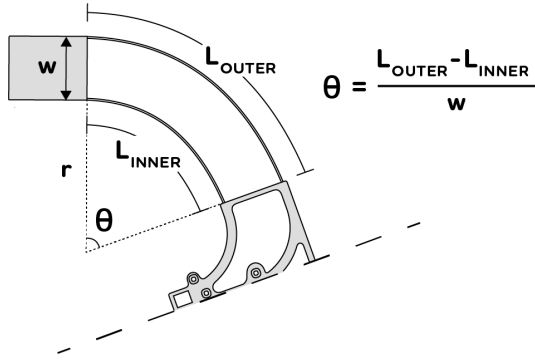


Figure 17: We can compare the measured lengths of the inner and outer bands to detect the direction and amount of curvature.

This approach allows us to capture the general physical form, preserving important distances relevant to human users (e.g., depth and width of a seat, height of an overhang).

6.2 Geometric Representation

We represent each adjustable-length member with a polyline consisting of both fixed and variable-length segments (Figure 16). Six points define this polyline, each mapping to a particle in our simulation.

The two endpoints, p_0 and p_5 , correspond to the "A" and "B" vertices of the T-module. Two intermediate points, p_1 and p_4 , are placed on the perimeters of the mechanical vertices (the plastic cylinders where T-modules are joined together). The remaining two points, p_2 and p_3 , represent the edges of the rigid guide.

In our simulation, particles are shared between mesh members. Specifically, this occurs at the "A" and "B" vertices. For example, p_5 for one mesh member may be p_0 for another. Our solver does

not distinguish between these shared particles and simply sums all energies acting on each one, resulting in a cohesive mesh structure.

6.3 Energy Terms

Our simulation incorporates several types of energy terms that collectively determine the final configuration:

- (1) *Length Energies*. These act like springs between vertices, pushing or pulling particles until the distance between them matches the sensed physical length.
- (2) *Coplanarity Energies*. These maintain local smoothness at mechanical vertices by ensuring surrounding points lie approximately in the same plane.
- (3) *Bending Energies*. These model the physical stiffness of the FR4 strips, restricting how sharply the mesh can curve.
- (4) *Anchor Constraints*. These fix certain points in space, typically along the boundary of the mesh.
- (5) *Bias Energies*. These help resolve bistable configurations by pushing vertices in the correct curvature direction.

These energies work in concert, with length energies primarily determining the overall shape, while the others refine local behavior to match physical constraints.

6.4 Handling Bistability

When attempting to reconstruct the mesh from length measurements alone, we encounter a fundamental ambiguity: knowing only the distance between endpoints does not tell us whether a curved edge bends inward or outward. This bistability creates a challenge for our reconstruction algorithm.

To address this, we begin with a coarse solver that considers only "length energies" and optimizes just the particles corresponding to the "A" and "B" vertices. These length energies function like springs — when the distance between p_0 and p_5 equals the sensed, real-world distance, the particles exist in a zero-energy state. Otherwise, a correction vector guides the particles toward the correct distance.

However, with length constraints alone, the solver may converge to configurations where areas that should be convex appear concave, or vice versa. When a T-Module is equipped with all four length sensors (both inner and outer bands), we can determine the direction of curvature by comparing these readings (Figure 16). When the outer band is longer than the inner band, we know the module curves away from the outer band.

Once the curvature direction is determined, we apply a constant-magnitude bias force in that direction. This force is "artificial" in that it doesn't model a physical phenomenon but instead helps guide the solver toward the correct solution by breaking the symmetry of bistable configurations. The magnitude of this force remains constant regardless of how much curvature is detected—it merely indicates the correct direction.

After the coarse solver has converged, points p_1 through p_4 are interpolated between p_0 and p_5 as an initial guess for their final positions. We then proceed to reconstruct the curvature by incorporating additional energy terms.

6.5 Local Coplanarity

The mechanical vertices of our mesh structure are rigid cylinders, 6 cm in diameter. This physical constraint requires that all points

along the circumference of each vertex remain approximately coplanar. We enforce this with a "coplanarity energy" term in our simulation.

For each vertex, we gather the four surrounding particles and fit a plane to all five points (the vertex itself plus its four neighbors). The best-fit plane is calculated using standard least-squares methods. For each of these points, we generate a correction vector perpendicular to the plane, with magnitude equal to the point's distance from the plane.

When applied to our simulation, this coplanarity constraint has the effect of smoothing the sharp features that may result from the coarse solver. It also effectively "clamps" the ends of the flexible bands, constraining the bending energy that we add in the next section and better reflecting the physical behavior of our T-Modules.

6.6 Axial Bending

For every trio of adjacent points in our polyline representation, we calculate the bending stresses using the method described by Adriessens and Barnes [3]. This approach models each segment as a rod with internal bending resistance, yielding a shearing transformation that, when applied, drives the member towards its natural curvature.

This method, however, assumes a circular cross-section with a uniform moment of inertia — an assumption not met by our system. To account for this, we constrain the bending to occur only in the plane where the physical module can actually flex, eliminating forces that would attempt to bend the module in physically impossible ways. Specifically, we project the particle correction

vectors onto the plane defined by p_0, p_5 , and the average of the vertex normals at those points. While this adaptation simplifies the true mechanical behavior, it captures the essential characteristics needed for our reconstruction.

6.7 Post-Processing for Final Form

Once the energy minimization simulation converges to a stable configuration that satisfies our measured constraints, we apply several post-processing steps to create a smoother, more usable digital model.

First, we fit natural cubic splines through the polyline representations of each member. This smooths any remaining discontinuities in the simulated mesh while preserving the overall shape and measured dimensions.

After creating splines for all members, we generate Coons patches between adjacent splines. These surface patches interpolate between the boundary curves, creating smooth surface transitions across the entire mesh structure. Finally, the individual Coons patches are joined together to create a unified surface representation of the sculpted form.

This post-processed model can then be directly imported into CAD software for further refinement or manufacturing preparation, as demonstrated in the library bench example in Section 2.

7 Discussion and Future Work

Our shape-aware mesh system enables low-fidelity form-finding through tangible interaction, serving as an important link between hands-on prototyping and the digital manufacturing pipeline. While our current implementation demonstrates the potential of this approach, there are several areas where future work could enhance functionality and expand applications.

7.1 Limitations

Perhaps the most visible limitation of our system is resolution. While our tool is well-suited for room-scale applications (e.g., benches, alcoves, facades), more detailed forms may demand a larger number of small-scale modules. The current module size creates a tradeoff between workspace coverage and feature resolution — a challenge when designing objects that contain both broad surfaces and fine details.

Another consideration is the material properties of the mesh. Although the flexible FR4 panels used in our structure are well-suited for modeling curved surfaces, they are subject to small amounts of plastic deformation when held in place over time. This did not noticeably affect the shapes we were able to sculpt during our testing, but it is worth noting as a potential long-term failure point, particularly for installations that might remain in a fixed configuration for extended periods.

Finally, the accuracy of our reconstruction routine is contingent on several assumptions about material behavior. In practice, small inconsistencies in the flexibility of the members, or external forces like gravity, may cause deviations from the predicted shapes.

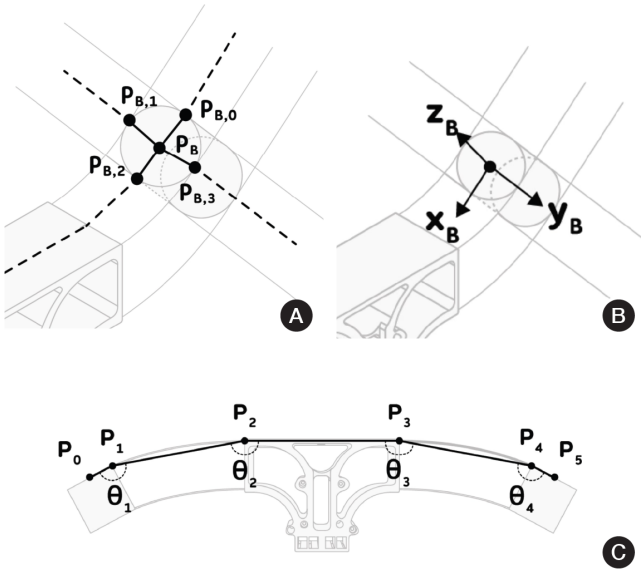


Figure 18: (a) Vertex particles, along with their four immediate neighboring particles, are softly constrained to be coplanar. (b) A reference frame is constructed at each module vertex. (c) A bending energy attempts to minimize the angle between adjacent line segments.

7.2 Comparison with Existing Digitization Techniques

Unlike real-time 3D scanning approaches that capture surface geometry through point clouds, our shape-aware mesh provides direct structural information through edge length measurements. This distinction offers several advantages: our system eliminates occlusion issues common in optical scanning, requires no post-processing to clean up point cloud data, and seamlessly integrates with CAD workflows through a direct data pipeline. Additionally, while scanning approaches passively record geometry, our system actively captures designer intent through direct manipulation of structural elements, creating a more deliberate bridge between physical intuition and digital representation.

Furthermore, traditional form-finding techniques typically operate as one-way processes, either moving from physical models to digital representations (through scanning) or from digital designs to physical models (through fabrication). Our approach establishes a continuous feedback loop where physical manipulations can be immediately visualized digitally, evaluated, and refined. This bidirectional relationship transforms the mesh from a simple digitization tool into an interactive medium that combines the tactile benefits of physical prototyping with the precision and iterative capabilities of digital design.

7.3 Future Directions in Hardware and Sensing

To address the resolution limitations, future implementations could incorporate modules of different sizes within the same structure — allowing for higher-resolution patches alongside coarser features. This multi-scale approach would enable designers to focus on detail where needed, while maintaining manageable complexity in areas requiring less definition.

Alternate materials (with better fatigue resistance) could be explored as well. Composites that combine the resistive properties of our FR4 strips with improved mechanical characteristics might offer an ideal balance between sensing capability and physical behavior.

Lastly, the neighbor-to-neighbor communication in our mesh could potentially introduce latency for structures that are larger than room-scale (as the number of hops increases). A potential solution is to introduce localized hubs that all share a communication bus, and then hop from hub to hub instead of module to module.

7.4 Topologies and Configurations

Although our current implementation requires users to predefine the mesh topology, our system is also capable of supporting automatic topology discovery. By leveraging the communication capabilities of the modules, each unit is able to identify its connected neighbors by exchanging messages and recording connection points. Future implementations can leverage this existing feature — dynamically constructing the mesh topology without manual input, and streamlining the user experience.

Beyond flat meshes, our system can support various starting geometries, such as the cylindrical configurations in Figure 18. Future work could explore more complex initial topologies: spheres, toroidal surfaces, or other structures — expanding the design space for architectural applications.

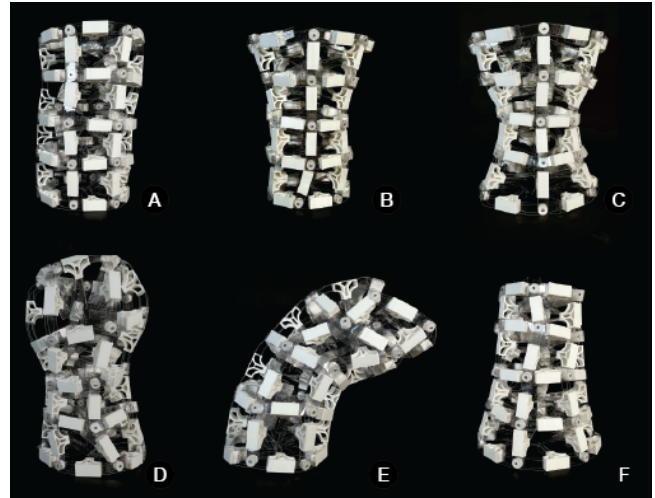


Figure 19: Our modules are capable of starting from geometries other than flat planes. Here we explore various forms based on a cylinder.

7.5 Bidirectional Information Flow

In this work, we have focused primarily on the flow of information from the physical to the digital. Moving forward, it may also be valuable to explore the reverse: enabling mesh structures with actuated, length-changing edges. Such a system would allow a seasoned CAD user to first create a form in software, and then experience it in physical space—perhaps making hands-on refinements when design issues become apparent. Since each module is already equipped with absolute length sensors, closed-loop control of this shape-changing mesh may be feasible without significant hardware modifications.

A system capable of physical-to-digital and digital-to-physical transformations could serve as a platform for exploring the interplay between computational optimization and human intuition. Algorithms could suggest modifications to human-created forms to improve structural performance or material efficiency, then physically manifest these changes for designer evaluation.

Such a system could also allow for collaborative design processes within geographically distributed teams. Architects in different locations could manipulate the same structure — changes made to a physical mesh in one location could be transmitted digitally and reproduced elsewhere (e.g. on an actuated mesh, or in virtual reality).

7.6 Applications Beyond Architecture

The system also shows promise for accessibility and inclusive design. For individuals who may struggle with traditional CAD interfaces, a tangible modeling system provides an alternative pathway to spatial design. This could be particularly valuable in participatory design sessions involving diverse stakeholders with varying technical backgrounds, as illustrated in Figure 20 (bottom), where children can directly engage in designing play spaces without needing to understand scale models or interpret virtual renderings.

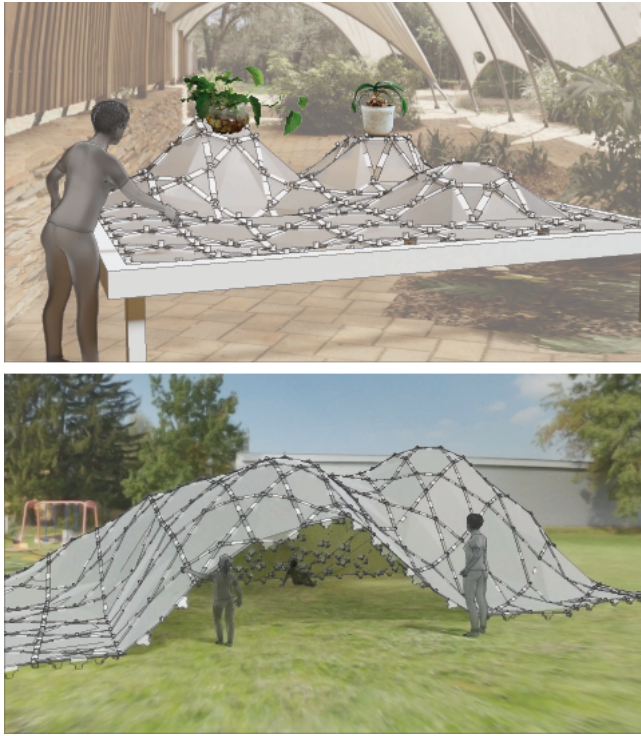


Figure 20: Envisioned uses. Top: An exhibit designer working with difficult-to-digitize items, such as plants, can arrange them directly on a physical display table. Bottom: Children can help design a play space without needing to understand scale models or virtual renderings.

In museum and exhibition design, our system offers unique advantages for working with difficult-to-digitize objects. As shown in Figure 20 (top), exhibition designers can arrange physical items like plants directly on a shape-aware surface, sculpting display fixtures around them in real time. The resulting configurations can be immediately captured in digital form, bypassing the typical challenges of 3D scanning organic objects with complex geometries.

8 Conclusion

In this paper, we introduced an interactive, large-scale sculptable mesh structure that bridges the gap between physical form-finding and digital design processes in architecture. Our system consists of adjustable-length members equipped with onboard sensors that detect changes in edge lengths, allowing for real-time digital reconstruction of the sculpted geometry. This enables architects, designers, and even community participants to physically manipulate mesh structures and immediately see the impact of their changes in a digital model.

Through a fictional participatory design scenario involving a library renovation, we demonstrated how users can intuitively shape the mesh to create various architectural elements, such as alcoves, shelves, and seating areas. The immediate physical feedback provided by this low-fidelity prototyping allows users to test and refine their ideas on location. Physical adjustments made to the

sculpted structure can be captured, reconstructed by our energy-minimization routine, and subsequently imported into CAD software for post-processing and fabrication. Through the type of enhanced user engagement that our system facilitates, we hope to further enrich the built environment.

Acknowledgments

The work in this paper was supported in part by the National Science Foundation under Award No. 2420434.

References

- [1] 2017. The value of the full-scale prototype in architectural education. *Charrette* 4, 1 (2017).
- [2] Robin Abrams. 2003. Byker revisited. *Built Environment (1978-)* (2003), 117–131.
- [3] SMLa Adriaenssens and Michael R Barnes. 2001. Tensegrity spline beam and grid shell structures. *Engineering structures* 23, 1 (2001), 29–36.
- [4] Harshit Agrawal, Udayan Umapathi, Robert Kovacs, Johannes Frohnhofen, Hsiang-Ting Chen, Stefanie Mueller, and Patrick Baudisch. 2015. Prototyper: Physically sketching room-sized objects at actual scale. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 427–436.
- [5] Moritz Bächer, Benjamin Hepp, Fabrizio Pece, Paul G Kry, Bernd Bickel, Bernhard Thomaszewski, and Otmar Hilliges. 2016. Defsense: Computational design of customized deformable input devices. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 3806–3816.
- [6] Ravin Balakrishnan, George Fitzmaurice, Gordon Kurtenbach, and Karan Singh. 1999. Exploring interactive curve and surface manipulation using a bend and twist sensitive input strip. In *Proceedings of the 1999 symposium on Interactive 3D graphics*. 111–118.
- [7] Bruce G Baumgart. 1975. A polyhedron representation for computer vision. In *Proceedings of the May 19-22, 1975, national computer conference and exposition*. 589–596.
- [8] Thomas Binder, Giorgio De Michelis, Pelle Ehn, Giulio Jacucci, and Per Linde. 2011. *Design things*. MIT press.
- [9] Eva Brandt. 2007. How tangible mock-ups support design collaboration. *Knowledge, Technology & Policy* 20, 3 (2007), 179–192.
- [10] Marion Buchenau and Jane Fulton Suri. 2000. Experience prototyping. In *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*. 424–433.
- [11] Bill Buxton. 2010. *Sketching user experiences: getting the design right and the right design*. Morgan kaufmann.
- [12] Chin-yu Chien, Rong-Hao Liang, Long-Fei Lin, Liwei Chan, and Bing-Yu Chen. 2015. Flexibend: Enabling interactivity of multi-part, deformable fabrications using single shape-sensing strip. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 659–663.
- [13] Artem Dementyev, Hsin-Liu Kao, and Joseph A Paradiso. 2015. Sensortape: Modular and programmable 3d-aware dense sensor network on a tape. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 649–658.
- [14] Mark A Evans. 2005. Rapid prototyping and industrial design practice: can haptic feedback modelling provide the missing tactile link? *Rapid Prototyping Journal* 11, 3 (2005), 153–159.
- [15] Martin Feick, Scott Bateman, Anthony Tang, André Miede, and Nicolai Marquardt. 2020. Tangi: Tangible proxies for embodied object exploration and manipulation in virtual reality. In *2020 IEEE international symposium on mixed and augmented reality (ISMAR)*. IEEE, 195–206.
- [16] Edward Glaessgen and David Stargel. 2012. The digital twin paradigm for future NASA and US Air Force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*. 1818.
- [17] Jesse T Gonzalez, Sonia Prashant, Sapna Tayal, Juhi Kedia, Alexandra Ion, and Scott E Hudson. 2023. Constraint-Driven Robotic Surfaces, At Human-Scale. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–12.
- [18] Tovi Grossman, Ravin Balakrishnan, and Karan Singh. 2003. An interface for creating and manipulating curves using a high degree-of-freedom curve input device. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 185–192.
- [19] Emrehan Gulay and Andrés Lucero. 2021. Understanding the Role of Physical and Digital Techniques in the Initial Design Processes of Architecture. In *IFIP Conference on Human-Computer Interaction*. Springer, 312–329.
- [20] Seungwoo Je, Hyunseung Lim, Kongpyung Moon, Shan-Yuan Teng, Jas Brooks, Pedro Lopes, and Andrea Bianchi. 2021. Elevate: A walkable pin-array for large

- shape-changing terrains. In *Proceedings of the 2021 CHI Conference on human Factors in Computing Systems*. 1–11.
- [21] David Jones, Chris Snider, Aydin Nassehi, Jason Yon, and Ben Hicks. 2020. Characterising the Digital Twin: A systematic literature review. *CIRP journal of manufacturing science and technology* 29 (2020), 36–52.
- [22] Siavash H Khajavi, Naser Hossein Motlagh, Alireza Jaribion, Liss C Werner, and Jan Holmström. 2019. Digital twin: vision, benefits, boundaries, and creation for buildings. *IEEE access* 7 (2019), 147406–147419.
- [23] Jay Lee, Victor Su, Sandia Ren, and Hiroshi Ishii. 2000. HandSCAPE: a vectorizing tape measure for on-site measuring applications. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 137–144.
- [24] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A low-fidelity construction kit to iteratively model, test, and adapt 3D objects. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. 471–479.
- [25] Youn-Kyung Lim, Erik Stolterman, and Josh Tenenber. 2008. The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Transactions on Computer-Human Interaction (TOCHI)* 15, 2 (2008), 1–27.
- [26] Rachael Luck. 2018. Participatory design in architectural practice: Changing practices in future making in uncertain times. *Design Studies* 59 (2018), 139–157.
- [27] Juhani Pallasmaa. 2012. *The Eyes of the Skin: Architecture and the Senses*. Wiley-Academy ; John Wiley & Sons.
- [28] Daniel Piker. 2013. Kangaroo: form finding with computational physics. *Architectural Design* 83, 2 (2013), 136–137.
- [29] Cecil Piya and Karthik Ramani. 2014. Proto-tai: Quick design prototyping using tangible assisted interfaces. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 46285. American Society of Mechanical Engineers, V01AT02A097.
- [30] Lukas Rambold, Robert Kovacs, Conrad Lempert, Muhammad Abdullah, Helena Lendowski, Lukas Fritzsche, Martin Taraz, and Patrick Baudisch. 2023. AirTied: Automatic Personal Fabrication of Truss Structures. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–10.
- [31] Elizabeth B-N Sanders and Pieter Jan Stappers. 2014. Probes, toolkits and prototypes: three approaches to making in codesigning. *CoDesign* 10, 1 (2014), 5–14.
- [32] Henry Sanoff. 1999. *Community participation methods in design and planning*. John Wiley & Sons.
- [33] Henry Sanoff. 2002. Schools Designed with Community Participation. (2002).
- [34] Henry Sanoff. 2005. Community participation in riverfront development. *CoDesign* 1, 1 (2005), 61–78.
- [35] Donald A Schon. 1992. Designing as reflective conversation with the materials of a design situation. *Research in engineering design* 3, 3 (1992), 131–147.
- [36] Fereshteh Shahmiri and Paul H Dietz. 2020. Sharc: A geometric technique for multi-bend/shape sensing. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [37] Jia Sheng, Ravin Balakrishnan, and Karan Singh. 2006. An interface for virtual 3D sculpting via physical proxy.. In *GRAPHITE*, Vol. 6. 213–220.
- [38] Karan Singh. 2006. Industrial motivation for interactive shape modeling: a case study in conceptual automotive design. In *Acm siggraph 2006 courses*. 3–9.
- [39] Miana Smith, Jack Forman, Amira Abdel-Rahman, Sophia Wang, and Neil Gershenfeld. 2025. Voxel Invention Kit: Reconfigurable Building Blocks for Prototyping Interactive Electronic Structures. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–15.
- [40] Pieter Jan Stappers, Helma van Rijn, SC Kistemaker, AE Hennink, and F Sleeswijk Visser. 2009. Designing for other people's strengths and motivations: Three cases using context, visions, and experiential prototypes. *Advanced engineering informatics* 23, 2 (2009), 174–183.
- [41] Ryo Suzuki, Hooman Hedayati, Clement Zheng, James L Bohn, Daniel Szafir, Ellen Yi-Luen Do, Mark D Gross, and Daniel Leithinger. 2020. Roomshift: Room-scale dynamic haptics for vr with furniture-moving swarm robots. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–11.
- [42] Saiganesh Swaminathan, Michael Rivera, Runchang Kang, Zheng Luo, Kadri Bagra Ozutemiz, and Scott E Hudson. 2019. Input, output and construction methods for custom fabrication of room-scale deployable pneumatic structures. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–17.
- [43] Yasaman Tahouni, Isabel PS Qamar, and Stefanie Mueller. 2020. NURBSforms: a modular shape-changing interface for prototyping curved surfaces. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction*. 403–409.
- [44] Christian Weichel, Jason Alexander, Abhijit Karnik, and Hans Gellersen. 2015. SPATA: Spatio-tangible tools for fabrication-aware design. In *Proceedings of the ninth international conference on tangible, embedded, and embodied interaction*. 189–196.
- [45] Michael Philetus Weller, Ellen Yi-Luen Do, and Mark D Gross. 2008. Posey: instrumenting a poseable hub and strut construction toy. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*. 39–46.
- [46] Michael Wessely, Theophanis Tsandilas, and Wendy E Mackay. 2018. Shape-aware material: Interactive fabrication with shapeme. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 127–139.
- [47] Karl DD Willis, Cheng Xu, Kuan-Ju Wu, Golan Levin, and Mark D Gross. 2010. Interactive fabrication: new interfaces for digital fabrication. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction*. 69–72.
- [48] Jin Yu, Prabodh Sakhardande, Ruchita Parmar, and HyunJoo Oh. 2022. Strawcures: A modular electronic construction kit for human-scale interactive structures. In *Proceedings of the Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction*. 1–10.