

ARTICLE

<https://doi.org/10.1038/s41467-019-14015-2>

OPEN

# Programming temporal morphing of self-actuated shells

Ruslan Guseinov <sup>1</sup>, Connor McMahan <sup>2</sup>, Jesús Pérez <sup>1,3</sup>, Chiara Daraio <sup>2\*</sup> & Bernd Bickel <sup>1\*</sup>

Advances in shape-morphing materials, such as hydrogels, shape-memory polymers and light-responsive polymers have enabled prescribing self-directed deformations of initially flat geometries. However, most proposed solutions evolve towards a target geometry without considering time-dependent actuation paths. To achieve more complex geometries and avoid self-collisions, it is critical to encode a spatial and temporal shape evolution within the initially flat shell. Recent realizations of time-dependent morphing are limited to the actuation of few, discrete hinges and cannot form doubly curved surfaces. Here, we demonstrate a method for encoding temporal shape evolution in architected shells that assume complex shapes and doubly curved geometries. The shells are non-periodic tessellations of pre-stressed contractile unit cells that soften in water at rates prescribed locally by mesostructure geometry. The ensuing midplane contraction is coupled to the formation of encoded curvatures. We propose an inverse design tool based on a data-driven model for unit cells' temporal responses.

<sup>1</sup>Institute of Science and Technology Austria, Am Campus 1, Klosterneuburg 3400, Austria. <sup>2</sup>California Institute of Technology, 1200 E California Blvd, Pasadena, CA 91125, USA. <sup>3</sup>Universidad Rey Juan Carlos, Calle Tulipán, s/n, Móstoles, 28933 Madrid, Spain. \*email: [daraio@caltech.edu](mailto:daraio@caltech.edu) [bernd.bickel@ist.ac.at](mailto:bernd.bickel@ist.ac.at)

Morphing flat sheets into complex, three-dimensional geometries is a challenge that has been pursued for centuries by artists, and more recently by mathematicians and scientists<sup>1–3</sup>. In engineering, the search for materials suitable for such transformations has been motivated by the ease of two-dimensional fabrication<sup>4</sup>, which relies on subtractive processes, such as punching, machining, water jetting, or laser cutting. Flat objects can be stacked in volumetrically efficient arrangements, which simplifies transportation and storage. While flat sheets are easy to fabricate and store, many structural and functional applications across scales rely on changing surface curvatures (e.g., tunable mirrors<sup>5,6</sup> and parabolic antennae<sup>7</sup>). Morphing between flat geometries and desired curved surfaces requires methods for prescribing local deformations.

To reach non-zero Gaussian curvatures from initially flat shells, bending must be coupled to in-plane stretches, according to Gauss' *Theorema Egregium*<sup>8</sup>. Several frameworks have been proposed to achieve this. Notably, the out-of-plane deformations of auxetic and kirigami sheets are defined by the architecture of voids or cut patterns<sup>9–11</sup>. Kinematic frustration has recently been embraced for changing the curvature of initially flat shells<sup>12,13</sup>. These examples are suitable for lightweight structures, but require mechanical stimuli to achieve 3D shapes through manual forming, boundary loading, or through the release of a pre-stretched sheet. Self-actuation is desirable in morphing shells because it enables untethered structural adaptation to changing environmental stimuli. To this end, responsive materials combine structural, sensing, and actuation capabilities in structures that remain flat until a non-mechanical environmental stimulus triggers the actuation process.

For example, self-actuation has been demonstrated in shells through hydrogel swelling<sup>2,14</sup> and nematic-to-isotropic phase changes in liquid-crystal elastomers<sup>15</sup>. A variety of 4D-printed systems can also be used to achieve desired shapes by coupling locally prescribed in-plane kinematics to changes in curvature<sup>16–22</sup>. Self-actuation has been extended to the folding of origami<sup>23,24</sup>, which is one of the most common and well-studied methods for inducing shape changes in initially flat objects<sup>3,4,25,26</sup>. However, none of these proposed solutions demonstrate control over deformation rates during morphing processes. Consequently, self-collisions may occur in attempts to realize more complex geometries.

The ability to locally control the shape evolution in time drastically expands the design space of self-morphing shells. More broadly, an intrinsic capacity for pre-programmed temporal responses allows designing materials that can perform complex tasks, like self-deployment and locomotion, without the need for external controllers or power supplies<sup>27</sup>. While time-dependent folding has been demonstrated in structures that are wired to power sources and electronic control devices<sup>28,29</sup>, a small number of architected shells made of materials with intrinsic actuation capabilities have incorporated temporal programming through the sequential folding of discrete hinges<sup>30–32</sup>. However, none of these examples allow for changes in Gaussian curvature. They realize sequential folding of few discrete hinges<sup>30</sup>, must be fabricated in their target shape prior to manual programming<sup>31</sup>, or rely on ad hoc empirical designs that do not account for characterizations of their materials' time-dependent constitutive responses<sup>32</sup>.

Here, we show that spatio-temporal information can be embedded in the geometry of architected shells that morph from flat to smooth three-dimensional shapes. This programmed temporal evolution enables reaching target geometries that would be impeded by collisions if shells actuated with uniform or unplanned deformation rates. Furthermore, our shells use polymers that actuate when the temperature in their environment is

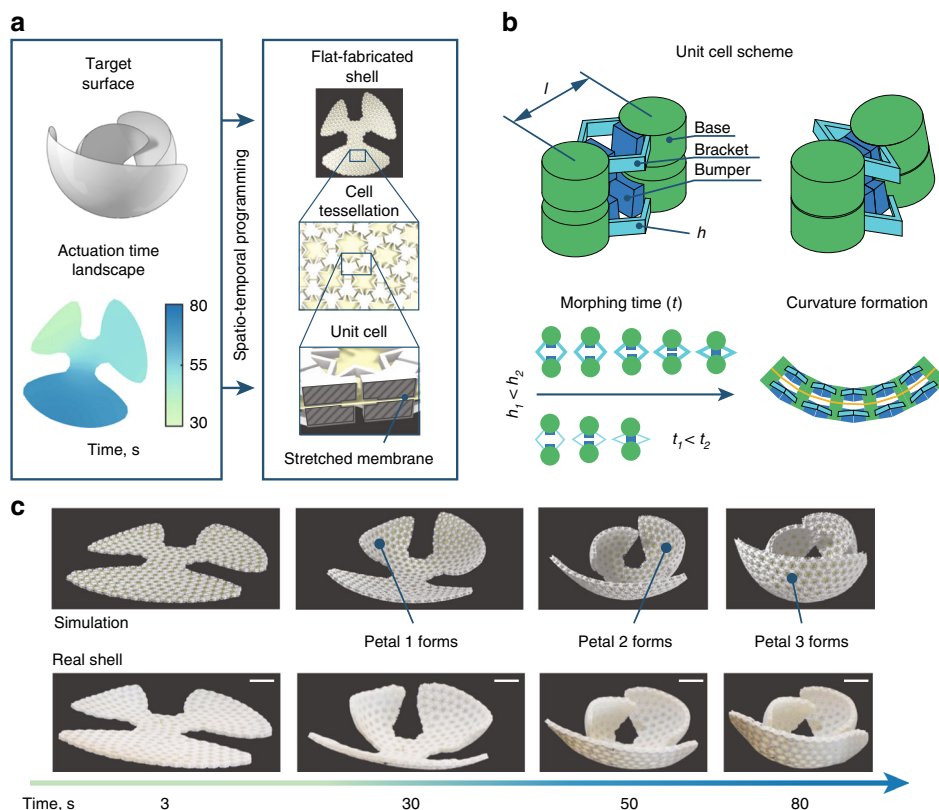
set to a critical value. At room temperature, they remain flat, storing the energy necessary to drive the deformation.

## Results

**Spatio-temporal programming of self-actuated shell mechanisms.** We propose an inverse design algorithm for shell architectures and the temporal evolution of their shapes (Fig. 1a). The algorithm collects user inputs at two stages: the first input is the desired 3D target surface and the second is the specification of local deformation rates. We term this temporal map input an *actuation time landscape*. The algorithm outputs the mesostructure for initially flat shells that we fabricate and test. These shells have three layers, with a  $\sim 4.6$  mm total thickness. The two outer layers are 3D-printed tessellations of non-uniform unit cells, made of Vero PureWhite (Stratasys). The middle layer is a 0.5 mm thick pre-stretched elastic membrane, which stores the energy required to drive the morphing process. Actuation from the flat to the curved profile is triggered by immersing the shells into 56 °C water, which causes the outer layers to soften over the course of approximately 30–80 s.

The unit cells have a grid spacing of  $\sim 10$  mm, and are composed of cylindrical bases connected at the external shell surfaces by pairs of V-shaped brackets (Fig. 1b). The bases serve as attachment points to the elastic membrane and as mounting points for the brackets. The brackets serve as nonlinear springs: they hold the structure flat prior to being placed in water, and then they guide the temporal morphing process, softening at rates determined by their geometric parameters when heated (thicker brackets soften at slower rates). There are also bumpers attached to the bases in the space between the brackets, which collide when the local contraction reaches the target magnitude and control the contact angle between adjacent unit cells. The membrane provides energy for actuation, compressing the brackets as they soften. All of these components play an important role for reaching targeted geometries through spatio-temporal programming. For example, the petalled structure shown in Fig. 1c has been programmed so that the petals reach their target shape sequentially, actuating from smallest to largest (Supplementary Movie 1). If all petals deformed at the same rate, they would collide and would not reach the target geometry. More details on shell design are available in the Supplementary Note 1.

We incorporated a discretized mechanical model of our shells in an inverse design algorithm for obtaining desired temporal morphing. Given a target geometry and a smooth time landscape, the algorithm automatically generates the flat shell mesostructure that will produce the corresponding morphing process. It first composes a continuous target shell out of compressed unit cells. To do this, the target surface is isotropically triangulated, producing a stencil that serves as a placeholder for base locations. With the arrangement of bases in the target shape, the bumper geometries are defined to ensure that they are in contact in this target shape (Fig. 1b top-right). Then, a minimal distortion conformal map<sup>33</sup> flattens the stencil. The bases with bumpers are then relocated to the flat stencil and are interconnected by brackets. Note that this stencil has to be free of overlaps to enable fabrication. Therefore, base placement, bumper arrangements, and bracket lengths are configured automatically given a target surface input. However, the selection of bracket thicknesses is governed by the designer's specification of the actuation time landscape. Thicker brackets soften at a slower rate than narrower ones, enabling distinct target deformation times to be realized in each region of the shell for collision avoidance, visual impression or other desired functionalities. Given that there can be a broad range of morphing sequences that yield certain target geometries, the morphing process can be designed according to the designer's



**Fig. 1 Encoding spatial and temporal shape evolution in a flat shell mesostructure.** **a** A user-specified target surface and actuation time landscape (a field of deformation completion times) are inputs to an inverse design procedure that defines the mesostructure of flat-fabricated shells that morph into the target geometries. The shells are composed of inhomogeneous tessellations of unit cells with an interior pre-stretched membrane. **b** Each unit cell has an initial central length  $l$ . Brackets control actuation time through their softening rate, which is controlled by their thickness,  $h$ , and a set of bumpers prescribe final local curvatures upon collision. **c** Morphing of a petalled structure with an actuation time landscape ensuring that larger petals cover their smaller neighbors avoiding collisions on the way. Simulation and experiments are compared at 3, 30, 50, and 80 seconds in water. The structure replicates the encoded actuation time landscape shown in (a). Scale bars, 3 cm.

goals by iterating through actuation time landscapes and observing their effect.

The time evolution of the shells is simulated quasistatically by coupling a finite element simulation of the rubber membrane with a data-driven spring model for the brackets and a rigid body model for bases. Bumper collisions are described as sharp increases in bracket stiffness in the model. A summary of the energy model is given below. Its constitutive parts are the energy associated with bracket compression ( $W_c$ ) obtained from fitting and interpolating experimental data, an energy penalty to shearing ( $W_s$ ) that replicates the effect of the shear-resisting bracket geometry, and the elastic membrane energy ( $W_m$ ):

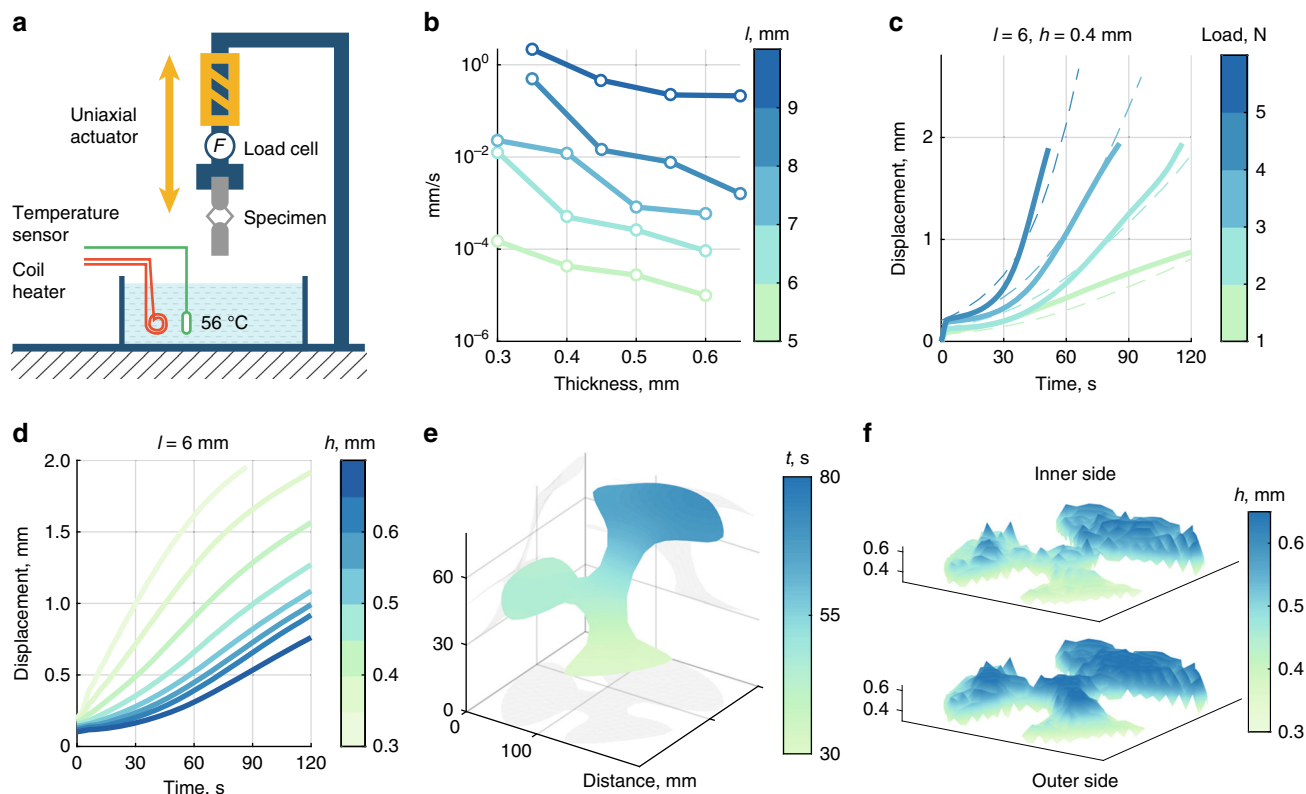
$$W(\mathbf{x}) = \sum_{c_{ij}} W_c + \sum_{s_{ij}} W_s + \sum_{T_i} W_m(\mathbf{G}_i). \quad (1)$$

Here,  $c_{ij}$  refers to the contractile springs that join the  $i$ th and the  $j$ th bases. Each unit cell is modeled with four of these springs to capture bending effects. Bumper collisions are modeled as a sharp stiffening of these elements. Shear-resisting elements  $s_{ij}$  have analogous indexing.  $T_i$  refers to the  $i$ th element of the membrane discretization, and  $\mathbf{G}_i$  is the deformation gradient of the membrane evaluated at this element. A complete description of the simulation procedure is available in the Supplementary Note 2.

To construct the constitutive model for bracket softening, we conducted experiments (Fig. 2a) on brackets of varying length  $l$  (in a range 5–9 mm) and thickness  $h$  (0.3–0.65 mm), applying constant forces (1–5 N) and tracking their compression over time

spent in water using a Zwick tensile tester (Figs. 2b, c). Fits to the experimental data were then sampled from the space of bracket parameters and immersion times to build a time-dependent force–displacement model used in the simulation (Fig. 2d). Material measurement and modeling are discussed more extensively in the Supplementary Note 3. From this sampling, we select bracket thicknesses that yield target deformation timings under the loads generated by the membrane (see Supplementary Note 4).

**Examples of temporally programmed structures.** We highlight the effect that different actuation time landscapes have on the final shapes of initially flat shells by comparing the example discussed in Fig. 1c to the shell shown in Fig. 3a. Both shells have similar flat geometries but different actuation time landscapes are encoded in their mesostructures. In the first example, smaller petals are covered by their larger neighbors. Meanwhile, each petal shown in the second example has an edge that covers a neighbor and one edge that is covered. For the latter case, all the petals have been programmed with the same actuation time landscape so they deform simultaneously. We slightly increased target actuation times for some petal tips to increase the distance between neighboring petals on their morphing paths (Supplementary Movie 2). This way, the interior edge of each petal completes its deformation before being covered by its neighbor. Shape-morphing precision was measured using a 3D scan of the final geometry. The blue markers were matched with their simulated locations. The resulting mean error for the pairwise



**Fig. 2** From measurement of brackets' mechanical properties to inverse design of temporal morphing. **a** Load-controlled tensile tests were used to determine the deformation rates of unit cells in 56° C water. **b** Average deformation rates for specimens subject to constant loads of 4 N for  $l < 7$  and 5 N for  $l \geq 7$  N. These values are close to the inner membrane tractions on each unit cell in real shells. **c** Deformation rate measurements (solid lines) are fit (dashed lines) to produce a model of bracket softening. Here we show the fit for  $l = 6$  mm,  $h = 0.4$  mm. **d** The model is interpolated and queried to infer the mesostructure that yields target curvatures and deformation completion times in each section of the shell. Here, we show deformations of unit cells with central length  $l = 6$  mm and a range of bracket thicknesses from 0.3 mm to 0.65 mm. **e** Smooth actuation time landscape that induces the sequential deformation process demonstrated in Fig. 1. **f** Bracket thickness fields for both sides of the petalled shape. Though the prescribed time landscape is smooth, the field of bracket thickness is highly irregular because bracket thicknesses also depend on initial unit cell lengths and their target deformations.

distances in the experimental realization of the structure shown in Fig. 3a was 3.6% relative to the diameter of the target markers' point cloud.

Developable target surfaces can also be achieved using this structural framework. The actuation time landscape for the double-loop spiral shown in Fig. 3b is a constant gradient from one end to the other. This allows the interior to curl without colliding with the outer loop (Supplementary Movie 3). The 3D scan reveals a 2.4% mean error. Geometries with negative Gaussian curvature can also be realized, such as the saddle shown in Fig. 3c (Supplementary Movie 4). The relative mean error for the saddle's base positions was 0.8%.

We showcase the complexity of achievable shapes through the self-interweaving structure shown in Fig. 3d. This shape requires embedding precise morphing trajectories and time landscapes in the flat-printed structure in order to thread each arm through the loop created by a neighboring one without colliding (Supplementary Movie 5). The experimental replication of this challenging target geometry yields a 9.7% 3D scan mean error, and highlights the versatility of our design and simulation framework.

## Discussion

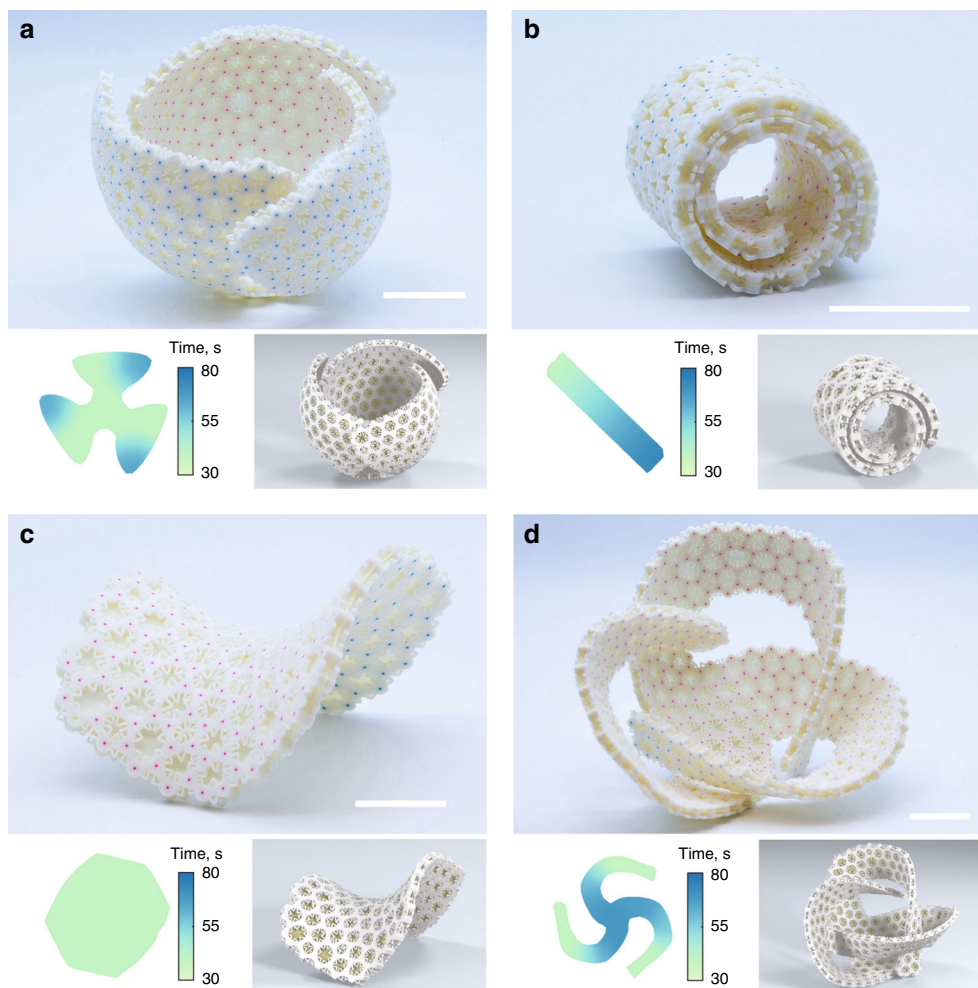
The realization of complex 3D geometries from flat-fabricated structures, which are easier to manufacture and transport, is a compelling motivation for developing shape-morphing frameworks. However, to be used in a broad range of engineering applications, the morphed structures must remain structurally

stable. In our examples, the shells' outer layers soften during deformation in hot water, but become stiff when returned to room temperature (see the details on the shells' fabrication procedure in the Supplementary Note 5). We show their load-bearing ability realizing a mobile phone stand (Fig. 4a). Note that the mass of the phone (113 g) exceeds the mass of the shell (55 g). Data from simple mechanical tests on flat structures are available in Supplementary Note 6. Structural stability could be further increased by using snap-locking mechanisms instead of bumpers or coating the structure after actuation has completed.

Temporally programmed self-morphing opens the door to a broad range of industrial design applications. Sequentially interlocking joints, for example, can be employed in the self-assembly of initially flat furniture (Fig. 4b), while doubly curved surfaces can be used in drone shells (Fig. 4c). The Schwarz P triply periodic minimal surface structure (Fig. 4d) can be obtained from layered sheets that are programmed to have time-evolving porosity. Such structure could be used as an energy absorbing material or a microfluidic device with programmable channel cross-sections (Fig. 4e).

Our method for programming temporal morphing in shells is based on a combination of a pre-stressed midplane and outer layers with effective stiffness differentials that are configured to evolve over time according to user specifications. The encodement is performed using an inverse design algorithm that takes a target surface and an actuation time landscape as inputs and outputs a mesostructure with embedded self-morphing





**Fig. 3 Spatio-temporally programmed shells.** Each panel shows a real shell (top), its actuation time landscape (bottom-left), and its corresponding simulated shell (bottom-right). **a** Doubly curved shell where petals morph synchronously to cover each other in a cyclic manner. One corner of each petal is programmed to morph slower to increase the distance between petals during morphing. **b** A double spiral that approximates a developable surface. A gradient time landscape enables the inner spiral to curl first. **c** A saddle shape with negative curvature. **d** A shell with a complex self-interweaving shape prone to multiple collisions in the course of its morphing process. Scale bars, 3 cm.

information. The significance of this method is that it enables collision-avoidance during deformations from flat shapes to curved geometries. We built a design system based on a data-driven mechanical model of mesostructures to predict shape evolution in time, enabling temporal morphing design. Applications of self-actuating shells to biomedical and construction industries are close to becoming reality with the fast advances in this field of study. Further generalizations of our approach to other materials such as liquid-crystal elastomers, bio-compatible polymers, and conventional engineering materials whose properties evolve in time due to other stimuli such as temperature, humidity, light, pH, etc., could enable rapid manufacturing of load-bearing structures that can only assume desired geometries through temporally planned deformations upon deployment as well as robotic materials temporally programmed for a broader range of functionalities.

## Methods

**Experiments.** All specimens used for bracket characterization and the outer layers of the shape-morphing structures were 3D-printed using a Stratasys J750 using Vero Pure White material. Colored markers were included in the bases to facilitate visualization. Water was kept at 56 °C using a temperature controller, and two Canon 700D cameras were used for imaging.

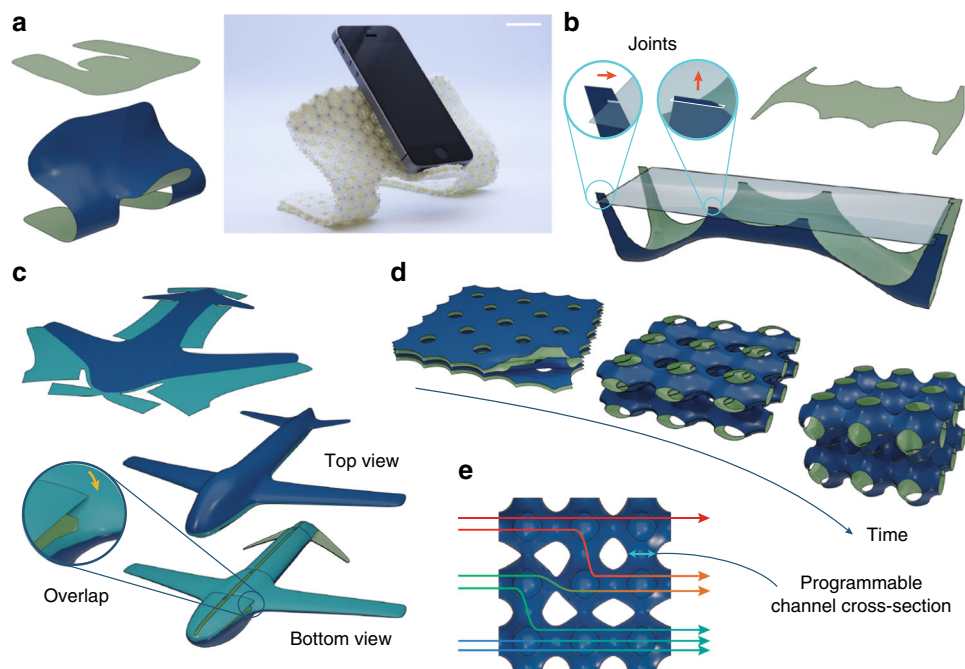
The mechanical properties of the brackets were measured using a Zwick tensile tester with a custom-built water tank attachment. Experiments measuring strain

restitution after unloading were conducted to estimate the plastic fraction of the deformation. These experiments are discussed more extensively in the Supplementary Information.

All shells were fabricated by first uniformly stretching a latex sheet of thickness 0.5 mm to 900% its initial area. After cleaning the membrane surfaces with 2-Propanol, the printed lattices are glued to the membrane. In each structure, several bases have holes to align the opposite shell layers using push-pins. Latex surplus surrounding the assembled flat shell is removed, then the shells are submerged into a 350 × 350 × 350 mm water tank to induce shape-morphing. See the Supplementary Information for more details.

**Data fitting and simulation.** Experimental data was used to generate a force-displacement model of brackets by combining simpler fitting components. First, displacement-force curves were fitted so that the initial state corresponded to the behavior at room temperature. Second, displacement rates dependent on time in water were fitted. Based on the combination of these two fittings, the parameter domain was resampled to yield time-dependent force-displacement relationships. Data from the plasticity experiments were used to build the final elastoplastic model used in the simulation software for inverse design.

In addition to implementing this data-driven bracket compression model in simulations, shear-resisting elements are included to capture brackets' geometry-based shear resistance. Bases are simulated as rigid bodies and a FEM approach is used for the membrane simulation. Bumper collisions are modeled as abrupt bracket stiffness increases. All simulation elements are coupled via shared vertices. We implemented the simulations using a C++ code developed in-house. A simple user interface was designed to import target surfaces and specify time landscapes. User inputs are automatically processed to configure the simulated structure and display resulting morphing processes. Once the



**Fig. 4** Future concepts for spatio-temporally programmed materials. Potential applications for shells with programmable self-morphing, showing initial flat and final states. Opposite sides of each sheet are colored blue and green. **a** The load-bearing capability of our morphing shells is shown by a mobile phone stand. Scale bar, 3 cm. **b** Self-morphing can be applied to industrial design, for example to build aesthetically curved furniture with sequentially interlocking joints. **c** The outer shell of a self-morphing drone can be fabricated from a single sheet. The smooth aerodynamic shape is achieved with temporal programming of sequentially overlapping parts. **d** Stacked multilayer sheets can be used to create stiff, yet lightweight periodic mesostructures with temporally programmed porosity. Four sheet layers (left) morph into a Schwarz P surface (right) which efficiently distributes external stresses through the structure. **e** Similar structures with programmable channel cross-section between the periodic blocks can be used to adjust flow rates through the device.

desired morphing is achieved, the system automatically generates structures for 3D printing.

The data-fitting and simulation procedures are explained in more detail in the Supplementary Materials.

**3D scanning.** An HP David SLS-3 structured light scanner with two cameras was used to generate textured 3D meshes in OBJ format. Then, the textures were filtered to obtain binary images with markers. Markers were lifted to their actual scanned positions using the 3D mesh and resolved to single points. The obtained point clouds were registered to the clouds generated by the simulation software and points were matched using Munkres' algorithm.

### Data availability

All raw data is publicly available through IST Austria Research Explorer <https://doi.org/10.15479/AT:ISTA:7154>.

### Code availability

Source code and processed data are publicly available on <https://github.com/russelmann/temporal-morphing-ncomms>.

Received: 31 October 2019; Accepted: 11 December 2019;

Published online: 13 January 2020

### References

- Baek, C., Sageman-Furnas, A. O., Jawed, M. K. & Reis, P. M. Form finding in elastic gridshells. *Proc. Natl Acad. Sci. USA* **115**, 75–80 (2018).
- Klein, Y., Efrati, E. & Sharon, E. Shaping of elastic sheets by prescription of non-euclidean metrics. *Science* **315**, 1116–1120 (2007).
- Lang, R. J. A computational algorithm for origami design. In *Proc. 12th Annual Symposium on Computational Geometry, SCG '96* (ed Whitesides, S.), 98–105 (ACM, 1996).
- Callens, S. J. P. & Zadpoor, A. A. From flat sheets to curved geometries: Origami and kirigami approaches. *Mater. Today* **21**, 241–264 (2018).
- Clampin, M. Status of the james webb space telescope (JWST). In *Space Telescopes and Instrumentation*, Vol. 7010. <https://doi.org/10.1117/12.790388> (2008).
- Liu, W. & Talghader, J. J. Current-controlled curvature of coated micromirrors. *Opt. Lett.* **28**, 932–934 (2003).
- Peral, E. et al. Raincube: a proposed constellation of precipitation profiling radars in cubesat. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (eds Serpico, S. B. et al.) 1261–1264 (IEEE, 2015).
- Gauss, C. F. *Disquisitiones Generales Circa Superficies Curvas* Typis Ditericianis (1828).
- Konaković, M. et al. Beyond developable: computational design and fabrication with auxetic materials. *ACM Trans. Graph.* **35**, 89 (2016).
- Wang, F., Guo, X., Xu, J., Zhang, Y. & Chen, C. Q. Patterning curved three-dimensional structures with programmable kirigami designs. *J. Appl. Mech.* **84**, 061007 (2017).
- Zhang, Y. et al. A mechanically driven form of kirigami as a route to 3d mesostructures in micro/nanomembranes. *Proc. Natl Acad. Sci. USA* **112**, 11757–11764 (2015).
- Celli, P. et al. Shape-morphing architected sheets with non-periodic cut patterns. *Soft Matter* **14**, 9744–9749 (2018).
- Guseinov, R., Miguel, E. & Bickel, B. Curveups: shaping objects from flat plates with tension-actuated curvature. *ACM Trans. Graph.* **36**(4), 64 (2017).
- Kim, J., Hanna, J. A., Byun, M., Santangelo, C. D. & Hayward, R. C. Designing responsive buckled surfaces by halftone gel lithography. *Science* **335**, 1201–1205 (2012).
- Ware, T. H., McConney, M. E., Wie, J. J., Tondiglia, V. P. & White, T. J. Voxellated liquid crystal elastomers. *Science* **347**, 982–984 (2015).
- Ambulo, C. P. et al. Four-dimensional printing of liquid crystal elastomers. *ACS Appl. Mater. Interfaces* **9**, 37332–37339 (2017).
- Bauhofer, A. A. et al. Harnessing photochemical shrinkage in direct laser writing for shape morphing of polymer sheets. *Adv. Mater.* **29**, 1703024 (2017).

18. Boley, J. W. et al. Shape-shifting structured lattices via multimaterial 4d printing. *Proc. Natl Acad. Sci.* page 201908806. <https://doi.org/10.1073/pnas.1908806116> (2019).
19. Gladman, A. S., Matsumoto, E. A., Nuzzo, R. G., Mahadevan, L. & Lewis, J. A. Biomimetic 4D printing. *Nat. Mater.* **15**, 413–418 (2016).
20. Kotikian, A., Truby, R. L., Boley, J. W., White, T. J. & Lewis, J. A. 3d printing of liquid crystal elastomeric actuators with spatially programmed nematic order. *Adv. Mater.* **30**, 1706164 (2018).
21. Raviv, D. et al. Active printed materials for complex self-evolving deformations. *Sci. Rep.* **4**, 7422 (2014).
22. Saed, M. O. et al. Molecularly-engineered, 4d-printed liquid crystal elastomer actuators. *Adv. Funct. Mater.* **29**, 1806412 (2019).
23. Na, J. H. et al. Programming reversibly self-folding origami with micropatterned photo-crosslinkable polymer trilayers. *Adv. Mater.* **27**, 79–85 (2014).
24. Plucinsky, P., Kowalski, B. A., White, T. J. & Bhattacharya, K. Patterning nonisometric origami in nematic elastomer sheets. *Soft Matter* **14**, 3127–3134 (2018).
25. Demaine, E. D. and Tachi, T. Origamizer: a practical algorithm for folding any polyhedron. In *33rd Int. Symp. on Computational Geometry (SoCG 2017)* Vol. 77 (eds Aronov, B. et al.) 34:1–34:16 (Schloss Dagstuhl, Germany, 2017).
26. Dudte, L. H., Vouga, E., Tachi, T. & Mahadevan, L. Programming curvature using origami tessellations. *Nat. Mater.* **15**, 583–588 (2016).
27. Kotikian, A. et al. Untethered soft robotic matter with passive control of shape morphing and propulsion. *Sci. Robotics* **4**, eaax7044 (2019).
28. Felton, S., Tolley, M., Demaine, E., Rus, D. & Wood, R. A method for building self-folding machines. *Science* **345**, 644–646 (2014).
29. Hawkes, E. et al. Programmable matter by folding. *Proc. Natl Acad. Sci. U.S.A.* **107**, 12441–12445 (2010).
30. Liu, Y., Shaw, B., Dickey, M. D. & Genzer, J. Sequential self-folding of polymer sheets. *Sci. Adv.* **3**, e1602417 (2017).
31. Mao, Y. et al. Sequential self-folding structures by 3d printed digital shape memory polymers. *Sci. Rep.* **5**, 13616 (2015).
32. Tao, Y. et al. Demonstrating thermorph: democratizing 4d printing of self-folding materials and interfaces. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems, CHI EA '18* (eds Mandryk, R. L., Hancock, M., Perry, M. & Cox, A. L.) D405:1–D405:4. (ACM, 2018).
33. Springborn, B., Schröder, P. & Pinkall, U. Conformal equivalence of triangle meshes. *ACM Trans. Graph.* **27**, 77:1–77:11 (2008).

## Acknowledgements

R.G. and B.B. were supported by the European Research Council (ERC) under grant agreement No 715767 - MATERIALIZABLE: Intelligent fabrication-oriented Computational Design and Modeling. J.P. was supported by the European Union's Horizon 2020

Marie Skłodowska-Curie COFUND Action ISTPlus under Grant Agreement No. 754411. C.M. and C.D. were supported by the US Army Research Office Grant W911NF-17-1-0147. C.M. was supported by a NASA Space Technology Research Fellowship.

## Author contributions

R.G., C.D. and B.B. designed the research. R.G. and C.M. conducted experiments and performed data modeling and fitting for bracket specimens. J.P. implemented physical simulator. R.G., C.M. and J.P. contributed to mechanical modeling. R.G. designed target shape geometries. R.G. and J.P. developed the inverse design tool and performed simulations. R.G. fabricated, tested, and scanned shells. C.D. and B.B. supervised the research. All authors analyzed data, wrote the manuscript, and developed the figures.

## Competing interests

The authors declare no competing interests.

## Additional information

**Supplementary information** is available for this paper at <https://doi.org/10.1038/s41467-019-14015-2>.

**Correspondence** and requests for materials should be addressed to C.D. or B.B.

**Peer review information** *Nature Communications* thanks Jonathan Hopkins for their contribution to the peer review of this work. Peer reviewer reports are available.

**Reprints and permission information** is available at <http://www.nature.com/reprints>

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2020