

Supplementary Materials for
**Shape and topology morphing of closed surfaces integrating origami
and kirigami**

Xiangxin Dang *et al.*

Corresponding author: Damiano Pasini, damiano.pasini@mcgill.ca

Sci. Adv. **11**, eads5659 (2025)
DOI: 10.1126/sciadv.ads5659

The PDF file includes:

Supplementary Text
Figs. S1 to S17
Legends for movies S1 to S5

Other Supplementary Material for this manuscript includes the following:

Movies S1 to S5

Supplementary Text

S1 Mesh

We propose two classes of ori-kiri assemblages with RS meshes and UV meshes, respectively. The abbreviation “RS” denotes “rotating square”, i.e., the classical kirigami pattern exhibiting auxetic properties by the rotation of square cut units. The abbreviation “UV” represents “UV mapping”, a texture mapping technique that can unwrap images on a sphere to a planar surface. As shown in Figure S1A, the cube with an RS mesh has six faces of 4×4 squares. Each square is divided into two triangles by a diagonal crease. The square can fold along its crease and rotate relative to its neighbor square at the common vertex. The RS mesh can also be made on curved surfaces. As shown in Figure S1B, the sphere with an RS mesh has six faces as well, while the squares have been generalized to skew quadrilaterals to approximate the curved surface. In addition, we can also approximate a sphere with a UV mesh. As shown in Figure S1C, the standard UV mesh is composed of trapezoids except for triangles at the poles. Similar to the RS mesh, each trapezoid is divided by a diagonal crease.

Both the RS meshes and the UV meshes have deployed forms with open slits. As shown in Figure S1D, the cube with an RS mesh has six deployed faces. The green lines indicate discontinuous intersections across which the panels are not connected, while the red lines represent continuous intersections where the panels are connected at their common vertices. Figures S2A and S2B demonstrate all the continuous and discontinuous intersections on an RS mesh. Figure S1E shows the continuous and discontinuous intersections of a deployed RS mesh on a sphere. Figure S1F shows a torus with a deployed UV mesh where the trapezoids become skew quadrilaterals. The green line at the inner circumference of the torus is a discontinuous intersection where the panels on opposite sides are not connected with each other.

S2 Parametrization

Cube with RS mesh. Following the notations illustrated in Figure S2C, we can index the six faces of the cube by 1 to 6, and the eight vertices by 1 to 8. Figure S4A shows a cube in a Cartesian coordinate system. We denote the eight vertices by V_1 to V_8 . Given the edge length of the cube $2L$,

the positions of V_1 to V_8 can be written as

$$\begin{aligned} V_1 &= (-L, -L, -L), & V_2 &= (L, -L, -L), & V_3 &= (L, L, -L), & V_4 &= (-L, L, -L), \\ V_5 &= (-L, -L, L), & V_6 &= (L, -L, L), & V_7 &= (L, L, L), & V_8 &= (-L, L, L). \end{aligned} \quad (S1)$$

We additionally introduce local indices of the four vertices on each face, as shown in Figure S2D. We denote the locally-indexed vertices on the k -th face by \tilde{V}_{k1} to \tilde{V}_{k4} for $k = 1, 2, \dots, 6$. The relationship between the global and local indices is given by

$$\begin{aligned} \text{face 1: } & \tilde{V}_{11} = V_1, \quad \tilde{V}_{12} = V_2, \quad \tilde{V}_{13} = V_6, \quad \tilde{V}_{14} = V_5; \\ \text{face 2: } & \tilde{V}_{21} = V_2, \quad \tilde{V}_{22} = V_3, \quad \tilde{V}_{23} = V_7, \quad \tilde{V}_{24} = V_6; \\ \text{face 3: } & \tilde{V}_{31} = V_3, \quad \tilde{V}_{32} = V_4, \quad \tilde{V}_{33} = V_8, \quad \tilde{V}_{34} = V_7; \\ \text{face 4: } & \tilde{V}_{41} = V_4, \quad \tilde{V}_{42} = V_1, \quad \tilde{V}_{43} = V_5, \quad \tilde{V}_{44} = V_8; \\ \text{face 5: } & \tilde{V}_{51} = V_4, \quad \tilde{V}_{52} = V_3, \quad \tilde{V}_{53} = V_2, \quad \tilde{V}_{54} = V_1; \\ \text{face 6: } & \tilde{V}_{61} = V_5, \quad \tilde{V}_{62} = V_6, \quad \tilde{V}_{63} = V_7, \quad \tilde{V}_{64} = V_8; \end{aligned} \quad (S2)$$

For the k -th face, we refer to the segment $\tilde{V}_{k1}\tilde{V}_{k4}$ as the left boundary, and the segment $\tilde{V}_{k1}\tilde{V}_{k2}$ as the bottom boundary. On each face, the position of the point P can be determined by its distance away from the left and bottom boundaries, denoted by u and v , respectively. We scale u and v by

$$p = \frac{u}{2L}, \quad q = \frac{v}{2L}. \quad (S3)$$

As a result, the parameters p and q are within the interval $[0, 1]$.

We denote the origin point of the Cartesian coordinate system by O . Naturally, the coordinates of the origin is $O = (0, 0, 0)$. Supposing the coordinates of the point P is (x, y, z) , we have $\overrightarrow{OP} = (x, y, z)$. For the point P on the k -th face, the parameters p and q can be calculated by

$$p = \frac{\overrightarrow{\tilde{V}_{k1}\tilde{V}_{k2}} \cdot (\overrightarrow{OP} - \overrightarrow{O\tilde{V}_{k1}})}{4L^2}, \quad q = \frac{\overrightarrow{\tilde{V}_{k1}\tilde{V}_{k4}} \cdot (\overrightarrow{OP} - \overrightarrow{O\tilde{V}_{k1}})}{4L^2}. \quad (S4)$$

Inversely, we can use the parameters p and q to locate the point P by

$$\overrightarrow{OP} = \overrightarrow{O\tilde{V}_{k1}} + p\overrightarrow{\tilde{V}_{k1}\tilde{V}_{k2}} + q\overrightarrow{\tilde{V}_{k1}\tilde{V}_{k4}}, \quad (S5)$$

where $p, q \in [0, 1]$. Equation (S4) maps (x, y, z) to (p, q) , while Equation (S5) maps (p, q) to (x, y, z) . We define the following functions

$$(p, q) = \mathbf{f}_{\text{cube-RS}}(x, y, z; k, L), \quad (S6)$$

$$(x, y, z) = \mathbf{g}_{\text{cube-RS}}(p, q; k, L). \quad (\text{S7})$$

The function $\mathbf{f}_{\text{cube-RS}}$ represents the map given by Equation (S4). The function $\mathbf{g}_{\text{cube-RS}}$ represents the map given by Equation (S5).

Sphere with RS mesh. Figure S4B shows a sphere with an RS mesh in a Cartesian coordinate system. The sphere is divided into six spherical square faces with eight vertices. We follow the same index rules as those for the cube (Figures S2C and S2D). Given the radius R , the positions of V_1 to V_8 are written as

$$\begin{aligned} V_1 &= (-R, -R, -R)/\sqrt{3}, & V_2 &= (R, -R, -R)/\sqrt{3}, \\ V_3 &= (R, R, -R)/\sqrt{3}, & V_4 &= (-R, R, -R)/\sqrt{3}, \\ V_5 &= (-R, -R, R)/\sqrt{3}, & V_6 &= (R, -R, R)/\sqrt{3}, \\ V_7 &= (R, R, R)/\sqrt{3}, & V_8 &= (-R, R, R)/\sqrt{3}. \end{aligned} \quad (\text{S8})$$

The local-global index relationship follows Equation (S2) again. For the k -th face, the left boundary is the great-circle arc (or equivalently, geodesic line) connecting vertices \tilde{V}_{k1} and \tilde{V}_{k4} , and the bottom boundary is the great-circle arc connecting \tilde{V}_{k1} and \tilde{V}_{k2} . We can locate a point on each face by its relative position away from the left and bottom boundaries. For example, in Figure S4B left, the point P falls on face 1. The left boundary can be extended to the two antipodal points on the z axis and form a half great circle. We can find another half great circle ending at this pair of antipodal points but passing through the point P . These two half great circles bound a spherical lune of a dihedral angle α . We can follow the same procedure to obtain the other spherical lune intersecting with the bottom boundary. The resulting dihedral angle is denoted by β . The dihedral angles α and β determine the position of the point P on the face. We map α and β onto the interval $[0, 1]$ by

$$p = \frac{2\alpha}{\pi}, \quad q = \frac{2\beta}{\pi}. \quad (\text{S9})$$

In general, for a point on the k -th face, we can obtain the parameters p and q based on the left and bottom boundaries of the face. We denote the origin point of the coordinate system by $O = (0, 0, 0)$, and the coordinate of the point P by (x, y, z) . Then, we have $\overrightarrow{OP} = (x, y, z)$. We aim to express the parametrization for the six faces with a unified formulation. To this end, we take the 2nd face (i.e., the one intersecting with the positive semi x axis) as a reference face and rotate the points on the other faces to the 2nd face, around the spherical center (i.e., the origin of the coordinate

system O). Specifically, we rotate the boundaries of a face along with the target point and preserve their relative locations, such that the corresponding boundaries (say, the left, bottom, right, and top boundaries) of this face coincide with the corresponding boundaries on the 2nd face. We suppose the point P on the k -th face moves to the point \tilde{P} on the 2nd face after the rotation. We denote the coordinates of \tilde{P} by $(\tilde{x}, \tilde{y}, \tilde{z})$. The rotations from the points P to \tilde{P} can be expressed as

$$(\tilde{x}, \tilde{y}, \tilde{z}) = \begin{cases} (x, y, z) \cdot R_z\left(\frac{\pi}{2}\right) & \text{if } k = 1, \\ (x, y, z) & \text{if } k = 2, \\ (x, y, z) \cdot R_z\left(-\frac{\pi}{2}\right) & \text{if } k = 3, \\ (x, y, z) \cdot R_z(\pi) & \text{if } k = 4, \\ (x, y, z) \cdot R_x\left(-\frac{\pi}{2}\right) R_z\left(\frac{\pi}{2}\right) & \text{if } k = 5, \\ (x, y, z) \cdot R_x\left(\frac{\pi}{2}\right) R_z\left(\frac{\pi}{2}\right) & \text{if } k = 6, \end{cases} \quad (\text{S10})$$

in which $R_x(t)$ and $R_z(t)$ are the rotations around the x axis and the z axis, respectively. The expressions of $R_x(t)$ and $R_z(t)$ can be written as

$$R_x(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos t & \sin t \\ 0 & -\sin t & \cos t \end{bmatrix}, \quad R_z(t) = \begin{bmatrix} \cos t & \sin t & 0 \\ -\sin t & \cos t & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (\text{S11})$$

After the rotations, the dihedral angles α and β remain unchanged:

$$\tilde{\alpha} = \alpha, \quad \tilde{\beta} = \beta. \quad (\text{S12})$$

On the reference face, to parameterize the point \tilde{P} , first, we compress the Cartesian coordinates $(\tilde{x}, \tilde{y}, \tilde{z})$ to the azimuth angle $\tilde{\theta}$ and the elevation angle $\tilde{\varphi}$ by

$$\tilde{\theta} = \text{atan2}(\tilde{y}, \tilde{x}), \quad \tilde{\varphi} = \text{atan2}(\tilde{z}, \sqrt{\tilde{x}^2 + \tilde{y}^2}), \quad (\text{S13})$$

where the function atan2 is the four-quadrant inverse tangent defined as

$$\text{atan2}(y, x) = \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{if } x > 0, \\ \arctan\left(\frac{y}{x}\right) + \pi & \text{if } x < 0 \text{ and } y \geq 0, \\ \arctan\left(\frac{y}{x}\right) - \pi & \text{if } x < 0 \text{ and } y < 0, \\ +\frac{\pi}{2} & \text{if } x = 0 \text{ and } y > 0, \\ -\frac{\pi}{2} & \text{if } x = 0 \text{ and } y < 0, \\ \text{undefined} & \text{if } x = 0 \text{ and } y = 0. \end{cases} \quad (\text{S14})$$

Second, we convert the azimuth angle $\tilde{\theta}$ and the elevation angle $\tilde{\varphi}$ to the dihedral angles $\tilde{\alpha}$ and $\tilde{\beta}$, scaling $\tilde{\alpha}$ and $\tilde{\beta}$ with Equation (S9), and obtain the parameters p and q . The final expressions of p and q become

$$p = \frac{1}{2} + \frac{2}{\pi}\tilde{\theta}, \quad q = \frac{1}{2} + \frac{2}{\pi} \arctan\left(\frac{\tan \tilde{\varphi}}{\cos \tilde{\theta}}\right). \quad (\text{S15})$$

Inversely, we can obtain the spatial location of a point with the parameters p and q . First, we calculate the elevation angle $\tilde{\theta}$ and the azimuth angle $\tilde{\varphi}$ by

$$\tilde{\theta} = \frac{\pi}{2}p - \frac{\pi}{4}, \quad \tilde{\varphi} = \arctan\left[\tan\left(\frac{\pi}{2}q - \frac{\pi}{4}\right) \cos \tilde{\theta}\right], \quad (\text{S16})$$

where $p, q \in [0, 1]$. Second, we calculate $(\tilde{x}, \tilde{y}, \tilde{z})$ by

$$\tilde{x} = R \cos \tilde{\varphi} \cos \tilde{\theta}, \quad \tilde{y} = R \cos \tilde{\varphi} \sin \tilde{\theta}, \quad \tilde{z} = R \sin \tilde{\varphi}. \quad (\text{S17})$$

Finally, we rotate the point \tilde{P} from the reference face to the face that the point P belongs to. The rotations can be expressed as

$$(x, y, z) = \begin{cases} (\tilde{x}, \tilde{y}, \tilde{z}) \cdot R_z\left(-\frac{\pi}{2}\right) & \text{if } k = 1, \\ (\tilde{x}, \tilde{y}, \tilde{z}) & \text{if } k = 2, \\ (\tilde{x}, \tilde{y}, \tilde{z}) \cdot R_z\left(\frac{\pi}{2}\right) & \text{if } k = 3, \\ (\tilde{x}, \tilde{y}, \tilde{z}) \cdot R_z(\pi) & \text{if } k = 4, \\ (\tilde{x}, \tilde{y}, \tilde{z}) \cdot R_z\left(-\frac{\pi}{2}\right) R_x\left(\frac{\pi}{2}\right) & \text{if } k = 5, \\ (\tilde{x}, \tilde{y}, \tilde{z}) \cdot R_z\left(-\frac{\pi}{2}\right) R_x\left(-\frac{\pi}{2}\right) & \text{if } k = 6. \end{cases} \quad (\text{S18})$$

Equations (S10)–(S15) map (x, y, z) to (p, q) , while Equations (S16)–(S18) map (p, q) to (x, y, z) . We define the following functions

$$(p, q) = \mathbf{f}_{\text{sphere-RS}}(x, y, z; k), \quad (\text{S19})$$

$$(x, y, z) = \mathbf{g}_{\text{sphere-RS}}(p, q; k, R). \quad (\text{S20})$$

The function $\mathbf{f}_{\text{sphere-RS}}$ represents the map given by Equations (S10)–(S15). The function $\mathbf{g}_{\text{sphere-RS}}$ represents the map given by Equations (S16)–(S18).

Sphere with UV mesh. Figure S4C shows a sphere with UV mesh in a Cartesian coordinate system. We can simply use the elevation angle θ and the azimuth angle φ to determine the position of the point P on the sphere. Here, we stipulate that the direction of the negative semi x axis corresponds to zero azimuth angle $\theta = 0$. Then we scale θ and φ onto the interval $[0, 1]$ by

$$p = \frac{\theta}{2\pi}, \quad q = \frac{\varphi}{\pi} + \frac{1}{2}. \quad (\text{S21})$$

We denote the coordinates of the point P by (x, y, z) . Then, p and q can be calculated by

$$p = \frac{1}{2\pi} \text{atan2}(y, x) + \frac{1}{2}, \quad q = \frac{1}{\pi} \text{atan2}(z, \sqrt{x^2 + y^2}) + \frac{1}{2}, \quad (\text{S22})$$

where the function atan2 is defined by Equation (S14). Inversely, we can use p and q to locate the point P by

$$x = -R \sin(\pi q) \cos(2\pi p), \quad y = -R \sin(\pi q) \sin(2\pi p), \quad z = -R \cos(\pi q), \quad (\text{S23})$$

where $p, q \in [0, 1]$ and R is the radius of the sphere. Equation (S22) maps (x, y, z) to (p, q) , while Equation (S23) maps (p, q) to (x, y, z) . We define the following functions

$$(p, q) = \mathbf{f}_{\text{sphere-UV}}(x, y, z), \quad (\text{S24})$$

$$(x, y, z) = \mathbf{g}_{\text{sphere-UV}}(p, q; R). \quad (\text{S25})$$

The function $\mathbf{f}_{\text{sphere-UV}}$ represents the map given by Equation (S22). The function $\mathbf{g}_{\text{sphere-UV}}$ represents the map given by Equation (S23).

Torus with UV mesh. Figure S4D shows a torus with a UV mesh (left) and its section view (right) in a Cartesian coordinate system. We denote the major radius of the torus by R and the minor radius

by r . The position of a point P on the torus is determined by two orientation angles θ and φ . The toroidal angle θ represents the orientation in the toroidal direction (around the circle of radius R) while the poloidal angle φ represents the orientation in the poloidal direction (around the circle of radius r). We stipulate that the zero toroidal angle ($\theta = 0$) corresponds to the direction of the negative semi x axis, and the zero poloidal angle ($\varphi = 0$) corresponds to the direction pointing to the centroid of the torus. Both the angles θ and φ are within the interval $[0, 2\pi]$, satisfying that $\theta = 0$ or 2π represents the same position in the toroidal direction, and $\varphi = 0$ or 2π represents the same position in the poloidal direction. We scale θ and φ onto the interval $[0, 1]$ by

$$p = \frac{\theta}{2\pi}, \quad q = \frac{\varphi}{2\pi}. \quad (\text{S26})$$

We denote the coordinates of the point P by (x, y, z) . The parameters p and q can be calculated by

$$p = \frac{1}{2\pi} \text{atan2}(y, x) + \frac{1}{2}, \quad q = \frac{1}{2\pi} \text{atan2}(z, \sqrt{x^2 + y^2} - R) + \frac{1}{2}, \quad (\text{S27})$$

where the function atan2 is defined by Equation (S14). Inversely, we can use p and q to locate the point P by

$$x = -[R - r \cos(2\pi q)] \cos(2\pi p), \quad y = -[R - r \cos(2\pi q)] \sin(2\pi p), \quad z = -r \sin(2\pi q), \quad (\text{S28})$$

where $p, q \in [0, 1]$. Equation (S27) maps (x, y, z) to (p, q) , while Equation (S28) maps (p, q) to (x, y, z) . We define the following functions

$$(p, q) = \mathbf{f}_{\text{torus-UV}}(x, y, z; R), \quad (\text{S29})$$

$$(x, y, z) = \mathbf{g}_{\text{torus-UV}}(p, q; R, r). \quad (\text{S30})$$

The function $\mathbf{f}_{\text{torus-UV}}$ represents the map given by Equation (S27). The function $\mathbf{g}_{\text{torus-UV}}$ represents the map given by Equation (S28).

S3 Regular patterns

Compact RS mesh. For each of the six faces of an RS mesh, (p, q) form another mesh in a square region $[0, 1] \times [0, 1]$ in the parameter space. We refer to this mesh in the parameter space as *RS-parameter mesh*. Figure S5A shows a 4×4 regular RS-parameter mesh, which can parameterize one face of an RS mesh. We assign indices to the square cells and nodes on the RS-parameter mesh,

as shown in Figure S5D. Supposing the RS-parameter mesh corresponds to the k -th face of the RS mesh, we use the triple (i, j, k) to index the square cell on the i -th column and the j -th row. For each square cell, we index its four nodes counterclockwise starting from the lower-left corner by 1 to 4. In general, for the square cell (i, j, k) on the $N \times N$ RS-parameter mesh, we denote the coordinates of the l -th node by $(p_{i,j,k,l}, q_{i,j,k,l})$, in which $i, j = 1, 2, \dots, N$; $k = 1, 2, \dots, 6$; $l = 1, 2, 3, 4$. In this work, for the simplicity of formulation, we restrict N to be a positive *even* integer. Then the coordinates $(p_{i,j,k,l}, q_{i,j,k,l})$ can be calculated as

$$\begin{aligned} p_{i,j,k,1} &= (i-1)/N, & p_{i,j,k,2} &= i/N, & p_{i,j,k,3} &= i/N, & p_{i,j,k,4} &= (i-1)/N, \\ q_{i,j,k,1} &= (j-1)/N, & q_{i,j,k,2} &= (j-1)/N, & q_{i,j,k,3} &= j/N, & q_{i,j,k,4} &= j/N. \end{aligned} \quad (\text{S31})$$

By substituting Equation (S31) into Equation (S7), we can map the parameters $(p_{i,j,k,l}, q_{i,j,k,l})$ to the spatial vertex positions, denoted by $(x_{i,j,k,l}, y_{i,j,k,l}, z_{i,j,k,l})$. The map can be expressed by

$$(x_{i,j,k,l}, y_{i,j,k,l}, z_{i,j,k,l}) = \mathbf{g}_{\text{cube-RS}}(p_{i,j,k,l}, q_{i,j,k,l}; k, L). \quad (\text{S32})$$

Through this map, the panels and vertices of the RS mesh inherit the indices i, j, k, l of the cells and nodes of the RS-parameter mesh. In addition, to express the assignment of the creases, we define the bi-value parameter

$$c_{i,j,k} = \begin{cases} +1, & \text{if the crease connects vertices 1 and 3,} \\ -1, & \text{if the crease connects vertices 2 and 4,} \end{cases} \quad (\text{S33})$$

where the triple (i, j, k) represents the panel to which the indices is assigned. Figure S3A shows an RS-parameter mesh which parametrizes all the six faces of an RS mesh. By setting $c_{i,j,k} = +1$ for all i, j, k , we can obtain a compact cube with a uniform crease assignment, as shown in Figure S3B. Furthermore, we can substitute Equation (S31) into Equation (S20) and obtain the vertex positions $(x_{i,j,k,l}, y_{i,j,k,l}, z_{i,j,k,l})$ on a sphere:

$$(x_{i,j,k,l}, y_{i,j,k,l}, z_{i,j,k,l}) = \mathbf{g}_{\text{sphere-RS}}(p_{i,j,k,l}, q_{i,j,k,l}; k, R). \quad (\text{S34})$$

Figure S3C shows the compact spherical RS mesh with a uniform crease assignment. The cube in Figure S3B and the sphere in Figure S3C share the same RS-parameter mesh in Figure S3A.

Deployed RS mesh. Figure S5B shows the RS-parameter mesh composed of 4×4 obliquely placed squares in the unit square region in the parameter space. The rotation angle ξ determines the nodal

coordinates of the square cells. As shown in Figure S5C, for an $N \times N$ RS-parameter mesh, each obliquely placed square cell occupies a larger square region of side length $1/N$ which is divided into two parts of lengths t_1 and t_2 . We have $t_1 + t_2 = 1/N$ and $t_1/t_2 = \tan(\xi/2)$. Therefore we can solve t_1 and t_2 by

$$t_1 = \frac{1}{N} \frac{\tan(\xi/2)}{1 + \tan(\xi/2)}, \quad t_2 = \frac{1}{N} \frac{1}{1 + \tan(\xi/2)}. \quad (\text{S35})$$

Then we can calculate the coordinates of the nodes in the deployed RS mesh. We follow the same index rules that are applied to the compact RS-parameter mesh. For the obliquely placed square cell (i, j, k) on the $N \times N$ RS-parameter mesh, the coordinates of the l -th node $(p_{i,j,k,l}, q_{i,j,k,l})$ can be calculated by

$$\begin{aligned} \text{for even } i + j: \quad & p_{i,j,k,1} = (i - 1)/N, & q_{i,j,k,1} &= (j - 1)/N + t_1, \\ & p_{i,j,k,2} = i/N - t_1, & q_{i,j,k,2} &= (j - 1)/N, \\ & p_{i,j,k,3} = i/N, & q_{i,j,k,3} &= j/N - t_1, \\ & p_{i,j,k,4} = (i - 1)/N + t_1, & q_{i,j,k,4} &= j/N; \\ \text{for odd } i + j: \quad & p_{i,j,k,1} = (i - 1)/N + t_1, & q_{i,j,k,1} &= (j - 1)/N, \\ & p_{i,j,k,2} = i/N, & q_{i,j,k,2} &= (j - 1)/N + t_1, \\ & p_{i,j,k,3} = i/N - t_1, & q_{i,j,k,3} &= j/N, \\ & p_{i,j,k,4} = (i - 1)/N, & q_{i,j,k,4} &= j/N - t_1. \end{aligned} \quad (\text{S36})$$

We can specify different values for ξ and obtain various deployed RS-parameter meshes. First, if we specify $\xi = 0$, the deployed RS-parameter mesh degenerates to the compact RS-parameter mesh. Second, specifying $\xi = \pi/3$ leads to the RS-parameter mesh shown in Figure S3D. In this case, we substitute Equation (S36) into Equation (S32) and generate the cubic RS mesh in Figure S3E. Also, we can substitute Equation (S36) into Equation (S34) and obtain the spherical RS mesh in Figure S3F. Third, specifying $\xi = \pi/2$ leads to the RS-parameter mesh in Figure S3G. In this case, we substitute Equation (S36) into Equation (S32) and Equation (S34), and then we can generate the cubic RS mesh as shown in Figure S3H and the spherical RS mesh as shown in Figure S3I, respectively.

Compact UV mesh. We refer to the mesh in the parameter space of a UV mesh as *UV-parameter mesh*. Figure S16A shows an 8×4 regular and compact UV-parameter mesh on the square region

$[0, 1] \times [0, 1]$. We assign indices to the cells and nodes on the compact UV-parameter mesh, as shown in Figure S16D. The tuple (i, j) is used to index the rectangle or triangle on the i -th column and the j -th row. For each rectangle, we index its four nodes counterclockwise starting from the lower-left corner by 1 to 4. For each triangle, we assign two indices to the nodes on the top or bottom boundary of the UV-parameter mesh, so that the indexing is consistent between the rectangles and triangles. As a result, the triangles can be considered as degenerate quadrilaterals in the sense that a double-indexed node of a triangle represents two coinciding nodes of a quadrilateral. Generally, an $M \times N$ UV-parameter mesh consists of $M \times N$ quadrilaterals (including $M \times 2$ degenerate triangles). We restrict M and N to be positive *even* integers. For the cell (i, j) on the $M \times N$ UV-parameter mesh, we denote the coordinates of the l -th node by $(p_{i,j,l}, q_{i,j,l})$, in which $i = 1, 2, \dots, M$; $j = 1, 2, \dots, N$; $l = 1, 2, 3, 4$. The coordinates $(p_{i,j,l}, q_{i,j,l})$ can be calculated by

$$\begin{aligned}
&\text{if } j = 1 \text{ and } l = 1, 2: \quad p_{i,j,1} = p_{i,j,2} = (2i - 1)/(2M), \quad q_{i,j,1} = q_{i,j,2} = 0; \\
&\text{if } j = N \text{ and } l = 3, 4: \quad p_{i,j,3} = p_{i,j,4} = (2i - 1)/(2M), \quad q_{i,j,3} = q_{i,j,4} = 1; \\
&\text{otherwise:} \quad p_{i,j,1} = (i - 1)/M, \quad q_{i,j,1} = (j - 1)/N, \\
&\quad p_{i,j,2} = i/M, \quad q_{i,j,2} = (j - 1)/N, \\
&\quad p_{i,j,3} = i/M, \quad q_{i,j,3} = j/N, \\
&\quad p_{i,j,4} = (i - 1)/M, \quad q_{i,j,4} = j/N.
\end{aligned} \tag{S37}$$

By substituting Equation (S37) into Equation (S25), we can map the parameters $(p_{i,j,l}, q_{i,j,l})$ to the spatial vertex positions, denoted by $(x_{i,j,l}, y_{i,j,l}, z_{i,j,l})$. The map can be expressed as

$$(x_{i,j,l}, y_{i,j,l}, z_{i,j,l}) = \mathbf{g}_{\text{sphere-UV}}(p_{i,j,l}, q_{i,j,l}; R). \tag{S38}$$

Through this map, the panels and vertices of the UV meshes inherit the indices i, j, l of the cells and nodes of the UV-parameter mesh. In addition, to express the assignment of the creases, we define the bi-value parameter

$$c_{i,j} = \begin{cases} +1, & \text{if the crease connects vertices 1 and 3,} \\ -1, & \text{if the crease connects vertices 2 and 4,} \end{cases} \tag{S39}$$

where $i = 1, 2, \dots, M$, $j = 2, 3, \dots, N - 1$, and the index (i, j) represents the quadrilaterals to which the assignment is applied. Figure S15A shows a UV-parameter mesh which can generate 16×8

compact UV meshes. By setting $c_{i,j} = +1$ for all i, j , we can obtain the compact spherical UV mesh demonstrated in Figure S15B.

Deployed UV mesh. Figure S16B shows the regular deployed UV-parameter mesh composed of 8×4 obliquely placed parallelograms and triangles in the unit square region in the parameter space. As shown in Figure S16C left, for an $M \times N$ UV-parameter mesh, each obliquely placed parallelogram occupies a larger rectangular region of width $1/M$ and height $1/(N-1)$ while each triangle occupies a rectangular region of width $1/M$ and height $1/(2N-2)$. The parallelogram cell divides the horizontal edge of the rectangle into u_1 and u_2 , and divides the vertical edge into v_1 and v_2 . We use Equation (S35) to determine u_1 and u_2 :

$$u_1 = \frac{1}{M} \frac{\tan(\xi/2)}{1 + \tan(\xi/2)}, \quad u_2 = \frac{1}{M} \frac{1}{1 + \tan(\xi/2)}. \quad (\text{S40})$$

On the vertical side, v_1 and v_2 are determined by

$$v_1 = \frac{1}{N-1} \frac{\tan(\xi/2)}{1 + \tan(\xi/2)}, \quad v_2 = \frac{1}{N-1} \frac{1}{1 + \tan(\xi/2)}. \quad (\text{S41})$$

The triangular cell divides the horizontal edge of the rectangle into u_1 and u_2 and divides the vertical edge into v_1 and \hat{v}_2 . The expressions of u_1 , u_2 , and v_1 are the same as those for the parallelogram cells, while \hat{v}_2 is expressed by

$$\hat{v}_2 = \frac{1}{2N-2} \frac{1 - \tan(\xi/2)}{1 + \tan(\xi/2)}. \quad (\text{S42})$$

We note that in Equations (S40)–(S42), the variable ξ is not any existing angle in the deployed UV-parameter mesh, but is only a parameter that controls the deployment of the pattern. If $\xi = 0$, u_1 and v_1 equal zero, and therefore lead to a compact UV-parameter mesh. Next, we give the coordinates of the nodes of the parallelogram and triangles. To this end, we index the cells and nodes as shown in Figure S16E. The index rules are the same as those applied to the compact UV-parameter mesh, where the triangles are considered as degenerate quadrilaterals assigned four nodal indices. For the cell (i, j) on the deployed $M \times N$ UV-parameter mesh, the coordinates of the

l -th node $(p_{i,j,l}, q_{i,j,l})$ can be calculated by

for even $i + j$:

$$\text{if } j = 1 \text{ and } l = 1, 2: \quad p_{i,j,1} = p_{i,j,2} = (i - 1)/M, \quad q_{i,j,1} = q_{i,j,2} = 0;$$

$$\text{if } j = N \text{ and } l = 3, 4: \quad p_{i,j,3} = p_{i,j,4} = (i - 1)/M, \quad q_{i,j,3} = q_{i,j,4} = 1;$$

$$\text{otherwise:} \quad p_{i,j,1} = (i - 1)/M, \quad q_{i,j,1} = (j - 1)/N + v_1,$$

$$p_{i,j,2} = i/M - u_1, \quad q_{i,j,2} = (j - 1)/N,$$

$$p_{i,j,3} = i/M, \quad q_{i,j,3} = j/N - v_1,$$

$$p_{i,j,4} = (i - 1)/M + u_1, \quad q_{i,j,4} = j/N;$$

for odd $i + j$:

$$\text{if } j = 1 \text{ and } l = 1, 2: \quad p_{i,j,1} = p_{i,j,2} = i/M, \quad q_{i,j,1} = q_{i,j,2} = 0;$$

$$\text{if } j = N \text{ and } l = 3, 4: \quad p_{i,j,3} = p_{i,j,4} = i/M, \quad q_{i,j,3} = q_{i,j,4} = 1;$$

$$\text{otherwise:} \quad p_{i,j,1} = (i - 1)/M + u_1, \quad q_{i,j,1} = (j - 1)/N, \quad (\text{S43})$$

$$p_{i,j,2} = i/M, \quad q_{i,j,2} = (j - 1)/N + v_1,$$

$$p_{i,j,3} = i/M - u_1, \quad q_{i,j,3} = j/N,$$

$$p_{i,j,4} = (i - 1)/M, \quad q_{i,j,4} = j/N - v_1.$$

Then we can substitute Equation (S43) into Equation (S30), mapping the parameters $(p_{i,j,l}, q_{i,j,l})$ to the spatial vertex positions, denoted by $(x_{i,j,l}, y_{i,j,l}, z_{i,j,l})$. The map can be expressed as

$$(x_{i,j,l}, y_{i,j,l}, z_{i,j,l}) = \mathbf{g}_{\text{torus-UV}}(p_{i,j,l}, q_{i,j,l}; R, r). \quad (\text{S44})$$

We can assign different values to ξ and obtain various deployed UV-parameter meshes. If we specify $\xi = \pi/3$, we can obtain the UV-parameter mesh in Figure S15C. In this case, we substitute Equation (S43) into Equation (S44) and then generate the toric UV mesh in Figure S15D. If we specify $\xi = \pi/2$, we can obtain the UV-parameter mesh in Figure S15E. In this case, we substitute Equation (S43) into Equation (S44) and then generate the toric UV mesh in Figure S15F.

S4 Optimization for compatibility

Independent parameters for compact RS mesh. Consider an RS mesh enclosing a cube of half side length L or a sphere of radius R . The RS mesh can be determined by the size variable L or R

and the elements in the following set

$$\mathbf{P}_{\text{RS}} = \{(p_{i,j,k,l}, q_{i,j,k,l}) \mid i, j = 1, 2, \dots, N; k = 1, 2, \dots, 6; l = 1, 2, 3, 4\}, \quad (\text{S45})$$

where $p_{i,j,k,l}$ and $q_{i,j,k,l}$ are the parameters that determine the vertex positions. In terms of optimization variables, however, there are numerous redundant parameters in \mathbf{P}_{RS} due to the following reasons. (1) The common vertices of adjacent panels are at the same position and therefore share the same parameters. (2) Some vertices are restricted to move only on the boundaries and therefore have only one independent parameter. (3) The vertices at both ends of a boundary are fixed (see Equations (S1) and (S8)) and therefore have zero independent parameter. Considering the reasons above, we classify the vertices of an RS mesh into three categories: the interior vertices, the boundary vertices, and the corner vertices. Figure S6A shows a 4×4 compact RS-parameter mesh with the independent local indices of the interior, boundary, and corner vertices. Compared with the indices in Figure S5D, we have removed the redundant indices such that only one index is reserved for the vertices at the same position. We can see that on the interior region, each index corresponds to four vertices; on the boundaries, each index corresponds to two vertices; at the corner, the index and vertex have a one-to-one correspondence. We recall that on the continuous intersections (see edges 1 to 8 in Figure S2C) of an RS mesh, the adjacent panels, which belong to different faces, are connected by their common vertices. These common vertices are also located at the same position so that there exist additional redundant parameters on the continuous intersections. Now we extract the independent parameters from \mathbf{P}_{RS} for a general compact $N \times N \times 6$ RS mesh. First, the interior vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{int.-c-RS}} = \{(p_{i,j,k,l}, q_{i,j,k,l}) \mid i, j = 2, 3, \dots, N, k = 1, 2, \dots, 6, l = 1\}. \quad (\text{S46})$$

Second, the boundary vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{bdy.-c-RS}} = \mathbf{P}_{\text{bdy.-c-RS},1} \cup \mathbf{P}_{\text{bdy.-c-RS},2} \cup \mathbf{P}_{\text{bdy.-c-RS},3} \cup \mathbf{P}_{\text{bdy.-c-RS},4}, \quad (\text{S47})$$

where the four subsets $\mathbf{P}_{\text{bdy.-c-RS},1}$, $\mathbf{P}_{\text{bdy.-c-RS},2}$, $\mathbf{P}_{\text{bdy.-c-RS},3}$, and $\mathbf{P}_{\text{bdy.-c-RS},4}$ correspond to the bottom, right, top, and left boundaries of the RS mesh, respectively, and their elements are given by

$$\begin{aligned}\mathbf{P}_{\text{bdy.-c-RS},1} &= \{(p_{i,j,k,l}, 0) \mid i = 2, 3, \dots, N; j = 1; k = 1, 2, 3, 4; l = 1\}, \\ \mathbf{P}_{\text{bdy.-c-RS},2} &= \{(1, q_{i,j,k,l}) \mid i = N; j = 2, 3, \dots, N; k = 5, 6; l = 2\}, \\ \mathbf{P}_{\text{bdy.-c-RS},3} &= \{(p_{i,j,k,l}, 1) \mid i = 1, 2, \dots, N-1; j = N; k = 1, 2, 3, 4; l = 3\}, \\ \mathbf{P}_{\text{bdy.-c-RS},4} &= \{(0, q_{i,j,k,l}) \mid i = 1; j = 1, 2, \dots, N-1; k = 1, 2, \dots, 6; l = 4\}.\end{aligned}\tag{S48}$$

We follow the global edge indices in Figure S2C. Then, the subset $\mathbf{P}_{\text{bdy.-c-RS},1}$ represents the boundaries 1, 3, 9, and 11; the subset $\mathbf{P}_{\text{bdy.-c-RS},2}$ represents the boundaries 10 and 14; the subset $\mathbf{P}_{\text{bdy.-c-RS},3}$ represents the boundaries 2, 4, 13, and 15; the subset $\mathbf{P}_{\text{bdy.-c-RS},4}$ represents the boundaries 5, 6, 7, 8, 12, and 16. Third, the corner vertices are fixed, corresponding to the following set of constant elements:

$$\mathbf{P}_{\text{cor.-c-RS}} = \{(0, 0)_k, (1, 0)_k, (1, 1)_k, (0, 1)_k \mid k = 1, 3\}.\tag{S49}$$

Altogether, we define the following union set

$$\mathbf{P}_{\text{c-RS}} = \mathbf{P}_{\text{int.-c-RS}} \cup \mathbf{P}_{\text{bdy.-c-RS}} \cup \mathbf{P}_{\text{cor.-c-RS}},\tag{S50}$$

which has no redundant parameters and determines all the vertex positions. Importantly, we have the following equivalence

$$\mathbf{P}_{\text{c-RS}} \sim \mathbf{P}_{\text{RS}},\tag{S51}$$

in the sense that the parameters determining the same vertex positions are considered to be equivalent with each other. This equivalence can also be written in the expanded form as

for interior vertices:

$$\begin{aligned}p_{i,j,k,1} &= p_{i-1,j,k,2} = p_{i-1,j-1,k,3} = p_{i,j-1,k,4}, \quad i, j > 1; \quad k = 1, 2, \dots, 6; \\ q_{i,j,k,1} &= q_{i-1,j,k,2} = q_{i-1,j-1,k,3} = q_{i,j-1,k,4}, \quad i, j > 1; \quad k = 1, 2, \dots, 6;\end{aligned}$$

for boundary vertices on the same face:

$$\begin{aligned}
p_{i,1,k,1} &= p_{i-1,1,k,2}, & q_{i,1,k,1} &= q_{i-1,1,k,2} = 0, & i > 1; & k = 1, 2, \dots, 6; \\
q_{N,j,k,2} &= q_{N,j-1,k,3}, & p_{N,j,k,2} &= p_{N,j-1,k,3} = 0, & j > 1; & k = 1, 2, \dots, 6; \\
p_{i,N,k,3} &= p_{i+1,N,k,4}, & q_{i,N,k,3} &= q_{i+1,N,k,4} = 0, & i < N; & k = 1, 2, \dots, 6; \\
q_{1,j,k,4} &= q_{1,j+1,k,1}, & p_{1,j,k,4} &= p_{1,j+1,k,1} = 0, & j < N; & k = 1, 2, \dots, 6;
\end{aligned}$$

for boundary vertices on adjacent faces:

$$\begin{aligned}
p_{i,1,k,1} &= p_{i,N,k',4}, & i > 1; & k = 1, k' = 5; \\
p_{i,1,k,1} &= p_{i,N,k',4}, & i > 1; & k = 6, k' = 1; \\
p_{i,1,k,1} &= p_{N-i+1,1,k',2}, & i > 1; & k = 3, k' = 5; \\
p_{i,N,k,3} &= p_{N-i+1,N,k',4}, & i < N; & k = 3, k' = 6; \\
q_{N,j,k,2} &= q_{1,j,k',1}, & j > 1; & k = 4, k' = 1; \\
q_{N,j,k,2} &= q_{1,j,k',1}, & j > 1; & k = 1, k' = 2; \\
q_{N,j,k,2} &= q_{1,j,k',1}, & j > 1; & k = 2, k' = 3; \\
q_{N,j,k,2} &= q_{1,j,k',1}, & j > 1; & k = 3, k' = 4;
\end{aligned}$$

for corner vetices:

$$\begin{aligned}
p_{1,1,k,1} &= 0, & q_{1,1,k,1} &= 0, & k = 1, 2, \dots, 6; \\
p_{N,1,k,1} &= 1, & q_{N,1,k,1} &= 0, & k = 1, 2, \dots, 6; \\
p_{N,N,k,1} &= 1, & q_{N,N,k,1} &= 1, & k = 1, 2, \dots, 6; \\
p_{1,N,k,1} &= 0, & q_{1,N,k,1} &= 1, & k = 1, 2, \dots, 6;
\end{aligned} \tag{S52}$$

where $i, j = 1, 2, \dots, N$. Inserting the parameters in $\mathbf{P}_{\text{c-RS}}$ into Equation (S7) or (S20), we can obtain all the vertex positions of a compact RS mesh. We collect the spatial coordinates of all these vertices to construct the following set:

$$\mathbf{X}_{\text{c-RS}} = \{(x_{i,j,k,l}, y_{i,j,k,l}, z_{i,j,k,l}) \mid (i, j, k, l) \in \mathbf{I}_{\text{c-RS}}\}, \tag{S53}$$

where the set $\mathbf{I}_{\text{c-RS}}$ contains indices (i, j, k, l) corresponding to the independent parameters $(p_{i,j,k,l}, q_{i,j,k,l}) \in \mathbf{P}_{\text{c-RS}}$. Then, we define a function $\mathbf{g}_{\text{c-RS}}$ to express the map from $\mathbf{P}_{\text{c-RS}}$ to

$\mathbf{X}_{\text{c-RS}}$ as

$$\mathbf{x}_c = \mathbf{g}_{\text{c-RS}}(\mathbf{p}_c; a_{\text{c-RS}}); \mathbf{x}_c \in \mathbf{X}_{\text{c-RS}}, \mathbf{p}_c \in \mathbf{P}_{\text{c-RS}}, \quad (\text{S54})$$

where the variable $a_{\text{c-RS}}$ is the size of the surface (L or R) with the compact RS mesh. Inversely, with Equation (S6) or (S19), we can define a function $\mathbf{f}_{\text{c-RS}}$ to express the map from $\mathbf{X}_{\text{c-RS}}$ to $\mathbf{P}_{\text{c-RS}}$ as

$$\mathbf{p}_c = \mathbf{f}_{\text{c-RS}}(\mathbf{x}_c; a_{\text{c-RS}}); \mathbf{p}_c \in \mathbf{P}_{\text{c-RS}}, \mathbf{x}_c \in \mathbf{X}_{\text{c-RS}}. \quad (\text{S55})$$

Independent parameters for deployed RS mesh. When an RS mesh is deployed from the compact state, some vertices at the same position will move apart, and therefore, their parameters are no longer identical. As a result, we have more independent parameters. Figure S6B shows a 4×4 deployed RS-parameter mesh and the local indices corresponding to the interior, boundary, and corner vertices. We can see that on the interior region, each index corresponds to two vertices; on the boundaries, the index and vertex have a one-to-one correspondence. It is worth noting that the previous corner vertices on the compact mesh become boundary vertices on the deployed mesh because they move away from the two ends on the boundaries due to deployment. We recall that the set \mathbf{P}_{RS} given by Equation (S45) contains all the parameter pairs that determine an RS mesh. In general, for a deployed $N \times N \times 6$ RS mesh, we extract the independent parameters from the set \mathbf{P}_{RS} in the following way. First, the interior vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{int.-d-RS}} = \mathbf{P}_{\text{int.-d-RS},1} \cup \mathbf{P}_{\text{int.-d-RS},2} \cup \mathbf{P}_{\text{int.-d-RS},3} \cup \mathbf{P}_{\text{int.-d-RS},4}, \quad (\text{S56})$$

where the subsets $\mathbf{P}_{\text{int.-d-RS},1}$ and $\mathbf{P}_{\text{int.-d-RS},2}$ correspond to the panels that rotate clockwise (even $i + j$) when deployed, and the subsets $\mathbf{P}_{\text{int.-d-RS},3}$ and $\mathbf{P}_{\text{int.-d-RS},4}$ correspond to the panels that rotate counterclockwise (odd $i + j$), and their elements are given by

$$\begin{aligned} \mathbf{P}_{\text{int.-d-RS},1} &= \{(p_{i,j,k,l}, q_{i,j,k,l}) \mid \text{even } i + j, i < N, l = 3\}, \\ \mathbf{P}_{\text{int.-d-RS},2} &= \{(p_{i,j,k,l}, q_{i,j,k,l}) \mid \text{even } i + j, j < N, l = 4\}, \\ \mathbf{P}_{\text{int.-d-RS},3} &= \{(p_{i,j,k,l}, q_{i,j,k,l}) \mid \text{odd } i + j, i < N; l = 2\}, \\ \mathbf{P}_{\text{int.-d-RS},4} &= \{(p_{i,j,k,l}, q_{i,j,k,l}) \mid \text{odd } i + j, j < N; l = 3\}. \end{aligned} \quad (\text{S57})$$

In the definitions above, we have $i, j = 1, 2, \dots, N$ and $k = 1, 2, \dots, 6$. Second, the boundary vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{bdy.-d-RS}} = \mathbf{P}_{\text{bdy.-d-RS},1} \cup \mathbf{P}_{\text{bdy.-d-RS},2} \cup \mathbf{P}_{\text{bdy.-d-RS},3} \cup \mathbf{P}_{\text{bdy.-d-RS},4}, \quad (\text{S58})$$

where the four subsets $\mathbf{P}_{\text{bdy.-d-RS},1}$, $\mathbf{P}_{\text{bdy.-d-RS},2}$, $\mathbf{P}_{\text{bdy.-d-RS},3}$, and $\mathbf{P}_{\text{bdy.-d-RS},4}$ correspond to the bottom, right, top, and left boundaries of the meshes, respectively, and their elements are given as follows

$$\begin{aligned}\mathbf{P}_{\text{bdy.-d-RS},1} &= \{(p_{i,j,k,l}, 0) \mid l = 2, \text{ even } i + j \text{ or } l = 1, \text{ odd } i + j; k = 1, 2, 3, 4\}, \\ \mathbf{P}_{\text{bdy.-d-RS},2} &= \{(1, q_{i,j,k,l}) \mid l = 3, \text{ even } i + j \text{ or } l = 2, \text{ odd } i + j; k = 5, 6\}, \\ \mathbf{P}_{\text{bdy.-d-RS},3} &= \{(p_{i,j,k,l}, 1) \mid l = 4, \text{ even } i + j \text{ or } l = 3, \text{ odd } i + j; k = 1, 2, 3, 4\}, \\ \mathbf{P}_{\text{bdy.-d-RS},4} &= \{(0, q_{i,j,k,l}) \mid l = 1, \text{ even } i + j \text{ or } l = 4, \text{ odd } i + j; k = 1, 2, \dots, 6\},\end{aligned}\tag{S59}$$

in which $i, j = 1, 2, \dots, N$. In analogy to the compact meshes, we have removed the redundant parameters on the continuous intersections—the subset $\mathbf{P}_{\text{bdy.-d-RS},1}$ corresponds to the boundaries 1, 3, 9, and 11 shown in Figure S2C; the subset $\mathbf{P}_{\text{bdy.-d-RS},2}$ corresponds to the boundaries 10 and 14; the subset $\mathbf{P}_{\text{bdy.-d-RS},3}$ corresponds to the boundaries 2, 4, 13, and 15; the subset $\mathbf{P}_{\text{bdy.-d-RS},4}$ corresponds to the boundaries 5, 6, 7, 8, 12, and 16. The following procedures is very similar to those for the compact patterns. We define the following union set

$$\mathbf{P}_{\text{d-RS}} = \mathbf{P}_{\text{int.-d-RS}} \cup \mathbf{P}_{\text{bdy.-d-RS}},\tag{S60}$$

and then we have the following equivalence

$$\mathbf{P}_{\text{d-RS}} \sim \mathbf{P}_{\text{RS}}.\tag{S61}$$

This equivalence can also be written in the expanded form as

for interior vertices:

$$\begin{aligned}p_{i,j,k,3} &= p_{i+1,j,k,4}, & q_{i,j,k,3} &= q_{i+1,j,k,4}, & \text{even } i + j, & i < N; & k = 1, 2, \dots, 6; \\ p_{i,j,k,4} &= p_{i,j+1,k,1}, & q_{i,j,k,4} &= q_{i,j+1,k,1}, & \text{even } i + j, & j < N; & k = 1, 2, \dots, 6; \\ p_{i,j,k,2} &= p_{i+1,j,k,1}, & q_{i,j,k,2} &= q_{i+1,j,k,1}, & \text{odd } i + j, & i < N; & k = 1, 2, \dots, 6; \\ p_{i,j,k,3} &= p_{i,j+1,k,2}, & q_{i,j,k,3} &= q_{i,j+1,k,2}, & \text{odd } i + j, & j < N; & k = 1, 2, \dots, 6;\end{aligned}$$

for boundary vertices on the same face:

$$\begin{aligned}q_{i-1,1,k,1} &= q_{i,1,k,2} = 0, & \text{even } i, & k = 1, 2, \dots, 6; \\ p_{N,j-1,k,2} &= p_{N,j,k,3} = 0, & \text{even } j, & k = 1, 2, \dots, 6; \\ q_{i-1,N,k,3} &= q_{i,N,k,4} = 0, & \text{even } i, & k = 1, 2, \dots, 6; \\ p_{1,j-1,k,1} &= p_{1,j,k,4} = 0, & \text{even } j, & k = 1, 2, \dots, 6;\end{aligned}$$

for boundary vertices on adjacent faces:

$$\begin{aligned}
p_{i,1,k,2} &= p_{i,N,k',3}, \text{ odd } i; & p_{i,1,k,1} &= p_{i,N,k',4}, \text{ even } i; & k &= 1, k' = 5; \\
p_{i,1,k,2} &= p_{i,N,k',3}, \text{ odd } i; & p_{i,1,k,1} &= p_{i,N,k',4}, \text{ even } i; & k &= 6, k' = 1; \\
p_{i,1,k,2} &= p_{N-i+1,1,k',1}, \text{ odd } i; & p_{i,1,k,1} &= p_{N-i+1,1,k',2}, \text{ even } i; & k &= 3, k' = 5; \\
p_{i,N,k,3} &= p_{N-i+1,N,k',4}, \text{ odd } i; & p_{i,N,k,4} &= p_{N-i+1,N,k',3}, \text{ even } i; & k &= 3, k' = 6; \\
q_{N,j,k,2} &= q_{1,j,k',1}, \text{ odd } j; & q_{N,j,k,3} &= q_{1,j,k',4}, \text{ even } j; & k &= 4, k' = 1; \\
q_{N,j,k,2} &= q_{1,j,k',1}, \text{ odd } j; & q_{N,j,k,3} &= q_{1,j,k',4}, \text{ even } j; & k &= 1, k' = 2; \\
q_{N,j,k,2} &= q_{1,j,k',1}, \text{ odd } j; & q_{N,j,k,3} &= q_{1,j,k',4}, \text{ even } j; & k &= 2, k' = 3; \\
q_{N,j,k,2} &= q_{1,j,k',1}, \text{ odd } j; & q_{N,j,k,3} &= q_{1,j,k',4}, \text{ even } j; & k &= 3, k' = 4;
\end{aligned} \tag{S62}$$

where $i, j = 1, 2, \dots, N$. The set $\mathbf{P}_{\text{d-RS}}$ has no redundant parameters and determines all the vertex positions of a deployed RS mesh when substituted into Equation (S7) or Equation (S20). We collect the spatial coordinates of all these vertices to construct the following set

$$\mathbf{X}_{\text{d-RS}} = \{(x_{i,j,k,l}, y_{i,j,k,l}, z_{i,j,k,l}) \mid (i, j, k, l) \in \mathbf{I}_{\text{d-RS}}\}, \tag{S63}$$

where the set $\mathbf{I}_{\text{d-RS}}$ contains indices (i, j, k, l) corresponding to the parameters $(p_{i,j,k,l}, q_{i,j,k,l}) \in \mathbf{P}_{\text{d-RS}}$. Then we define a function $\mathbf{g}_{\text{d-RS}}$ to express the map from $\mathbf{P}_{\text{d-RS}}$ to $\mathbf{X}_{\text{d-RS}}$ as

$$\mathbf{x}_d = \mathbf{g}_{\text{d-RS}}(\mathbf{p}_d; a_{\text{d-RS}}); \mathbf{x}_d \in \mathbf{X}_{\text{d-RS}}, \mathbf{p}_d \in \mathbf{P}_{\text{d-RS}}, \tag{S64}$$

where the variable $a_{\text{d-RS}}$ is the size of the surface (L or R) with the deployed RS mesh. Inversely, using Equation (S6) or (S19), we can define a function $\mathbf{f}_{\text{d-RS}}$ to express the map from $\mathbf{X}_{\text{d-RS}}$ to $\mathbf{P}_{\text{d-RS}}$ as

$$\mathbf{p}_d = \mathbf{f}_{\text{d-RS}}(\mathbf{x}_d; a_{\text{d-RS}}); \mathbf{p}_d \in \mathbf{P}_{\text{d-RS}}, \mathbf{x}_d \in \mathbf{X}_{\text{d-RS}}. \tag{S65}$$

Independent parameters for compact UV mesh. Consider an UV mesh enclosing a sphere of radius R or a torus of major radius R and minor radius r . The UV mesh can be determined by the radii R and r and the elements in the following set

$$\mathbf{P}_{\text{UV}} = \{(p_{i,j,l}, q_{i,j,l}) \mid i = 1, 2, \dots, M; j = 1, 2, \dots, N; l = 1, 2, 3, 4\}, \tag{S66}$$

where $p_{i,j,l}$ and $q_{i,j,l}$ are the parameters that determine the vertex positions. For a compact UV mesh, as shown in Figure S17A, we can classify the vertices into interior vertices and polar vertices.

In the interior region, the common vertices of adjacent panels are at the same position and therefore redundant parameters emerge here. The polar vertices are fixed at the two polar points of the sphere, so that there is no effective parameter. We extract the independent parameters from \mathbf{P}_{UV} of a compact $M \times N$ UV mesh. We recall that M and N are assumed to be positive even integers. First, the interior vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{int.-c-UV}} = \{(p_{i,j,l}, q_{i,j,l}) \mid i = 1, 2, \dots, M; j = 2, 3, \dots, N; l = 1\}. \quad (\text{S67})$$

Second, the polar vertices are fixed, corresponding to the following set of constant elements

$$\mathbf{P}_{\text{polar-c-UV}} = \{(0, 0), (0, 1)\}. \quad (\text{S68})$$

Altogether, we define the following union set

$$\mathbf{P}_{\text{c-UV}} = \mathbf{P}_{\text{int.-c-UV}} \cup \mathbf{P}_{\text{polar-c-UV}}, \quad (\text{S69})$$

which has no redundant parameters and determines all the vertex positions of a compact UV mesh.

We have the following equivalence

$$\mathbf{P}_{\text{c-UV}} \sim \mathbf{P}_{\text{UV}}, \quad (\text{S70})$$

which can be written in the expanded form as

for interior vertices:

$$p_{i,j,1} = p_{i-1,j,2} = p_{i-1,j-1,3} = p_{i,j-1,4}, \quad j > 1;$$

$$q_{i,j,1} = q_{i-1,j,2} = q_{i-1,j-1,3} = q_{i,j-1,4}, \quad j > 1;$$

for polar vertices:

$$p_{i,1,1} = p_{i,1,2} = 0, \quad q_{i,1,1} = q_{i,1,2} = 0; \quad (\text{S71})$$

$$p_{i,N,3} = p_{i,N,4} = 0, \quad q_{i,N,3} = q_{i,N,4} = 1;$$

where $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$, and we stipulate that $i - 1 = M$ for $i = 1$. Then, by inserting the parameters in $\mathbf{P}_{\text{c-UV}}$ to Equation (S25), we can obtain all the vertex positions of a compact UV mesh. We collect the spatial coordinates of all these vertices to construct the following set

$$\mathbf{X}_{\text{c-UV}} = \{(x_{i,j,l}, y_{i,j,l}, z_{i,j,l}) \mid (i, j, l) \in \mathbf{I}_{\text{c-UV}}\}, \quad (\text{S72})$$

where the set $\mathbf{I}_{\text{c-UV}}$ contains the indices (i, j, l) corresponding to the independent parameters $(p_{i,j,l}, q_{i,j,l}) \in \mathbf{P}_{\text{c-UV}}$. We define a function $\mathbf{g}_{\text{c-UV}}$ to express the map from $\mathbf{P}_{\text{c-UV}}$ to $\mathbf{X}_{\text{c-UV}}$ as

$$\mathbf{x}_{\text{c}} = \mathbf{g}_{\text{c-UV}}(\mathbf{p}_{\text{c}}; a_{\text{c-UV}}); \mathbf{x}_{\text{c}} \in \mathbf{X}_{\text{c-UV}}, \mathbf{p}_{\text{c}} \in \mathbf{P}_{\text{c-UV}}, \quad (\text{S73})$$

where the variable $a_{\text{c-UV}}$ is the radius R of the sphere with the compact UV mesh. Inversely, using Equation (S24), we can define a function $\mathbf{f}_{\text{c-UV}}$ to express the map from $\mathbf{X}_{\text{c-UV}}$ to $\mathbf{P}_{\text{c-UV}}$ as

$$\mathbf{p}_{\text{c}} = \mathbf{f}_{\text{c-UV}}(\mathbf{x}_{\text{c}}; a_{\text{c-UV}}); \mathbf{p}_{\text{c}} \in \mathbf{P}_{\text{c-UV}}, \mathbf{x}_{\text{c}} \in \mathbf{X}_{\text{c-UV}}. \quad (\text{S74})$$

Independent parameters for deployed UV mesh. Now we consider a deployed UV mesh enclosing a torus of major radius R and minor radius r . As shown in Figure S17B, we can classify the vertices into the interior vertices and the boundary vertices. We note that the previous polar vertices on the compact pattern become boundary vertices on the deployed pattern because they move from the polar points to the inner circumference after the deployment. We can see that either in the interior region or on the boundary, each index corresponds to two vertices. However, the boundary vertices are different from the interior vertices because they only move on the inner circumference (that is, the intersection) during the optimization. We recall that the set \mathbf{P}_{UV} given by Equation (S66) contain parameters that determine an UV mesh. In general, for a deployed $M \times N$ UV mesh, we extract the independent parameters from \mathbf{P}_{UV} in the following way. First, the interior vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{int.-d-UV}} = \mathbf{P}_{\text{int.-d-UV},1} \cup \mathbf{P}_{\text{int.-d-UV},2} \cup \mathbf{P}_{\text{int.-d-UV},3} \cup \mathbf{P}_{\text{int.-d-UV},4}, \quad (\text{S75})$$

where the subsets $\mathbf{P}_{\text{int.-d-UV},1}$ and $\mathbf{P}_{\text{int.-d-UV},2}$ correspond to the panels that rotate clockwise (even $i + j$) when deployed, and the subsets $\mathbf{P}_{\text{int.-d-UV},3}$ and $\mathbf{P}_{\text{int.-d-UV},4}$ correspond to the panels that rotate counterclockwise (odd $i + j$). Their elements are given by

$$\begin{aligned} \mathbf{P}_{\text{int.-d-UV},1} &= \{(p_{i,j,l}, q_{i,j,l}) \mid \text{even } i + j, j > 1, l = 1\}, \\ \mathbf{P}_{\text{int.-d-UV},2} &= \{(p_{i,j,l}, q_{i,j,l}) \mid \text{even } i + j, j < N, l = 4\}, \\ \mathbf{P}_{\text{int.-d-UV},3} &= \{(p_{i,j,l}, q_{i,j,l}) \mid \text{odd } i + j, j < N, l = 3\}, \\ \mathbf{P}_{\text{int.-d-UV},4} &= \{(p_{i,j,l}, q_{i,j,l}) \mid \text{odd } i + j, j < N, l = 4\}, \end{aligned} \quad (\text{S76})$$

in which $i, j = 1, 2, \dots, N$. Second, the boundary vertices are parameterized by the elements in the following set

$$\mathbf{P}_{\text{bdy.-d-UV}} = \mathbf{P}_{\text{bdy.-d-UV},1} \cup \mathbf{P}_{\text{bdy.-d-UV},2}, \quad (\text{S77})$$

where the subsets $\mathbf{P}_{\text{bdy.-d-UV},1}$ and $\mathbf{P}_{\text{bdy.-d-UV},2}$ correspond to the bottom ($q_{i,j,l} = 0$) and top ($q_{i,j,l} = 1$) boundaries of the meshes, respectively, and their elements are given by

$$\begin{aligned}\mathbf{P}_{\text{bdy.-d-UV},1} &= \{(p_{i,j,l}, 0) \mid i = 1, 3, \dots, M-1; j = 1; l = 1\}, \\ \mathbf{P}_{\text{bdy.-d-UV},2} &= \{(p_{i,j,l}, 1) \mid i = 1, 3, \dots, M-1; j = N; l = 3\}.\end{aligned}\tag{S78}$$

The following procedures are very similar to those for the compact meshes. We define the following union set

$$\mathbf{P}_{\text{d-UV}} = \mathbf{P}_{\text{int.-d-UV}} \cup \mathbf{P}_{\text{bdy.-d-UV}},\tag{S79}$$

and then we have the following equivalence

$$\mathbf{P}_{\text{d-UV}} \sim \mathbf{P}_{\text{UV}},\tag{S80}$$

which can be written in the expanded form as

for interior vertices:

$$\begin{aligned}p_{i,j,1} &= p_{i-1,j,2}, \quad q_{i,j,1} = q_{i-1,j,2}, \quad \text{even } i+j, \quad j > 1; \\ p_{i,j,4} &= p_{i,j+1,1}, \quad q_{i,j,4} = q_{i,j+1,1}, \quad \text{even } i+j, \quad j < N; \\ p_{i,j,3} &= p_{i,j+1,2}, \quad q_{i,j,3} = q_{i,j+1,2}, \quad \text{odd } i+j, \quad j < N; \\ p_{i,j,4} &= p_{i-1,j,3}, \quad q_{i,j,4} = q_{i-1,j,3}, \quad \text{odd } i+j, \quad j < N;\end{aligned}$$

for boundary vertices:

$$\begin{aligned}p_{i,1,1} &= p_{i,1,2} = p_{i-1,1,1} = p_{i-1,1,2}, \quad \text{odd } i; \\ p_{i,N,3} &= p_{i,N,4} = p_{i-1,N,3} = p_{i-1,N,4}, \quad \text{odd } i; \\ q_{i,1,1} &= q_{i,1,2} = 0; \\ q_{i,N,3} &= q_{i,N,4} = 1,\end{aligned}\tag{S81}$$

where $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$, and we stipulate that $i-1 = M$ for $i = 1$. The set $\mathbf{P}_{\text{d-UV}}$ has no redundant parameters and determines all the vertex positions of a deployed UV mesh when substituted into Equation (S30). We collect the spatial coordinates of all these vertices to construct the following set

$$\mathbf{X}_{\text{d-UV}} = \{(x_{i,j,l}, y_{i,j,l}, z_{i,j,l}) \mid (i, j, l) \in \mathbf{I}_{\text{d-UV}}\},\tag{S82}$$

where the set $\mathbf{I}_{\text{d-UV}}$ contains indices (i, j, l) corresponding to the independent parameters $(p_{i,j,l}, q_{i,j,l}) \in \mathbf{P}_{\text{d-UV}}$. Then, we define a function $\mathbf{g}_{\text{d-UV}}$ to express the map from $\mathbf{P}_{\text{d-UV}}$ to $\mathbf{X}_{\text{d-UV}}$ as

$$\mathbf{x}_d = \mathbf{g}_{\text{d-UV}}(\mathbf{p}_d; \mathbf{a}_{\text{d-UV}}); \mathbf{x}_d \in \mathbf{X}_{\text{d-UV}}, \mathbf{p}_d \in \mathbf{P}_{\text{d-UV}}, \quad (\text{S83})$$

where $\mathbf{a}_{\text{d-UV}} = (R, r)$ is the array that contains the major radius R and minor radius r of the torus with the deployed UV mesh. Inversely, with Equation (S29), we can define a function $\mathbf{f}_{\text{d-UV}}$ to express the map from $\mathbf{X}_{\text{d-UV}}$ to $\mathbf{P}_{\text{d-UV}}$ as

$$\mathbf{p}_d = \mathbf{f}_{\text{d-UV}}(\mathbf{x}_d; \mathbf{a}_{\text{d-UV}}); \mathbf{p}_d \in \mathbf{P}_{\text{d-UV}}, \mathbf{x}_d \in \mathbf{X}_{\text{d-UV}}. \quad (\text{S84})$$

Constraints for compatibility. To quantify the compatibility between the compact and deployed meshes, we define the following metric function

$$f_{\text{metric}}(\mathbf{X}_c, \mathbf{X}_d, \mathbf{c}) = \frac{1}{N_e} \sum_{k=1}^{N_e} (s_{c,k} - s_{d,k})^2, \quad (\text{S85})$$

where $s_{c,k}$ is the k -th side length (including edges and diagonal crease) of a compact RS or UV mesh; $s_{d,k}$ is the corresponding side length for the deployed mesh; \mathbf{X}_c is the set of vertex positions of the compact mesh, defined by Equation (S53) for an RS mesh or by Equation (S72) for a UV mesh; \mathbf{X}_d is the set of vertex positions of the deployed mesh, defined by Equation (S63) for an RS mesh or by Equation (S82) for a UV mesh; \mathbf{c} is the array of crease assignment defined by Equation (S33) for an RS mesh or by Equation (S39) for a UV mesh. The metric function f_{metric} can be equivalently expressed by counting the panels:

$$f_{\text{metric}}(\mathbf{X}_c, \mathbf{X}_d, \mathbf{c}) = \frac{1}{5N_p} \sum_{n=1}^{N_p} \sum_{m=1}^5 (s_{c,m,n} - s_{d,m,n})^2, \quad (\text{S86})$$

where $s_{c,m,n}$ is the m -th side (and crease) length of the n -th panel of a compact RS or UV mesh, and $s_{d,m,n}$ is the corresponding side length for the deployed mesh. The summation goes through all the N_p panels. For an $N \times N \times 6$ RS mesh, we have $N_p = 6N^2$. For an $M \times N$ UV mesh, we have $N_p = MN$. The panels of an RS or UV mesh can have different crease assignment (two diagonals) and different shapes (quadrilaterals or triangles), as shown in Figure S7. The lengths $s_{c,m,n}$ or $s_{d,m,n}$ correspond to the four edges ($m = 1, 2, 3, 4$) and the assigned crease ($m = 5$). It is worth noting that the triangular panels are considered to have an edge of zero length and the crease of zero length.

The constraint function of the optimization framework should receive the parameters \mathbf{P}_c and \mathbf{P}_d and the size variables a_c and a_d as input instead of the vertex positions \mathbf{X}_c and \mathbf{X}_d . We note that the size variable a_d can be a scalar $a_{d\text{-RS}}$ for the deployed cubic or spherical RS meshes or a vector $\mathbf{a}_{d\text{-UV}}$ for the deployed toric UV mesh. For conciseness, we use the scalar notation a_d to refer to both $a_{d\text{-RS}}$ and $\mathbf{a}_{d\text{-UV}}$. Then, we define the compatibility constraint function as

$$f_{\text{comp.}}(\mathbf{P}_c, \mathbf{P}_d, a_c, a_d, \mathbf{c}) = \frac{f_{\text{metric}}[\mathbf{g}_c(\mathbf{P}_c; a_c), \mathbf{g}_d(\mathbf{P}_d; a_d)]}{\ell_d^2}, \quad (\text{S87})$$

where \mathbf{P}_c is the set of independent parameters of the compact mesh, defined by Equation (S50) for an RS mesh or by Equation (S69) for a UV mesh; \mathbf{P}_d is the set of independent parameters of the deployed mesh, defined by Equation (S60) for an RS mesh or by Equation (S79) for a UV mesh; a_c is the size of the compact mesh, equal to the half side length of a cube or the radius of a sphere; a_d is either the size of the deployed mesh, equal to the half side length of a cube L , the radius of a sphere R , or the array of the major radius and minor radius of a torus (R, r) ; \mathbf{c} is the array of crease assignment defined by Equation (S33) for an RS mesh or by Equation (S39) for a UV mesh; \mathbf{g}_c represents the map from the 2D parameters to the spatial vertex positions given by Equation (S54) for a compact RS mesh or by Equation (S73) for a compact UV mesh; \mathbf{g}_d represents the map from the 2D parameters to the spatial vertex positions given by Equation (S64) for a deployed RS mesh or by Equation (S83) for a deploy UV mesh. We note that the constraint function is nondimensionalized by a characteristic length of the deployed mesh ℓ_d . For the cubic RS mesh, we define $\ell_d = 2a_{d\text{-RS}}/N$; for the spherical RS mesh, we define $\ell_d = 2\pi a_{d\text{-RS}}/N$; for the toric UV mesh, we define $\ell_d = [4\pi^2 Rr/(MN)]^{1/2}$, where R and r are components of $\mathbf{a}_{d\text{-UV}} = (R, r)$.

Constraints for relative positions of vertices. To avoid the distortion of the meshes, we should impose constraints to relative positions of vertices in panels and slits. However, the panels and slits are mainly skew quadrilaterals, which means their four vertices are not necessarily on the same plane. It is complicated to characterize the “distortion” of the skew quadrilaterals directly in the 3D space. We take advantage of the parametrization of the 3D coordinates and constrain the vertex positions with their 2D coordinates in the parameter space. The counterpart of a skew quadrilateral in the 3D space is a planar quadrilateral in the 2D space. As shown in Figure S9, during the optimization, an initially-valid planar quadrilateral may evolve to a distorted one that is non-convex, self-intersected, or orientation-reversing. We aim to avoid these circumstances by

imposing the following inequality constraints

$$\begin{aligned} \mathbf{n} \cdot [(\mathbf{p}_3 - \mathbf{p}_1) \times (\mathbf{p}_2 - \mathbf{p}_1)] &\leq 0, & \mathbf{n} \cdot [(\mathbf{p}_4 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)] &\leq 0, \\ \mathbf{n} \cdot [(\mathbf{p}_4 - \mathbf{p}_2) \times (\mathbf{p}_3 - \mathbf{p}_2)] &\leq 0, & \mathbf{n} \cdot [(\mathbf{p}_1 - \mathbf{p}_2) \times (\mathbf{p}_4 - \mathbf{p}_2)] &\leq 0, \end{aligned} \quad (\text{S88})$$

where \mathbf{n} is the normal vector pointing towards the outside direction of the image. Essentially, we have defined the “distortion” of a skew quadrilateral in the 3D space as the non-convexity, self-intersection, or orientation reversing of its counterpart in the parameter space. In addition, Equation (S88) can also avoid the orientation reversing of triangles if we consider the triangles as degenerate quadrilaterals with two coinciding vertices. Therefore, Equation (S88) is a unified constraint applicable to all the panels and slits of the RS or UV meshes. For both the compact and deployed meshes, we work through all the panels and slits and collect the inequalities given by Equation (S88). We denote all the inequalities by the following vectorized constraint

$$\mathbf{f}_{\text{pos.}}(\mathbf{P}_c, \mathbf{P}_d) \leq 0, \quad (\text{S89})$$

where \mathbf{P}_c is the set of independent parameters of compact meshes, defined by Equation (S50) for an RS mesh and Equation (S69) for a UV mesh; \mathbf{P}_d is the set of independent parameters of deployed meshes, defined by Equation (S60) for an RS mesh and Equation (S79) for a UV mesh.

Constraints for continuity. For compact RS meshes, we recall that there are eight discontinuous intersections (edges 9 to 16 in Figure S2C) at which the panels on different sides are not connected with each other. This discontinuity may lead to additional slits at the boundaries, which are not supposed to exist on a theoretically compact structure, as shown in Figure S10. To tackle this issue, we align the corresponding vertices on the discontinuous intersections to remove the redundant slits. These constraints can be easily expressed in the parameter space by

$$\begin{aligned} p_{i,1,k,1} + q_{N,N-i+2,k',2} &= 1, & i &= 2, 3, \dots, N; & k &= 2, k' = 5; \\ p_{i,1,k,1} - q_{1,i-1,k',4} &= 0, & i &= 2, 3, \dots, N; & k &= 4, k' = 5; \\ p_{i,N,k,3} - q_{N,i+1,k',2} &= 0, & i &= 1, 2, \dots, N-1; & k &= 2, k' = 6; \\ p_{i,N,k,3} + q_{1,N-i,k',4} &= 1, & i &= 1, 2, \dots, N-1; & k &= 4, k' = 6. \end{aligned} \quad (\text{S90})$$

We note that we cannot impose the continuity constraints to the compact and deployed states of an RS mesh at the same time, because this causes over constraints that impede the convergence of the

optimization algorithm. The compact UV meshes have no discontinuous intersections. Therefore, the compactness of a UV mesh can be guaranteed without additional constraints. Instead, we impose continuity constraints at the discontinuous intersection of a deployed UV mesh by

$$p_{i,1,1} - p_{i,N,3} = 0, \quad i = 1, 3, \dots, M - 1. \quad (\text{S91})$$

Altogether, we have developed the continuity constraints for compact RS meshes or deployed UV meshes. For conciseness, we define a unified expression to refer to Equations (S90) and (S91) by

$$\mathbf{f}_{\text{cont.}}(\mathbf{P}_c, \mathbf{P}_d) = 0, \quad (\text{S92})$$

where \mathbf{P}_c is the set of independent parameters of the compact meshes, defined by Equation (S50) for an RS mesh or by Equation (S69) for a UV mesh; \mathbf{P}_d is the set of independent parameters of the deployed meshes, defined by Equation (S60) for an RS mesh or by Equation (S79) for a UV mesh.

Constraints for symmetry. For the sake of the regularity of the optimized meshes, we can additionally impose constraints of symmetry to the vertex positions. Specifically, the RS meshes are constrained to have three planes of mirror symmetry (the xy , yz , and zx planes), and the UV meshes are constrained to have one plane of mirror symmetry (the xy plane) as well as an axis of rotational symmetry (the z axis). The constraints of mirror symmetry for the compact and deployed RS meshes can be expressed in the parameter space as

mirror symmetry with respect to yz plane:

$$\begin{aligned} p_{i,j,k,1} + p_{N+1-i,j,k,2} &= 1, & q_{i,j,k,1} - q_{N+1-i,j,k,2} &= 0, & k &= 1, 3, 5, 6; \\ p_{i,j,k,2} + p_{N+1-i,j,k,1} &= 1, & q_{i,j,k,2} - q_{N+1-i,j,k,1} &= 0, & k &= 1, 3, 5, 6; \\ p_{i,j,k,3} + p_{N+1-i,j,k,4} &= 1, & q_{i,j,k,3} - q_{N+1-i,j,k,4} &= 0, & k &= 1, 3, 5, 6; \\ p_{i,j,k,4} + p_{N+1-i,j,k,3} &= 1, & q_{i,j,k,4} - q_{N+1-i,j,k,3} &= 0, & k &= 1, 3, 5, 6; \\ p_{i,j,k,1} + p_{N+1-i,j,k',2} &= 1, & q_{i,j,k,1} - q_{N+1-i,j,k',2} &= 0, & k &= 2, \quad k' = 4; \\ p_{i,j,k,2} + p_{N+1-i,j,k',1} &= 1, & q_{i,j,k,2} - q_{N+1-i,j,k',1} &= 0, & k &= 2, \quad k' = 4; \\ p_{i,j,k,3} + p_{N+1-i,j,k',4} &= 1, & q_{i,j,k,3} - q_{N+1-i,j,k',4} &= 0, & k &= 2, \quad k' = 4; \\ p_{i,j,k,4} + p_{N+1-i,j,k',3} &= 1, & q_{i,j,k,4} - q_{N+1-i,j,k',3} &= 0, & k &= 2, \quad k' = 4; \end{aligned}$$

mirror symmetry with respect to zx plane:

$$\begin{aligned}
p_{i,j,k,1} + p_{N+1-i,j,k,2} &= 1, & q_{i,j,k,1} - q_{N+1-i,j,k,2} &= 0, & k &= 2, 4; \\
p_{i,j,k,2} + p_{N+1-i,j,k,1} &= 1, & q_{i,j,k,2} - q_{N+1-i,j,k,1} &= 0, & k &= 2, 4; \\
p_{i,j,k,3} + p_{N+1-i,j,k,4} &= 1, & q_{i,j,k,3} - q_{N+1-i,j,k,4} &= 0, & k &= 2, 4; \\
p_{i,j,k,4} + p_{N+1-i,j,k,3} &= 1, & q_{i,j,k,4} - q_{N+1-i,j,k,3} &= 0, & k &= 2, 4; \\
p_{i,j,k,1} - p_{i,N+1-j,k,4} &= 0, & q_{i,j,k,1} + q_{i,N+1-j,k,4} &= 1, & k &= 5, 6; \\
p_{i,j,k,2} - p_{i,N+1-j,k,3} &= 0, & q_{i,j,k,2} + q_{i,N+1-j,k,3} &= 1, & k &= 5, 6; \\
p_{i,j,k,3} - p_{i,N+1-j,k,2} &= 0, & q_{i,j,k,3} + q_{i,N+1-j,k,2} &= 1, & k &= 5, 6; \\
p_{i,j,k,4} - p_{i,N+1-j,k,1} &= 0, & q_{i,j,k,4} + q_{i,N+1-j,k,1} &= 1, & k &= 5, 6; \\
p_{i,j,k,1} + p_{N+1-i,j,k',2} &= 1, & q_{i,j,k,1} - q_{N+1-i,j,k',2} &= 0, & k &= 1, k' = 3; \\
p_{i,j,k,2} + p_{N+1-i,j,k',1} &= 1, & q_{i,j,k,2} - q_{N+1-i,j,k',1} &= 0, & k &= 1, k' = 3; \\
p_{i,j,k,3} + p_{N+1-i,j,k',4} &= 1, & q_{i,j,k,3} - q_{N+1-i,j,k',4} &= 0, & k &= 1, k' = 3; \\
p_{i,j,k,4} + p_{N+1-i,j,k',3} &= 1, & q_{i,j,k,4} - q_{N+1-i,j,k',3} &= 0, & k &= 1, k' = 3;
\end{aligned}$$

mirror symmetry with respect to xy plane:

$$\begin{aligned}
p_{i,j,k,1} - p_{i,N+1-j,k,4} &= 0, & q_{i,j,k,1} + q_{i,N+1-j,k,4} &= 1, & k &= 1, 2, 3, 4; \\
p_{i,j,k,2} - p_{i,N+1-j,k,3} &= 0, & q_{i,j,k,2} + q_{i,N+1-j,k,3} &= 1, & k &= 1, 2, 3, 4; \\
p_{i,j,k,3} - p_{i,N+1-j,k,2} &= 0, & q_{i,j,k,3} + q_{i,N+1-j,k,2} &= 1, & k &= 1, 2, 3, 4; \\
p_{i,j,k,4} - p_{i,N+1-j,k,1} &= 0, & q_{i,j,k,4} + q_{i,N+1-j,k,1} &= 1, & k &= 1, 2, 3, 4; \\
p_{i,j,k,1} - p_{i,N+1-j,k',4} &= 0, & q_{i,j,k,1} + q_{i,N+1-j,k',4} &= 1, & k &= 5, k' = 6; \\
p_{i,j,k,2} - p_{i,N+1-j,k',3} &= 0, & q_{i,j,k,2} + q_{i,N+1-j,k',3} &= 1, & k &= 5, k' = 6; \\
p_{i,j,k,3} - p_{i,N+1-j,k',2} &= 0, & q_{i,j,k,3} + q_{i,N+1-j,k',2} &= 1, & k &= 5, k' = 6; \\
p_{i,j,k,4} - p_{i,N+1-j,k',1} &= 0, & q_{i,j,k,4} + q_{i,N+1-j,k',1} &= 1, & k &= 5, k' = 6;
\end{aligned} \tag{S93}$$

where $i, j = 1, 2, \dots, N$. The constraints of mirror symmetry for the compact and deployed UV

meshes can be expressed in the parameter space as

mirror symmetry with respect to xy plane:

$$\begin{aligned}
p_{i,j,1} - p_{i,N+1-j,4} &= 0, & q_{i,j,1} + q_{i,N+1-j,4} &= 1; \\
p_{i,j,2} - p_{i,N+1-j,3} &= 0, & q_{i,j,2} + q_{i,N+1-j,3} &= 1; \\
p_{i,j,3} - p_{i,N+1-j,2} &= 0, & q_{i,j,3} + q_{i,N+1-j,2} &= 1; \\
p_{i,j,4} - p_{i,N+1-j,1} &= 0, & q_{i,j,4} + q_{i,N+1-j,1} &= 1;
\end{aligned} \tag{S94}$$

where $i = 1, 2, \dots, M$ and $j = 1, 2, \dots, N/2$. The constraints of rotational symmetry for the compact and deployed UV meshes can be expressed in the parameter space as

$$\begin{aligned}
p_{i+2,j,l} - p_{i,j,l} &= 2/M - 1, & q_{i,j,l} - q_{i,j,l} &= 0; & i &= M - 2; \\
p_{i+2,j,l} - p_{i,j,l} &= 2/M, & q_{i,j,l} - q_{i,j,l} &= 0; & \text{otherwise;}
\end{aligned} \tag{S95}$$

where $i = 1, 2, \dots, M - 2$ and $j = 1, 2, \dots, N$. For conciseness, we define a unified expression to refer to Equation (S93) for RS meshes, or Equations (S94) and (S95) for UV meshes, by

$$\mathbf{f}_{\text{sym.}}(\mathbf{P}_c, \mathbf{P}_d) = 0, \tag{S96}$$

where \mathbf{P}_c is the set of independent parameters of the compact meshes, defined by Equation (S50) for an RS mesh or by Equation (S69) for a UV mesh; \mathbf{P}_d is the set of independent parameters of the deployed meshes, defined by Equation (S60) for an RS mesh or by Equation (S79) for a UV mesh. We note that the constraints of symmetry are not necessary to guarantee the convergence of the optimization algorithm.

Objective function. We define the objective function $f_{\text{obj.}}$ to specify opening angles of some slits at the deployed compatible state so that, to a certain extent, we can control the degree of deployment of the optimized RS or UV mesh. We note that the selection of objective function is based on the specific demands in addition to the compatibility. One can define other objective functions, such as the variance of the panel areas to pursue a higher uniformity of the meshes.

For the purpose of specifying certain opening angles of a deployed mesh, first, we define the following angle function

$$f_{\text{angle}}(\mathbf{X}_d) = \frac{1}{N_a} \sum_{n=1}^{N_a} (\cos \omega_{d,n} - \cos \bar{\omega}_{d,n})^2, \tag{S97}$$

where $\omega_{d,n}$ is the n -th opening angle that we aim to specify on a deployed RS or UV mesh, and $\bar{\omega}_{d,n}$ is the specified value of the corresponding angle; \mathbf{X}_d is the set of vertex positions of the deployed meshes, defined by Equation (S63) for an RS mesh or by Equation (S82) for a UV mesh. The number of opening angles to be controlled is denoted by N_a . For RS meshes, we set $N_a = 6$, corresponding to the opening angles of the central slits of the six faces as shown in Figures S8A and S8B. These opening angles $\omega_{d,n}$ can be given by

$$\omega_{d\text{-RS},n} = \arccos \left[\frac{\left(\mathbf{x}_{\frac{N}{2}+1, \frac{N}{2}, n, 3} - \mathbf{x}_{\frac{N}{2}, \frac{N}{2}, n, 3} \right) \cdot \left(\mathbf{x}_{\frac{N}{2}, \frac{N}{2}, n, 4} - \mathbf{x}_{\frac{N}{2}, \frac{N}{2}, n, 3} \right)}{\left\| \mathbf{x}_{\frac{N}{2}+1, \frac{N}{2}, n, 3} - \mathbf{x}_{\frac{N}{2}, \frac{N}{2}, n, 3} \right\| \left\| \mathbf{x}_{\frac{N}{2}, \frac{N}{2}, n, 4} - \mathbf{x}_{\frac{N}{2}, \frac{N}{2}, n, 3} \right\|} \right], \quad (\text{S98})$$

where $n = 1, 2, \dots, 6$, and the vertex coordinates $\mathbf{x}_{i,j,k,l}$ belong to the set $\mathbf{X}_{d\text{-RS}}$ defined by Equation (S63) for the RS meshes. For UV meshes, we set $N_a = 4$, corresponding to the opening angles of four slits at the outer circumference of the torus as shown in Figure S8C. In this case, the opening angles $\omega_{d,n}$ are given by

$$\begin{aligned} \omega_{d\text{-UV},1} &= \arccos \left[\frac{\left(\mathbf{x}_{1, \frac{N}{2}, 3} - \mathbf{x}_{1, \frac{N}{2}, 4} \right) \cdot \left(\mathbf{x}_{M, \frac{N}{2}, 4} - \mathbf{x}_{1, \frac{N}{2}, 4} \right)}{\left\| \mathbf{x}_{1, \frac{N}{2}, 3} - \mathbf{x}_{1, \frac{N}{2}, 4} \right\| \left\| \mathbf{x}_{M, \frac{N}{2}, 4} - \mathbf{x}_{1, \frac{N}{2}, 4} \right\|} \right], \\ \omega_{d\text{-UV},2} &= \arccos \left[\frac{\left(\mathbf{x}_{\frac{M}{4}+1, \frac{N}{2}, 3} - \mathbf{x}_{\frac{M}{4}+1, \frac{N}{2}, 4} \right) \cdot \left(\mathbf{x}_{\frac{M}{4}, \frac{N}{2}, 4} - \mathbf{x}_{\frac{M}{4}+1, \frac{N}{2}, 4} \right)}{\left\| \mathbf{x}_{\frac{M}{4}+1, \frac{N}{2}, 3} - \mathbf{x}_{\frac{M}{4}+1, \frac{N}{2}, 4} \right\| \left\| \mathbf{x}_{\frac{M}{4}, \frac{N}{2}, 4} - \mathbf{x}_{\frac{M}{4}+1, \frac{N}{2}, 4} \right\|} \right], \\ \omega_{d\text{-UV},3} &= \arccos \left[\frac{\left(\mathbf{x}_{\frac{M}{2}+1, \frac{N}{2}, 3} - \mathbf{x}_{\frac{M}{2}+1, \frac{N}{2}, 4} \right) \cdot \left(\mathbf{x}_{\frac{M}{2}, \frac{N}{2}, 4} - \mathbf{x}_{\frac{M}{2}+1, \frac{N}{2}, 4} \right)}{\left\| \mathbf{x}_{\frac{M}{2}+1, \frac{N}{2}, 3} - \mathbf{x}_{\frac{M}{2}+1, \frac{N}{2}, 4} \right\| \left\| \mathbf{x}_{\frac{M}{2}, \frac{N}{2}, 4} - \mathbf{x}_{\frac{M}{2}+1, \frac{N}{2}, 4} \right\|} \right], \\ \omega_{d\text{-UV},4} &= \arccos \left[\frac{\left(\mathbf{x}_{\frac{3M}{4}+1, \frac{N}{2}, 3} - \mathbf{x}_{\frac{3M}{4}+1, \frac{N}{2}, 4} \right) \cdot \left(\mathbf{x}_{\frac{3M}{4}, \frac{N}{2}, 4} - \mathbf{x}_{\frac{3M}{4}+1, \frac{N}{2}, 4} \right)}{\left\| \mathbf{x}_{\frac{3M}{4}+1, \frac{N}{2}, 3} - \mathbf{x}_{\frac{3M}{4}+1, \frac{N}{2}, 4} \right\| \left\| \mathbf{x}_{\frac{3M}{4}, \frac{N}{2}, 4} - \mathbf{x}_{\frac{3M}{4}+1, \frac{N}{2}, 4} \right\|} \right], \end{aligned} \quad (\text{S99})$$

where the vertex coordinates $\mathbf{x}_{i,j,l}$ belong to the set $\mathbf{X}_{d\text{-UV}}$ defined by Equation (S82) for the UV meshes.

Similar to the constraint functions, the objective function should receive the parameters \mathbf{P}_d and the size variable a_d as input variables instead of the vertex positions \mathbf{X}_d . Therefore, we define the objective function as

$$f_{\text{obj}}(\mathbf{P}_d, a_d) = f_{\text{angle}}[\mathbf{g}_d(\mathbf{P}_d; a_d)], \quad (\text{S100})$$

where \mathbf{P}_d is the set of independent parameters of the deployed meshes, defined by Equation (S60) for an RS mesh or by Equation (S79) for a UV mesh.; a_d is the size variable of the deployed mesh, equal to the half side length of a cube L or the radius of a sphere R , or representing the array of the major radius and minor radius of a torus (R, r) ; \mathbf{g}_d represents the map from the 2D parameters to the spatial vertex positions given by Equation (S64) for deployed RS meshes or by Equation (S83) for deployed UV meshes.

Implementation. The optimization framework for the compatibility between the compact and deployed configurations of an RS or UV mesh can be described in a unified form as

$$\underset{\mathbf{P}_c, \mathbf{P}_d, a_c}{\text{minimize } f_{\text{obj.}}(\mathbf{P}_d, a_d)} \text{ subject to } \begin{cases} f_{\text{comp.}}(\mathbf{P}_c, \mathbf{P}_d, a_c, a_d, \mathbf{c}) = 0, \\ \mathbf{f}_{\text{sym.}}(\mathbf{P}_c, \mathbf{P}_d) = 0, \\ \mathbf{f}_{\text{pos.}}(\mathbf{P}_c, \mathbf{P}_d) \leq 0, \\ \mathbf{f}_{\text{cont.}}(\mathbf{P}_c, \mathbf{P}_d) \leq 0. \end{cases} \quad (\text{S101})$$

We use the sequential quadratic programming (SQP) algorithm to solve the optimization problem given by Equation (S101). In our calculation, the crease assignment is prescribed to have the same types of symmetry. The RS meshes are mirror symmetric with respect to three orthogonal planes, with the creases assigned as

$$c_{i,j,k} = \begin{cases} +1, & |i - j| < N/2, \\ -1, & |i - j| \geq N/2. \end{cases} \quad (\text{S102})$$

The UV meshes are mirror symmetric with respect to the horizontal plane and rotational symmetric with respect to the vertical axis, with the creases assigned as

$$c_{i,j} = \begin{cases} +1, & \text{even } i + j, \\ -1, & \text{odd } i + j. \end{cases} \quad (\text{S103})$$

The parameters of regular patterns (text S3) serve as the initial values of the optimization variables \mathbf{P}_c and \mathbf{P}_d . To generate initial compact meshes, the initial values of \mathbf{P}_c are given by Equation (S31) for an RS mesh or by Equation (S37) for a UV mesh. To generate initial deployed meshes, we substitute the parameter $\xi = 0.5\pi$ to Equation (S36) for an RS mesh or to Equation (S43) for a UV mesh. We fix the size of the deployed meshes with $a_d = 1$ for an RS mesh or $\mathbf{a}_d = (1, 0.5)$ for a UV mesh. The initial size of the compact meshes is specified as $a_c = 1/\sqrt{2}$ for an RS mesh or $a_c = 1$

for a UV mesh. In the objective function, the reference opening angles are specified to be a constant $\bar{\omega}_{\text{d-RS},n} = \omega = 0.5\pi$ ($n = 1, 2, \dots, 6$) for an RS mesh or a constant $\bar{\omega}_{\text{d-UV},n} = \omega = 0.4\pi$ ($n = 1, 2, 3, 4$) for a UV mesh.

S5 Analysis on kinematic indeterminacy

Rotation axis of hinges. A hinge should guide two panels to rotate around the hinge axis. The axis is fixed with respect to the frame of reference of each panel. In other words, the hinges are attached to the panels. We recall that each quadrilateral panel is divided into two triangles according to the crease assignment ($c = +1$ or -1). We suppose the panel vertices are indexed counterclockwise by 1, 2, 3, 4. For $c = +1$, the two triangles connect vertices 1, 2, 3 and 1, 3, 4 respectively. For $c = -1$, the two triangles connect vertices 1, 2, 4 and 2, 3, 4 respectively. Besides, the panels are distinguished by their clockwise (even $i + j$) or counterclockwise rotation (odd $i + j$) upon deployment. On these grounds, we summarize the attachment patterns in Figure S12A. We can see that each triangle has two attached hinges and each quadrilateral panel has four hinges in total. The different rotation directions and different crease assignments lead to different attachment patterns involving the hinges and the panels. The attachment rules above are applicable to the quadrilateral panels for both the RS and the UV meshes. In addition, we recall that the UV meshes have some panels that are originally triangles ($j = 1$ or N). These triangular panels can also be classified into two types according to their clockwise (even $i + j$ and $j = 1, N$) or counterclockwise (odd $i + j$ and $j = 1, N$) rotation upon deployment. In analogy to the quadrilateral panels, we give the attachment rules on these triangular panels, as shown in Figure S12B. We can see that there are three hinges in total for an original triangular panel. Altogether, in the panel-hinge system, each hinge connects two (either divided or original) triangles at the common vertex. The deployment of an ori-kiri assemblage requires the panels to rotate around the hinge axis from the compact state to the compatible deployed state. We can calculate the direction of the rotation axis for each hinge based on the compact and deployed configurations.

As shown in Figure S12C, at the compact state, two triangular panels have a common vertex **a** and each has two more vertices **b**, **c**, and **d**, **e**, respectively. From the compact state to the compatible deployed state, the triangle **abc** undergoes a rigid-body motion with its vertices moving to $\tilde{\mathbf{a}}$, $\tilde{\mathbf{b}}$, $\tilde{\mathbf{c}}$. The triangle **ade** follows the same rigid-body motion and additionally rotates around

the common vertex \mathbf{a} with respect to the triangle \mathbf{abc} , with the vertices \mathbf{a} , \mathbf{d} , \mathbf{e} moving to $\tilde{\mathbf{a}}$, $\tilde{\mathbf{d}}$, $\tilde{\mathbf{e}}$, respectively. The rotation matrix can be determined by the vertex locations of the compact and deployed configurations. First, we calculate the basis vectors of the local coordinate systems fixed on the panels of the compact and deployed configurations. On the triangle \mathbf{abc} , we set up a local coordinate system with basis vectors \mathbf{m}_1 , \mathbf{m}_2 , \mathbf{m}_3 as

$$\mathbf{m}_1 = \frac{\mathbf{c} - \mathbf{a}}{\|\mathbf{c} - \mathbf{a}\|}, \quad \mathbf{m}_3 = \frac{(\mathbf{b} - \mathbf{a}) \times \mathbf{m}_1}{\|(\mathbf{b} - \mathbf{a}) \times \mathbf{m}_1\|}, \quad \mathbf{m}_2 = \frac{\mathbf{m}_3 \times \mathbf{m}_1}{\|\mathbf{m}_3 \times \mathbf{m}_1\|}. \quad (\text{S104})$$

On the triangle \mathbf{ade} , we set up a local coordinate system with basis vectors \mathbf{n}_1 , \mathbf{n}_2 , \mathbf{n}_3 as

$$\mathbf{n}_1 = \frac{\mathbf{d} - \mathbf{a}}{\|\mathbf{d} - \mathbf{a}\|}, \quad \mathbf{n}_3 = \frac{\mathbf{n}_1 \times (\mathbf{e} - \mathbf{a})}{\|\mathbf{n}_1 \times (\mathbf{e} - \mathbf{a})\|}, \quad \mathbf{n}_2 = \frac{\mathbf{n}_3 \times \mathbf{n}_1}{\|\mathbf{n}_3 \times \mathbf{n}_1\|}. \quad (\text{S105})$$

On the deformed triangle $\tilde{\mathbf{abc}}$, we set up a local coordinate system with basis vectors $\tilde{\mathbf{m}}_1$, $\tilde{\mathbf{m}}_2$, $\tilde{\mathbf{m}}_3$ as

$$\tilde{\mathbf{m}}_1 = \frac{\tilde{\mathbf{c}} - \tilde{\mathbf{a}}}{\|\tilde{\mathbf{c}} - \tilde{\mathbf{a}}\|}, \quad \tilde{\mathbf{m}}_3 = \frac{(\tilde{\mathbf{b}} - \tilde{\mathbf{a}}) \times \tilde{\mathbf{m}}_1}{\|(\tilde{\mathbf{b}} - \tilde{\mathbf{a}}) \times \tilde{\mathbf{m}}_1\|}, \quad \tilde{\mathbf{m}}_2 = \frac{\tilde{\mathbf{m}}_3 \times \tilde{\mathbf{m}}_1}{\|\tilde{\mathbf{m}}_3 \times \tilde{\mathbf{m}}_1\|}. \quad (\text{S106})$$

On the deformed triangle $\tilde{\mathbf{ade}}$, we set up a local coordinate system with basis vectors $\tilde{\mathbf{n}}_1$, $\tilde{\mathbf{n}}_2$, $\tilde{\mathbf{n}}_3$ as

$$\tilde{\mathbf{n}}_1 = \frac{\tilde{\mathbf{d}} - \tilde{\mathbf{a}}}{\|\tilde{\mathbf{d}} - \tilde{\mathbf{a}}\|}, \quad \tilde{\mathbf{n}}_3 = \frac{\tilde{\mathbf{n}}_1 \times (\tilde{\mathbf{e}} - \tilde{\mathbf{a}})}{\|\tilde{\mathbf{n}}_1 \times (\tilde{\mathbf{e}} - \tilde{\mathbf{a}})\|}, \quad \tilde{\mathbf{n}}_2 = \frac{\tilde{\mathbf{n}}_3 \times \tilde{\mathbf{n}}_1}{\|\tilde{\mathbf{n}}_3 \times \tilde{\mathbf{n}}_1\|}. \quad (\text{S107})$$

Then, we calculate the rotation matrix $\tilde{\mathbf{R}}$ under the basis vectors \mathbf{m}_1 , \mathbf{m}_2 , \mathbf{m}_3 as

$$\tilde{\mathbf{R}} = \begin{bmatrix} \tilde{\mathbf{m}}_1 \cdot \tilde{\mathbf{n}}_1 & \tilde{\mathbf{m}}_1 \cdot \tilde{\mathbf{n}}_2 & \tilde{\mathbf{m}}_1 \cdot \tilde{\mathbf{n}}_3 \\ \tilde{\mathbf{m}}_2 \cdot \tilde{\mathbf{n}}_1 & \tilde{\mathbf{m}}_2 \cdot \tilde{\mathbf{n}}_2 & \tilde{\mathbf{m}}_2 \cdot \tilde{\mathbf{n}}_3 \\ \tilde{\mathbf{m}}_3 \cdot \tilde{\mathbf{n}}_1 & \tilde{\mathbf{m}}_3 \cdot \tilde{\mathbf{n}}_2 & \tilde{\mathbf{m}}_3 \cdot \tilde{\mathbf{n}}_3 \end{bmatrix} \begin{bmatrix} \mathbf{m}_1 \cdot \mathbf{n}_1 & \mathbf{m}_1 \cdot \mathbf{n}_2 & \mathbf{m}_1 \cdot \mathbf{n}_3 \\ \mathbf{m}_2 \cdot \mathbf{n}_1 & \mathbf{m}_2 \cdot \mathbf{n}_2 & \mathbf{m}_2 \cdot \mathbf{n}_3 \\ \mathbf{m}_3 \cdot \mathbf{n}_1 & \mathbf{m}_3 \cdot \mathbf{n}_2 & \mathbf{m}_3 \cdot \mathbf{n}_3 \end{bmatrix}^{-1}. \quad (\text{S108})$$

The matrix $\tilde{\mathbf{R}}$ transforms the triangle \mathbf{ade} to $\tilde{\mathbf{ade}}$ with respect to the material coordinate frame fixed on the triangle \mathbf{abc} . On the undeformed (compact) configuration, the material coordinate frame has the basis vectors \mathbf{m}_1 , \mathbf{m}_2 , \mathbf{m}_3 . On the deformed (deployed) configuration, the material coordinate frame has the basis vectors $\tilde{\mathbf{m}}_1$, $\tilde{\mathbf{m}}_2$, $\tilde{\mathbf{m}}_3$. Since the compact and the deployed configurations have been optimized to be compatible, the triangle \mathbf{abc} only undergoes rigid-body motion to $\tilde{\mathbf{abc}}$, so does the triangle \mathbf{ade} to $\tilde{\mathbf{ade}}$. As a result, the matrix $\tilde{\mathbf{R}}$ performs the relative rotation of the triangle $\tilde{\mathbf{ade}}$ with respect to the triangle $\tilde{\mathbf{abc}}$. Furthermore, the relative rotation in the matrix form $\tilde{\mathbf{R}}$ can also be characterized by the rotation around a rotation axis $\tilde{\mathbf{n}}$ by a rotation angle $\tilde{\gamma}$. We can calculate $\tilde{\gamma}$ and $\tilde{\mathbf{n}}$ by

$$\tilde{\gamma} = \arccos \left(\frac{\tilde{R}_{11} + \tilde{R}_{22} + \tilde{R}_{33} - 1}{2} \right), \quad (\text{S109})$$

$$\tilde{\mathbf{n}} = \frac{1}{2 \sin \gamma} [(\tilde{R}_{32} - \tilde{R}_{23})\tilde{\mathbf{m}}_1 + (\tilde{R}_{13} - \tilde{R}_{31})\tilde{\mathbf{m}}_2 + (\tilde{R}_{21} - \tilde{R}_{12})\tilde{\mathbf{m}}_3]. \quad (\text{S110})$$

The vector $\tilde{\mathbf{n}}$ is expressed under the local basis vectors $\tilde{\mathbf{m}}_1, \tilde{\mathbf{m}}_2, \tilde{\mathbf{m}}_3$ on the deployed configuration.

Besides, the rotation axis can also be written as

$$\mathbf{n} = \frac{1}{2 \sin \gamma} [(\tilde{R}_{32} - \tilde{R}_{23})\mathbf{m}_1 + (\tilde{R}_{13} - \tilde{R}_{31})\mathbf{m}_2 + (\tilde{R}_{21} - \tilde{R}_{12})\mathbf{m}_3]. \quad (\text{S111})$$

The vector \mathbf{n} is expressed under the local basis vectors $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ on the compact configuration.

The vectors \mathbf{n} and $\tilde{\mathbf{n}}$ represent the rotation axis of the same hinge on the compact and the deployed configurations, respectively. The local expressions benefit the analysis of kinematic indeterminacy below because the hinges are actually attached to the panels when the ori-kiri assemblages transit between the compatible configurations.

Kinematic indeterminacy. To investigate the mobility of the ori-kiri assemblages, we use a surrogate truss model in which the panels are replaced by bars alongside the edges and creases. Figure S11A shows the truss model of a deployed spherical RS mesh. We denote the number of bars by N_{bar} and the number of nodes by N_{node} . For an $N \times N \times 6$ RS mesh (either compact or deployed), we have $N_{\text{bar}} = 30N^2$ and $N_{\text{node}} = 12N^2 + 4N$. For an $M \times N$ UV mesh (either compact or deployed), we have $N_{\text{bar}} = M(5N - 4)$ and $N_{\text{node}} = M(2N - 1)$. We denote the position of a node i by \mathbf{x}_i . The node i has three degrees of freedom, that is, its infinitesimal displacement $d\mathbf{x}_i$. We aim to calculate the degree of kinematic indeterminacy of the bar system under infinitesimal displacements. To this end, we impose the first-order inextensibility constraint to the bars. Consider a bar connecting node i and j as shown in Figure S11B. The length of the bar can be calculated by $l_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$. Then we equate the first-order differential of the bar length $dl_{i,j}$ to zero to obtain the first-order inextensibility constraint

$$(\mathbf{x}_i - \mathbf{x}_j) \cdot (d\mathbf{x}_i - d\mathbf{x}_j) = 0, \quad (\text{S112})$$

This constraint can be applied to all bars and be gathered in the matrix form as

$$\mathbf{B}_{\text{bar}} \cdot d\mathbf{x} = 0, \quad (\text{S113})$$

where \mathbf{B}_{bar} is the $N_{\text{bar}} \times 3N_{\text{node}}$ constraint matrix, and $d\mathbf{x}$ is the vector of infinitesimal displacements for all nodes.

To simulate the axial rotation guided by the hinges, we need to additionally impose constraints to the relative motion of the nodes constituting adjacent triangular panels. We denote the number of

hinges by N_{hinge} . For an $N \times N \times 6$ RS mesh (either compact or deployed), we have $N_{\text{hinge}} = 12N^2 - 4N$. For an $M \times N$ UV mesh (either compact or deployed), we have $N_{\text{hinge}} = M(2N - 1)$. Figure S11C shows the hinge on the node i that connects the triangle ijk and the triangle ilm . On the triangle ijk , we set up a local coordinate system with basis vectors $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ as

$$\mathbf{m}_1 = \frac{\mathbf{x}_k - \mathbf{x}_i}{\|\mathbf{x}_k - \mathbf{x}_i\|}, \quad \mathbf{m}_3 = \frac{(\mathbf{x}_j - \mathbf{x}_i) \times \mathbf{m}_1}{\|(\mathbf{x}_j - \mathbf{x}_i) \times \mathbf{m}_1\|}, \quad \mathbf{m}_2 = \frac{\mathbf{m}_3 \times \mathbf{m}_1}{\|\mathbf{m}_3 \times \mathbf{m}_1\|}. \quad (\text{S114})$$

Under the local basis vectors $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$, we give the direction vector of the rotation axis $\tilde{\mathbf{n}}$ by

$$\tilde{\mathbf{n}} = \tilde{n}_1 \mathbf{m}_1 + \tilde{n}_2 \mathbf{m}_2 + \tilde{n}_3 \mathbf{m}_3, \quad (\text{S115})$$

where the components $\tilde{n}_1, \tilde{n}_2, \tilde{n}_3$ are constants that do not vary with the infinitesimal displacements $d\mathbf{x}_i, d\mathbf{x}_j, d\mathbf{x}_k$. This means that the direction of the rotation axis is fixed on the triangle ijk . In addition, since the hinge guides the axial rotation of both the triangle ijk and the triangle ilm , the direction of the rotation axis should also be fixed on the triangle ilm . To formulate this constraint, we calculate the angles included between the axis $\tilde{\mathbf{n}}$ and two linearly independent vectors on the triangle ilm , say, α and β as shown in Figure S11C. The angle α corresponds to the vector $\tilde{\mathbf{t}}_1 = \mathbf{x}_l - \mathbf{x}_i$, which can be expressed with the local basis vectors $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ by

$$\tilde{\mathbf{t}}_1 = [(\mathbf{x}_l - \mathbf{x}_i) \cdot \mathbf{m}_1] \mathbf{m}_1 + [(\mathbf{x}_l - \mathbf{x}_i) \cdot \mathbf{m}_2] \mathbf{m}_2 + [(\mathbf{x}_l - \mathbf{x}_i) \cdot \mathbf{m}_3] \mathbf{m}_3, \quad (\text{S116})$$

and the angle β corresponds to the vector $\tilde{\mathbf{t}}_2 = \mathbf{x}_m - \mathbf{x}_i$, which can be expressed by the local basis vectors $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ as

$$\tilde{\mathbf{t}}_2 = [(\mathbf{x}_m - \mathbf{x}_i) \cdot \mathbf{m}_1] \mathbf{m}_1 + [(\mathbf{x}_m - \mathbf{x}_i) \cdot \mathbf{m}_2] \mathbf{m}_2 + [(\mathbf{x}_m - \mathbf{x}_i) \cdot \mathbf{m}_3] \mathbf{m}_3. \quad (\text{S117})$$

Then the expressions of α and β can be written as

$$\alpha = \arccos \left(\tilde{\mathbf{n}} \cdot \frac{\tilde{\mathbf{t}}_1}{\|\tilde{\mathbf{t}}_1\|} \right), \quad \beta = \arccos \left(\tilde{\mathbf{n}} \cdot \frac{\tilde{\mathbf{t}}_2}{\|\tilde{\mathbf{t}}_2\|} \right). \quad (\text{S118})$$

Finally, the constraints for the axial rotation can be expressed by the first-order differentials of α and β as

$$\begin{aligned} \frac{\partial \alpha}{\partial \mathbf{x}_i} \cdot d\mathbf{x}_i + \frac{\partial \alpha}{\partial \mathbf{x}_j} \cdot d\mathbf{x}_j + \frac{\partial \alpha}{\partial \mathbf{x}_k} \cdot d\mathbf{x}_k + \frac{\partial \alpha}{\partial \mathbf{x}_l} \cdot d\mathbf{x}_l + \frac{\partial \alpha}{\partial \mathbf{x}_m} \cdot d\mathbf{x}_m &= 0, \\ \frac{\partial \beta}{\partial \mathbf{x}_i} \cdot d\mathbf{x}_i + \frac{\partial \beta}{\partial \mathbf{x}_j} \cdot d\mathbf{x}_j + \frac{\partial \beta}{\partial \mathbf{x}_k} \cdot d\mathbf{x}_k + \frac{\partial \beta}{\partial \mathbf{x}_l} \cdot d\mathbf{x}_l + \frac{\partial \beta}{\partial \mathbf{x}_m} \cdot d\mathbf{x}_m &= 0. \end{aligned} \quad (\text{S119})$$

These constraints can be applied to all the hinges and be gathered in the matrix form as

$$\mathbf{B}_{\text{hinge}} \cdot d\mathbf{x} = 0, \quad (\text{S120})$$

in which $\mathbf{B}_{\text{hinge}}$ is the $2N_{\text{hinge}} \times 3N_{\text{node}}$ constraint matrix, and $d\mathbf{x}$ is the vector of infinitesimal displacements for all nodes.

Now we can calculate the degree of kinematic indeterminacy of the surrogate truss model to quantify the mobility of the ori-kiri assemblages of which the panels are connected by hinges at the common vertices. To this end, we write the kinematic equation of the truss model as

$$\mathbf{B} \cdot d\mathbf{x} = \begin{bmatrix} \mathbf{B}_{\text{bar}} \\ \mathbf{B}_{\text{hinge}} \end{bmatrix} \cdot d\mathbf{x} = 0, \quad (\text{S121})$$

where the kinematic matrix \mathbf{B} consists of both the inextensibility constraints and the axial rotation constraints. Then the degree of kinematic indeterminacy of the truss model (denoted by b) can be calculated by

$$b = 3N_{\text{node}} - \text{rank}(\mathbf{B}) - 6. \quad (\text{S122})$$

Furthermore, to show the necessity of replacing spherical joints by hinges, we calculate the following degree of kinematic indeterminacy

$$b' = 3N_{\text{node}} - \text{rank}(\mathbf{B}') - 6, \quad (\text{S123})$$

where the kinematic matrix \mathbf{B}' consists of all the rows from \mathbf{B}_{bar} and part of the rows from $\mathbf{B}_{\text{hinge}}$. Therefore, b' represents the mobility of the ori-kiri assemblages with mixed ball and revolute joints at the vertices. Specifically, if $\mathbf{B}' = \mathbf{B}_{\text{bar}}$, b' represents the mobility of the ori-kiri assemblages with full spherical joints.

S6 Energy landscapes upon deployment

Energy formulation. We assume that the surrogate truss model has two sources of elastic energy, the stretching energy of the bars and the off-axial-rotation energy at the hinges. As shown in Figure S13A, the stretching energy is induced from the linear deformation of the bars. Supposing that the total number of bars is N_{bar} in the truss model, we can write the stretching energy E_S as

$$E_S(\mathbf{X}) = \frac{1}{2} \sum_{n=1}^{N_{\text{bar}}} [k_{S,n} (s'_n - s_n)^2], \quad (\text{S124})$$

where s_n is the original length of the n -th bar; s'_n is the deformed length of the n -th bar; $k_{S,n}$ is the stiffness of the n -th bar; \mathbf{X} is the set containing all the 3D vertex positions of the deformed truss. The original length s_n is determined by the geometry of a given ori-kiri assemblage. The deformed length s'_n can be calculated with the vertex positions \mathbf{X} .

The off-axial-rotation energy is induced from the restoring force resisting the relative off-axial rotation of adjacent panels. As shown in Figure S13B, we consider two triangles connected by a hinge at their common vertex \mathbf{a} . From the compact state to the compatible deployed state, the triangles \mathbf{abc} and \mathbf{ade} undergo rigid-body motion and relative rotation, moving to $\widetilde{\mathbf{abc}}$ and $\widetilde{\mathbf{ade}}$, respectively. Following Equations (S104)–(S111), we can set up the local coordinate frames on the triangles \mathbf{abc} , $\widetilde{\mathbf{abc}}$, \mathbf{ade} , and $\widetilde{\mathbf{ade}}$, respectively. Then we can calculate the rotation matrix $\widetilde{\mathbf{R}}$ and the rotation axis $\widetilde{\mathbf{n}}$. We write the rotation axis $\widetilde{\mathbf{n}}$ as

$$\widetilde{\mathbf{n}} = \widetilde{n}_1 \widetilde{\mathbf{m}}_1 + \widetilde{n}_2 \widetilde{\mathbf{m}}_2 + \widetilde{n}_3 \widetilde{\mathbf{m}}_3, \quad (\text{S125})$$

where the basis vectors $\widetilde{\mathbf{m}}_1, \widetilde{\mathbf{m}}_2, \widetilde{\mathbf{m}}_3$ are attached on the triangle $\widetilde{\mathbf{abc}}$. The components $\widetilde{n}_1, \widetilde{n}_2, \widetilde{n}_3$ are prescribed according to the vertex positions $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}$ and $\widetilde{\mathbf{a}}, \widetilde{\mathbf{b}}, \widetilde{\mathbf{c}}, \widetilde{\mathbf{d}}, \widetilde{\mathbf{e}}$, which are all known position vectors for a given ori-kiri assemblage that is optimized to have two compatible configurations.

Now we investigate an intermediate configuration other than the compact or the compatible deployed configuration. We suppose the vertices move to new positions $\mathbf{a}', \mathbf{b}', \mathbf{c}', \mathbf{d}', \mathbf{e}'$ at the intermediate state. If we replace the vertices $\widetilde{\mathbf{a}}, \widetilde{\mathbf{b}}, \widetilde{\mathbf{c}}, \widetilde{\mathbf{d}}, \widetilde{\mathbf{e}}$ by $\mathbf{a}', \mathbf{b}', \mathbf{c}', \mathbf{d}', \mathbf{e}'$ in Equations (S104)–(S108), we can obtain the local basis vectors $\mathbf{m}'_1, \mathbf{m}'_2, \mathbf{m}'_3$ on the triangle $\mathbf{a'b'c'}$, the local basis vectors $\mathbf{n}'_1, \mathbf{n}'_2, \mathbf{n}'_3$ on the triangle $\mathbf{a'd'e'}$, and the rotation matrix \mathbf{R}' for the intermediate configuration. Here, the matrix \mathbf{R}' transforms the triangle \mathbf{ade} to the triangle $\mathbf{a'd'e'}$ with respect to the material coordinate frame attached on the triangle \mathbf{abc} . On the undeformed (compact) configuration, the material coordinate frame has the basis vectors $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ on the triangle \mathbf{abc} . On the deformed (intermediate) configuration, the material coordinate frame has the basis vectors $\mathbf{m}'_1, \mathbf{m}'_2, \mathbf{m}'_3$ on the triangle $\mathbf{a'b'c'}$. It is worth noting that the intermediate configuration is not necessarily compatible with the compact configuration. In other word, the triangles \mathbf{abc} and \mathbf{ade} can undergo non-rigid deformation and evolve to $\mathbf{a'b'c'}$ and $\mathbf{a'd'e'}$, respectively. Consequently, the material coordinate frames on the triangles $\mathbf{a'b'c'}$ and $\mathbf{a'd'e'}$ can be disturbed by the non-rigid deformation. Under this circumstance, there is no straightforward way to characterize the relative rotation of the triangle

ade with respect to the triangle **abc**. To avoid this issue, we assume the panels of the ori-kiri assemblages are stiff enough such that the bars can only have small extension or compression. Under this assumption, the matrix \mathbf{R}' can approximately characterize the relative rotation between the triangle **a'd'e'** and the triangle **a'b'c'**. Then we can calculate the rotation axis \mathbf{n}' and the rotation angle γ' corresponding to the matrix \mathbf{R}' according to Equations (S109) and (S111). We write the rotation axis \mathbf{n}' as

$$\mathbf{n}' = n'_1 \mathbf{m}'_1 + n'_2 \mathbf{m}'_2 + n'_3 \mathbf{m}'_3. \quad (\text{S126})$$

We note that the rotation axis \mathbf{n}' is a virtual axis that reflects the direction in which the triangles **a'b'c'** and **a'd'e'** rotate with respect to each other. We denote the axis of the physical hinge by $\tilde{\mathbf{n}}'$ at the intermediate state. Then we can express $\tilde{\mathbf{n}}'$ as

$$\tilde{\mathbf{n}}' \approx \tilde{n}_1 \mathbf{m}'_1 + \tilde{n}_2 \mathbf{m}_2 + \tilde{n}_3 \mathbf{m}'_3, \quad (\text{S127})$$

where the basis vectors $\mathbf{m}'_1, \mathbf{m}'_2, \mathbf{m}'_3$ are attached on the triangle **a'b'c'**. The components of $\tilde{\mathbf{n}}'$ are approximately the same as the hinge axis $\tilde{\mathbf{n}}$ on the triangle $\tilde{\mathbf{abc}}$ because we have assumed the bars can only have small deformation. Then, at the intermediate state, if we find $\mathbf{n}' = \tilde{\mathbf{n}}'$, that is, $n'_1 = \tilde{n}_1, n'_2 = \tilde{n}_2, n'_3 = \tilde{n}_3$, we can say that the relative rotation between the triangles **a'b'c'** and **a'd'e'** is around the axis of the physical hinge. The reason is that the rotation axis of real motion $\tilde{\mathbf{n}}'$ coincides with the rotation axis of the physical hinge $\tilde{\mathbf{n}}$. In this case, there is no off-axial-rotation energy. Otherwise, if we find $\mathbf{n}' \neq \tilde{\mathbf{n}}'$, the relative rotation between the triangles **a'b'c'** and **a'd'e'** is off the axis of the physical hinge $\tilde{\mathbf{n}}$, and the off-axial-rotation energy is induced.

To characterize the off-axial-rotation energy, we use $\gamma' \mathbf{n}'$, the vector pointing towards the rotation axis \mathbf{n}' and having the length equal to the rotation angle γ' , to represent the total relative rotation between the triangles **a'b'c'** and **a'd'e'**. To extract the off-axial rotation from the total relative rotation, we decompose the $\gamma' \mathbf{n}'$ into

$$\gamma' \mathbf{n}' = \gamma \tilde{\mathbf{n}}' + \delta \tilde{\mathbf{n}}'_\perp, \quad (\text{S128})$$

where

$$\gamma = \gamma' \mathbf{n}' \cdot \tilde{\mathbf{n}}', \quad (\text{S129})$$

and $\tilde{\mathbf{n}}'_\perp$ is the unit vector that is perpendicular to the hinge axis $\tilde{\mathbf{n}}'$. In general, the rotation around

an axis \mathbf{n} by an angle α can be expressed by the matrix function $\mathbf{R}(\mathbf{n}, \alpha)$ as

$$\mathbf{R}(\mathbf{n}, \alpha) = \mathbf{I} \cos \alpha + (1 - \cos \alpha) \mathbf{n} \otimes \mathbf{n} - \boldsymbol{\varepsilon} \cdot \mathbf{n} \sin \alpha, \quad (\text{S130})$$

where \mathbf{I} is the 3×3 identity matrix and $\boldsymbol{\varepsilon}$ is the $3 \times 3 \times 3$ permutation tensor whose components are given by the 3D Levi-Civita symbol. We assume δ is a quantity that is much smaller than π , then we can obtain

$$[\mathbf{R}(\tilde{\mathbf{n}}'_\perp, \delta) \mathbf{R}(\tilde{\mathbf{n}}', \gamma)] [\mathbf{R}(\mathbf{n}', \gamma')]^{-1} = \mathbf{I} + O(\delta), \quad (\text{S131})$$

where $O(\delta)$ is a small quantity of the same order of magnitude as δ . Thus, the total relative rotation between the triangles $\mathbf{a}'\mathbf{b}'\mathbf{c}'$ and $\mathbf{a}'\mathbf{d}'\mathbf{e}'$ can be decomposed into the combination of the two successive rotations around the hinge axis $\tilde{\mathbf{n}}'$ and around the perpendicular axis $\tilde{\mathbf{n}}'_\perp$, respectively. The off-axial-rotation energy can be considered as the result of the second rotation, and the corresponding rotation angle δ can be obtained by

$$\delta = \|\gamma' \mathbf{n}' - (\gamma' \mathbf{n}' \cdot \tilde{\mathbf{n}}') \tilde{\mathbf{n}}'\|. \quad (\text{S132})$$

We use a linear rotational spring of stiffness k_R to model the source of the restoring force induced by the off-axial rotation, then the off-axial-rotation energy can be formulated as $k_R \delta^2 / 2$. Supposing the total number of hinges is N_{hinge} in the truss model, we can write the total off-axial-rotation energy E_R as

$$E_R(\mathbf{X}) = \frac{1}{2} \sum_{m=1}^{N_{\text{hinge}}} (k_R \delta_m^2), \quad (\text{S133})$$

where δ_m is the off-axial-rotation angle at the m -th hinge; \mathbf{X} is the set containing all the 3D vertex positions of the deformed truss. Finally, we can write the total elastic energy of the truss model as

$$E(\mathbf{X}) = \frac{1}{2} \sum_{n=1}^{N_{\text{bar}}} [k_{S,n} (s'_n - s_n)^2] + \frac{1}{2} \sum_{m=1}^{N_{\text{hinge}}} (k_R \delta_m^2). \quad (\text{S134})$$

We assume a constant axial rigidity EA for all the bars, then we have $k_{S,n} = EA/s_n$. We introduce a characteristic length ℓ_c to nondimensionalize the deformation energy. Then, the scaled stretching energy and rotational energy are expressed by

$$\bar{E}_S(\mathbf{X}) = E_S(\mathbf{X}) / (EA \ell_c) \quad \text{and} \quad \bar{E}_R(\mathbf{X}) = E_R(\mathbf{X}) / (EA \ell_c), \quad (\text{S135})$$

respectively. Finally, we obtain the total scaled energy:

$$\bar{E}_T(\mathbf{X}) = \bar{E}_S(\mathbf{X}) + \bar{E}_R(\mathbf{X}). \quad (\text{S136})$$

We assume $k_R = c_R EA \ell_c$, where c_R is a coefficient adjusting the proportion of \bar{E}_S and \bar{E}_R in \bar{E}_T . The coefficient can be expressed in terms of the stiffness by $c_R = k_R / (k_{S,n} s_n \ell_c)$. For a compact cube with an RS mesh, we set ℓ_c with the half length of the cube L . For a compact sphere with either an RS or an UV mesh, we set ℓ_c with the radius of the sphere R . By trial and error, we assume that $c_R = 0.1$, such that we can obtain both small off-axial-rotation angles and small bar strains (see Figure S14), which are consistent with our assumptions in the derivation above.

Optimization for energy landscapes. To obtain the energy landscapes upon deployment of the ori-kiri assemblages, we divide the deployment process into successive steps and minimize the energy $E(\mathbf{X})$ of the truss model in each step. To guide the deployment from the compact state to the compatible deployed state, we select some control nodes $\mathbf{x}_{\text{ctrl.}}$ and prescribe their positions upon deployment. We refer to the nodes other than the control nodes as the free nodes $\mathbf{x}_{\text{free.}}$. The control nodes $\mathbf{x}_{\text{ctrl.}}$ and the free nodes $\mathbf{x}_{\text{free.}}$ constitute all the elements in the set \mathbf{X} . We denote the number of deployment steps by N_{step} . At the compact state, the positions of the control and the free nodes are denoted by $\mathbf{x}_{\text{c-ctrl.}}$ and $\mathbf{x}_{\text{c-free.}}$, respectively. At the compatible deployed state, the positions of the control nodes and the free nodes are denoted by $\mathbf{x}_{\text{d-ctrl.}}$ and $\mathbf{x}_{\text{d-free.}}$, respectively. Then the energy landscapes can be obtained by implementing the procedure as follows.

1. Give the step number N_{step} and the select the control nodes $\mathbf{x}_{\text{ctrl.}}$.
2. At each step $k = 1, 2, \dots, N_{\text{step}}$, specify the spatial coordinates of the control nodes as $\mathbf{x}_{\text{ctrl.}}(k) = (1 - k/N_{\text{step}})\mathbf{x}_{\text{c-ctrl.}} + (k/N_{\text{step}})\mathbf{x}_{\text{d-ctrl.}}$.
3. At each step $k = 1, 2, \dots, N_{\text{step}}$, fix the control nodes $\mathbf{x}_{\text{ctrl.}}(k)$ and optimize the spatial coordinates of the free nodes $\mathbf{x}_{\text{free.}}$ to minimize the scaled energy $\bar{E}(\mathbf{X})$.

At each step k , we initialize the free nodal positions, denoted by $\mathbf{x}_{\text{free.,in}}(k)$, with the optimal nodal positions at the $(k - 1)$ -th step, denoted by $\mathbf{x}_{\text{free.,opt.}}(k - 1)$. The only exception is for the first step: to trigger the deployment of the truss model, we disturb all the compact free nodes towards the deployed configuration. Therefore, the initial nodal positions $\mathbf{x}_{\text{free.,in}}(k)$ can be expressed as

$$\mathbf{x}_{\text{free.,in}}(k) = \begin{cases} (1 - 1/N_{\text{step}})\mathbf{x}_{\text{c-free}} + (1/N_{\text{step}})\mathbf{x}_{\text{d-free}}, & k = 1, \\ \mathbf{x}_{\text{free.,opt.}}(k - 1), & k \geq 2. \end{cases} \quad (\text{S137})$$

It is worth noting that the selection of the control nodes can affect the results of the deployment simulation. We select the control nodes based on the following considerations. First, the control nodes should appear as pairs to characterize the opening trend of the slits. Second, the control nodes should travel a relatively long distance compared to the free nodes to effectively distinguish each step of the deployment. Third, the number of control nodes should be as small as possible to avoid excessively increasing the overall stiffness of the truss model. We select different control nodes for different ori-kiri assemblages as follows:

for $N \times N \times 6$ RS meshes of compact cubic shape:

$$\mathbf{x}_{\text{ctrl.}} \in \{\mathbf{x}_{N/2+1, N/2+1, k, 1}, \mathbf{x}_{N/2+1, N/2, k, 4} \mid k = 1, 2, \dots, 6\};$$

for $N \times N \times 6$ RS meshes of compact spherical shape:

$$\mathbf{x}_{\text{ctrl.}} \in \left\{ \begin{array}{cccc} \mathbf{x}_{1,1,2,2}, & \mathbf{x}_{1,1,4,2}, & \mathbf{x}_{1,1,6,1}, & \mathbf{x}_{1,1,5,1}, \\ \mathbf{x}_{N,1,2,1}, & \mathbf{x}_{N,1,4,1}, & \mathbf{x}_{N,1,6,2}, & \mathbf{x}_{N,1,5,2}, \\ \mathbf{x}_{N,N,2,4}, & \mathbf{x}_{N,N,4,4}, & \mathbf{x}_{N,N,6,3}, & \mathbf{x}_{N,N,5,3}, \\ \mathbf{x}_{1,N,2,3}, & \mathbf{x}_{1,N,4,3}, & \mathbf{x}_{1,N,6,4}, & \mathbf{x}_{1,N,5,4}, \end{array} \right\};$$

for $M \times N$ UV meshes of compact spherical shape:

$$\mathbf{x}_{\text{ctrl.}} \in \{\mathbf{x}_{i, N/2+1, 1}, \mathbf{x}_{i, N/2, 4} \mid i = 1, 3, \dots, M\}. \quad (\text{S138})$$

We can see that the control nodes for a compact RS cube are located at the center of each of the six faces, while the control nodes for a compact RS sphere are located at the corners of the six faces. For a compact UV sphere, the control nodes are located at the outer circumference of the sphere.

We use the sequential quadratic programming (SQP) algorithm to solve the optimization problem at each step of the deployment. To accelerate the convergence of the optimization algorithm, we impose the constraints of symmetry to the nodal positions which are consistent with the symmetry of the ori-kiri assemblages. The energy curves (see Figures 3D, 3E, 3F in the main text and movies S1, S3, S5) stand for the evolution of the dimensionless energy \bar{E} given by Equation (S136) with respect to the pseudo time t in the deployment process. The pseudo time t is simply defined by scaling the step number k to $t = k/N_{\text{step}}$. Finally, we note that the optimization results are affected by the relative location of the compact and compatible deployed configurations, because the boundary conditions for the minimum energy problem are given by the positions of the control nodes, which move from their initial locations on the compact configuration to the final locations on the

compatible deployed configuration step by step. To eliminate the effect of rigid-body motion on the deployment simulation, we place the compact and compatible deployed configurations in the same coordinates system, such that the centroids of the enclosed surfaces coincide with each other.

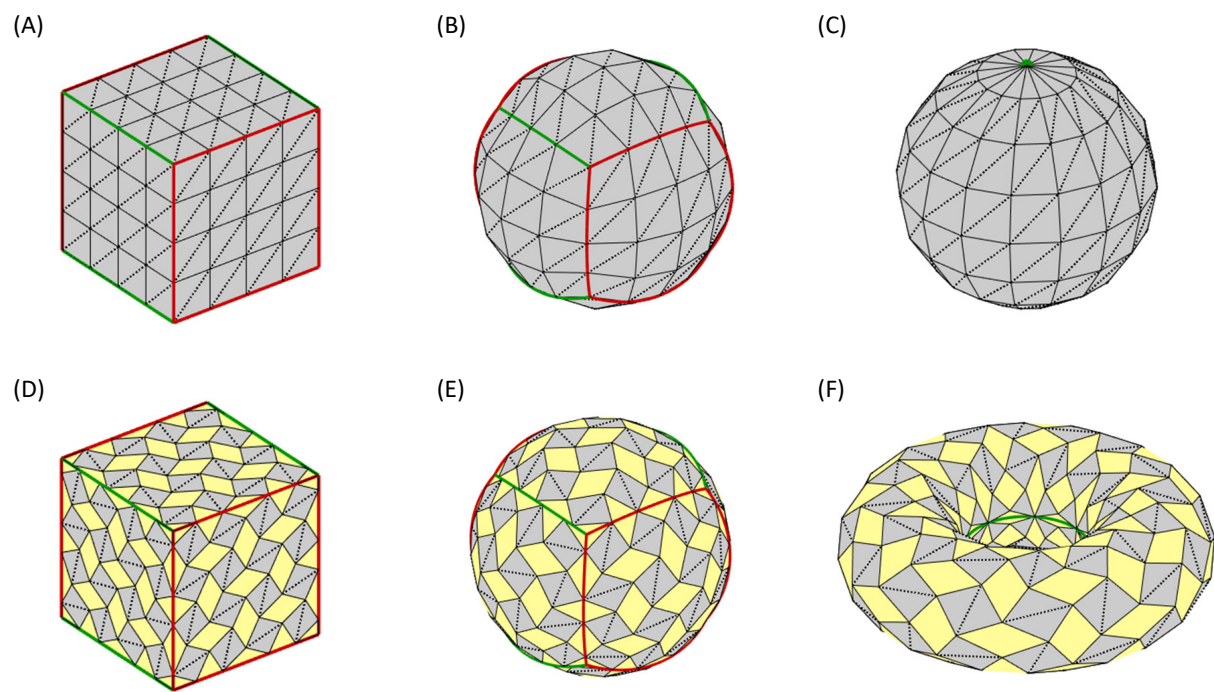


Figure S1: RS and UV meshes that enclose different surfaces. (A) The compact RS mesh that encloses the cube. (B) The compact RS mesh that encloses a sphere. (C) The compact UV mesh that encloses a sphere. (D) The deployed RS mesh that encloses the cube. (E) The deployed RS mesh that encloses a sphere. (F) The deployed UV mesh that encloses a torus.

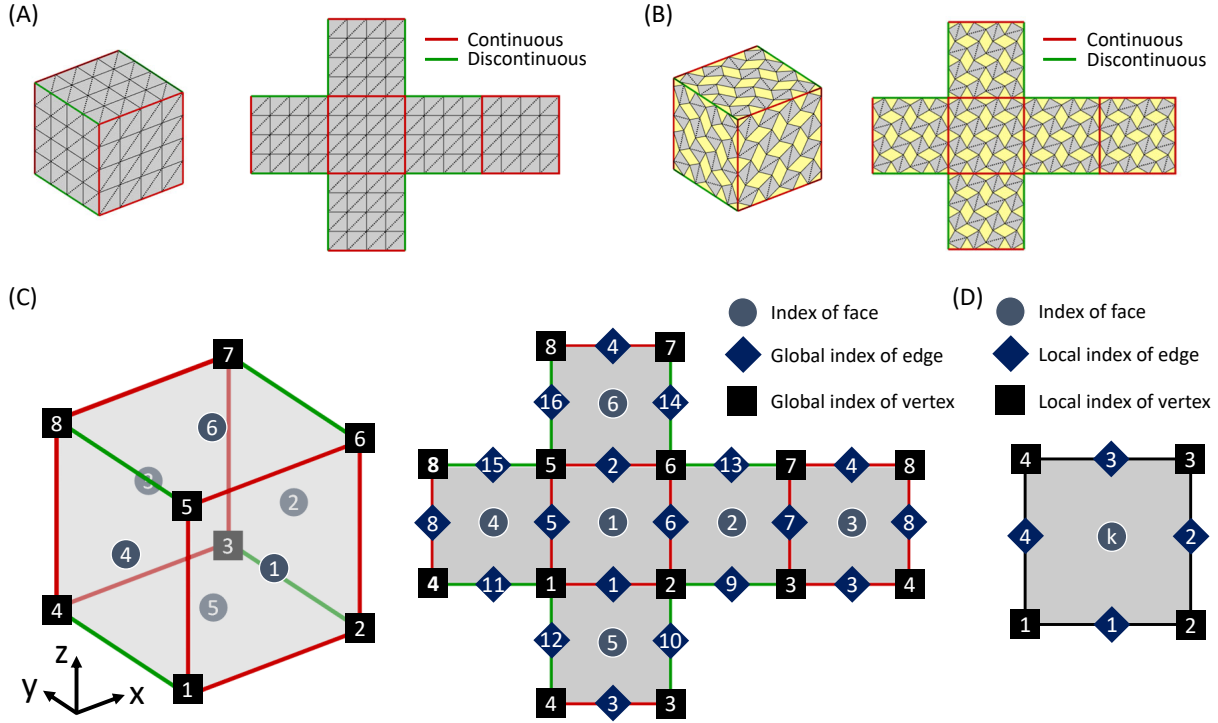


Figure S2: Faces and nodes of the RS mesh. (A) and (B) The cubic RS mesh and its developed pattern at (A) the compact state and (B) the deployed state. The continuous and discontinuous intersections of the six faces are highlighted in red and green, respectively. (C) The global indices of the six faces, eight vertices, and sixteen edges. (D) The local indices of the four edges and four vertices for each face.

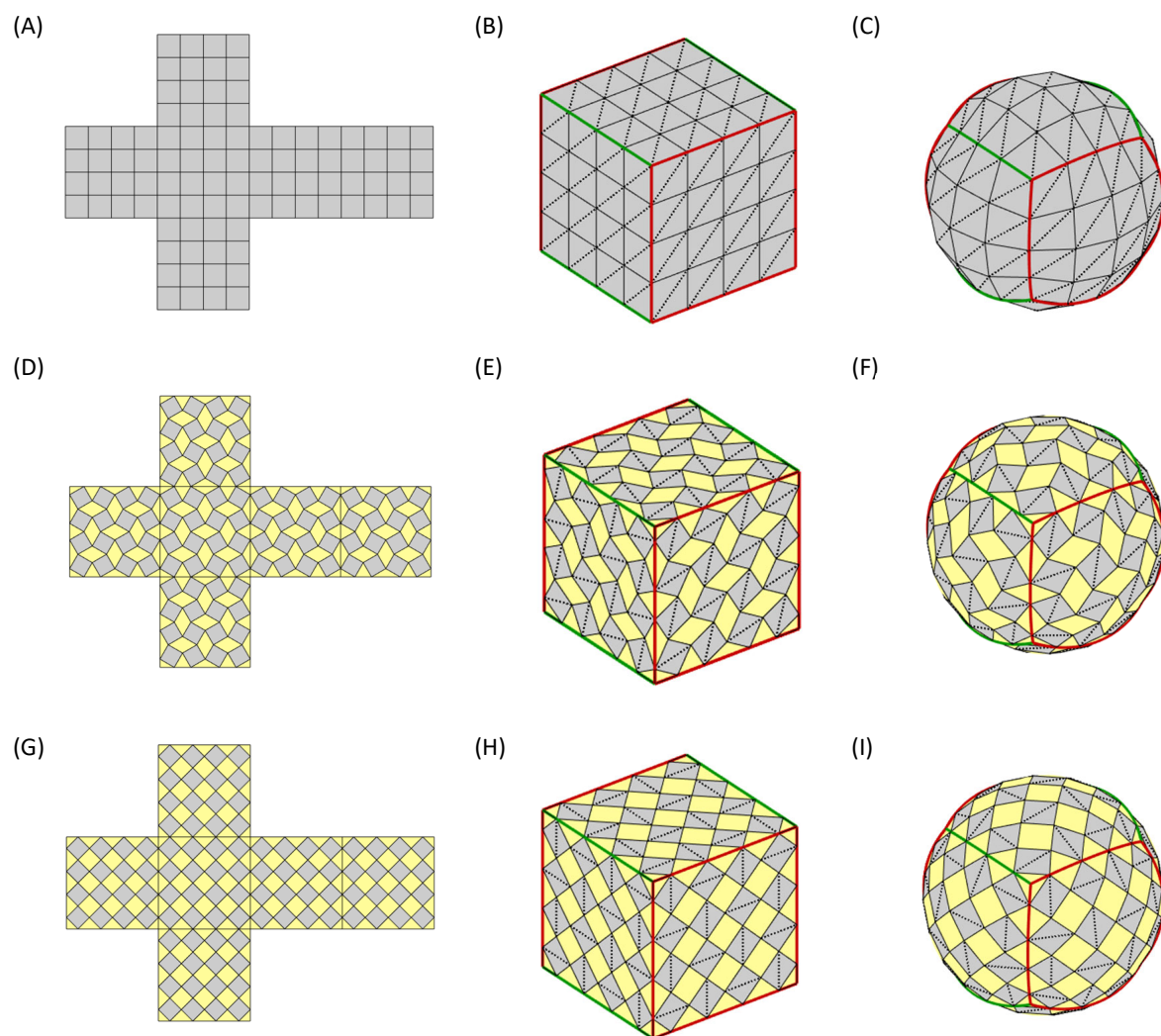


Figure S3: Regular RS-parameter meshes and the corresponding RS meshes. (A) Compact RS-parameter mesh, which can generate (B) a compact cubic RS mesh and (C) a compact spherical RS mesh. (D) Deployed RS-parameter mesh with uniform opening angles $\pi/3$, which can generate (E) the deployed cubic RS mesh and (F) the deployed spherical RS mesh. (G) Deployed RS-parameter meshes with uniform opening angles $\pi/2$, which can generate (H) the deployed cubic RS mesh and (I) the deployed spherical RS mesh.

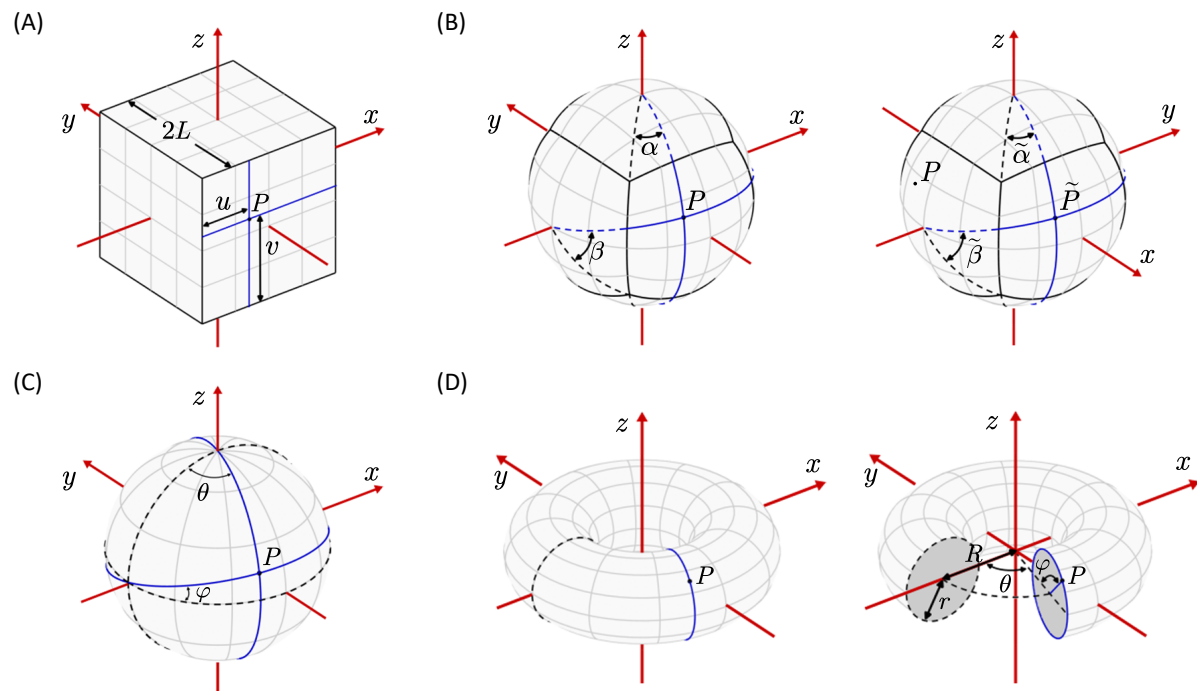


Figure S4: Parametrization of a point P on different closed surfaces. (A) The cube with RS mesh. (B) The sphere with RS mesh. (C) The sphere with UV mesh. (D) The torus with UV mesh.

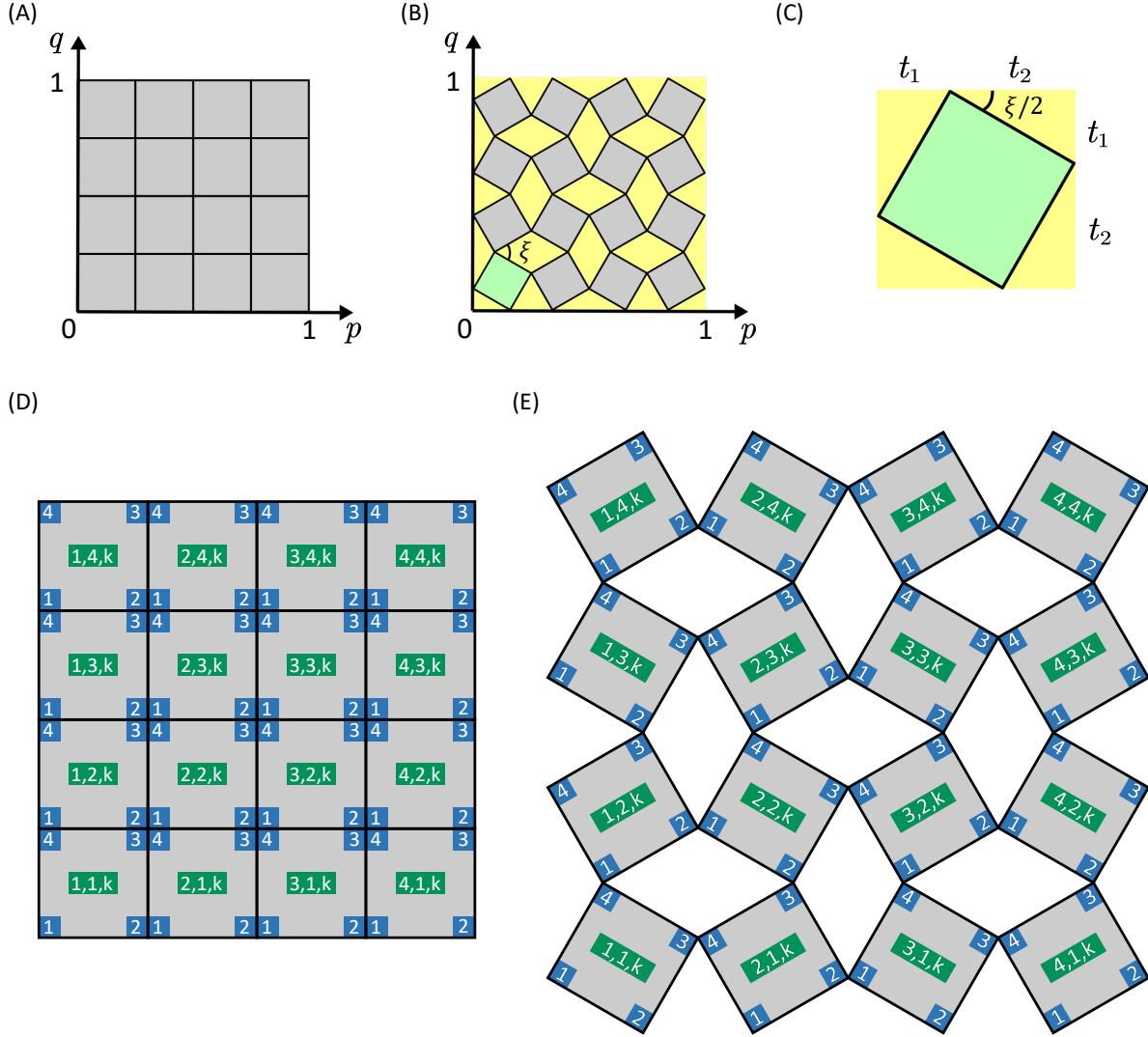
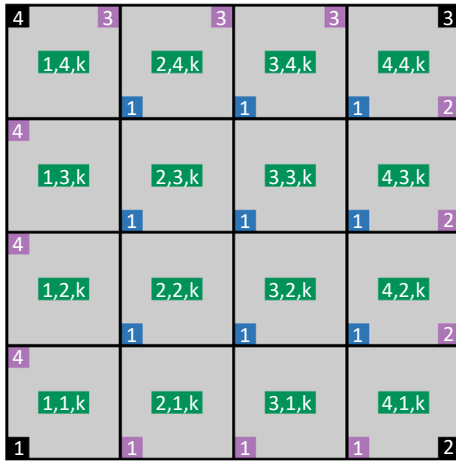


Figure S5: Regular RS-parameter meshes. (A) The compact RS-parameter mesh. (B) The deployed RS-parameter mesh. (C) Geometric notations of an obliquely placed square on the deployed RS-parameter mesh. (D) Indices of cells and nodes on the compact RS-parameter mesh. (E) Indices of cells and nodes on the deployed RS-parameter mesh.

(A)



- Index of panel
- Local index of interior vertex
- Local index of boundary vertex
- Local index of corner vertex

(B)

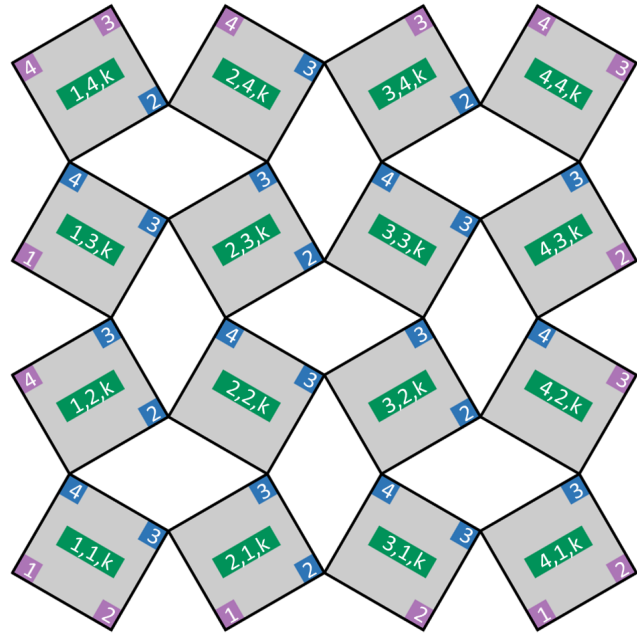


Figure S6: Interior vertices, boundary vertices, and corner vertices in the parameter space.

(A) The compact RS-parameter mesh. (B) The deployed RS-parameter mesh.

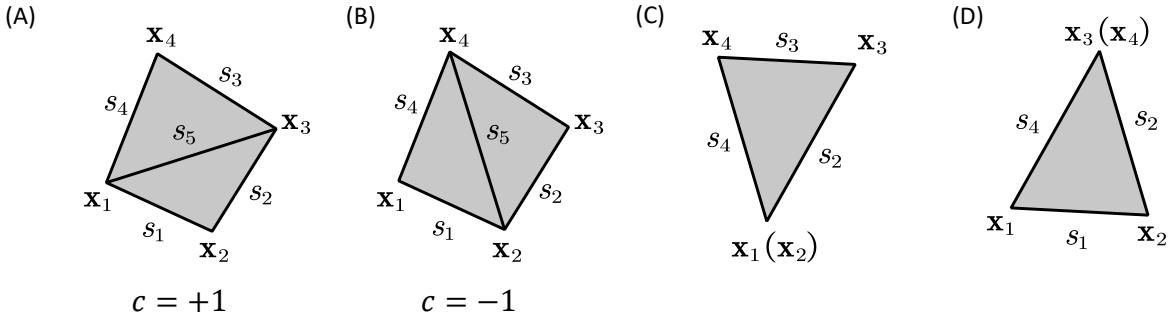
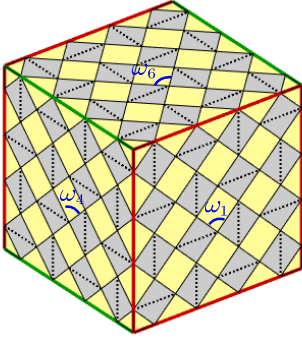


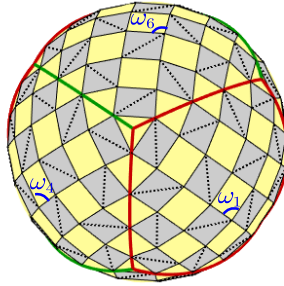
Figure S7: Metrics of panels. (A) The quadrilateral panel with a crease connecting vertices 1 and 3. (B) The quadrilateral panel with a crease connecting vertices 2 and 4. (C) The triangular panel that is degenerated from a quadrilateral with zero side length s_1 . (D) The triangular panel that is degenerated from a quadrilateral with zero side length s_3 . The crease length s_5 of a triangular panel is considered to be zero.

(A)

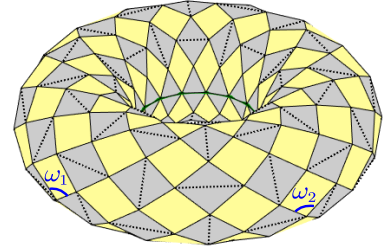


Constant reference opening angle:
 $\omega_1 = \omega_2 = \omega_3 = \omega_4 = \omega_5 = \omega_6 = \omega$

(B)



(C)



Constant reference opening angle:
 $\omega_1 = \omega_2 = \omega_3 = \omega_4 = \omega$

Figure S8: Reference opening angles of the RS and UV meshes. (A) RS mesh on a cube. **(B)** RS mesh on a sphere. The reference opening angles ω_2 , ω_3 , and ω_5 are at the opposite locations (backside) with respect to the angles ω_4 , ω_1 , and ω_6 . **(C)** UV mesh on a torus. The reference opening angles ω_3 and ω_4 are at the opposite locations (backside) with respect to the angles ω_1 and ω_2 .

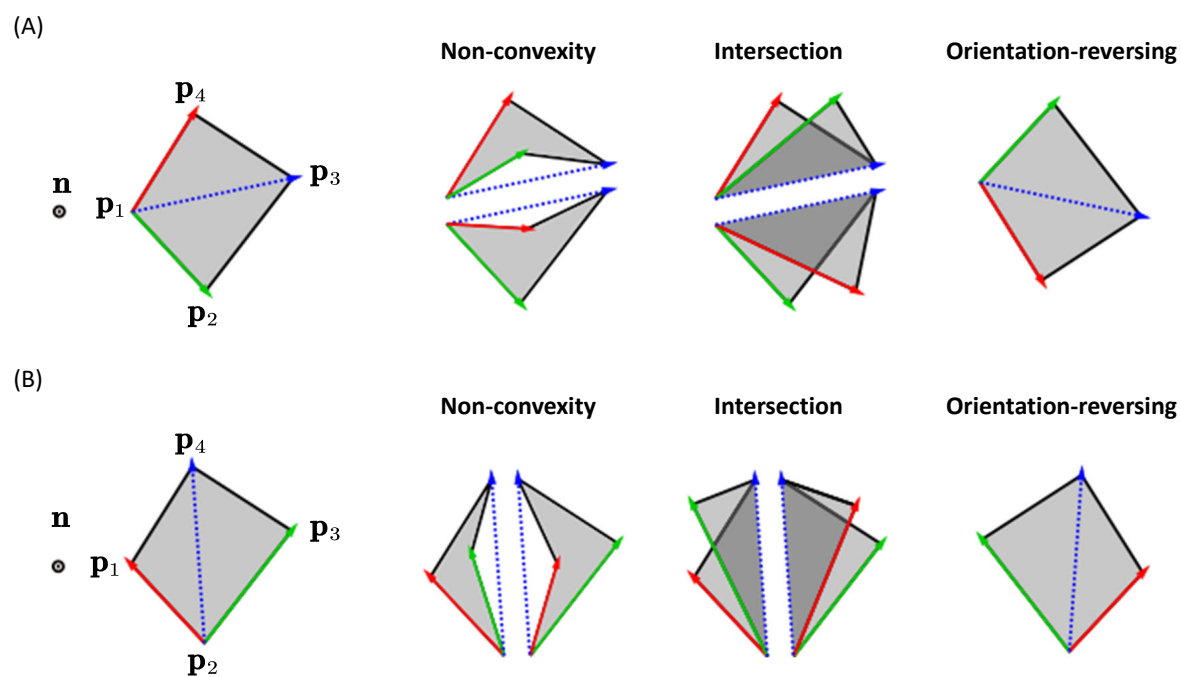


Figure S9: Valid (far left) and distorted (three on the right) cases of the planar quadrilaterals in the parameter space. (A) The diagonal connects points 1 and 3. (B) The diagonal connects points 2 and 4.

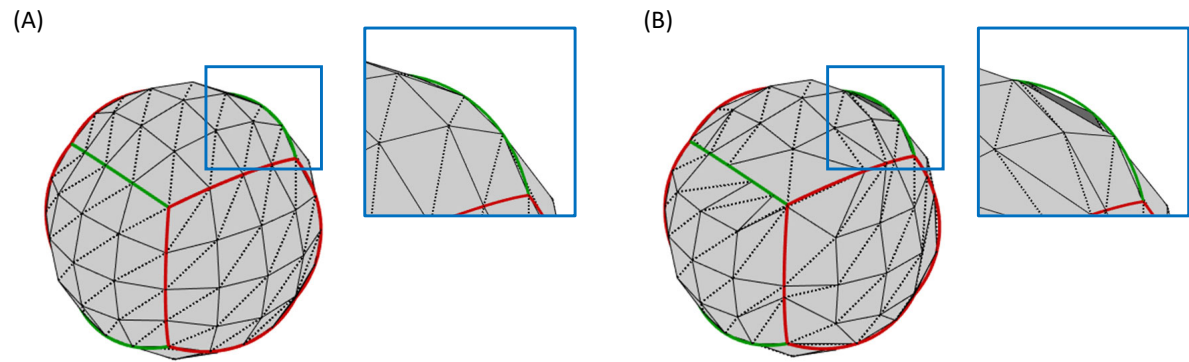


Figure S10: Compactness of RS meshes. (A) The compact RS mesh that has no slits at the discontinuous intersections. (B) The compact RS mesh that has slits at the discontinuous intersections.

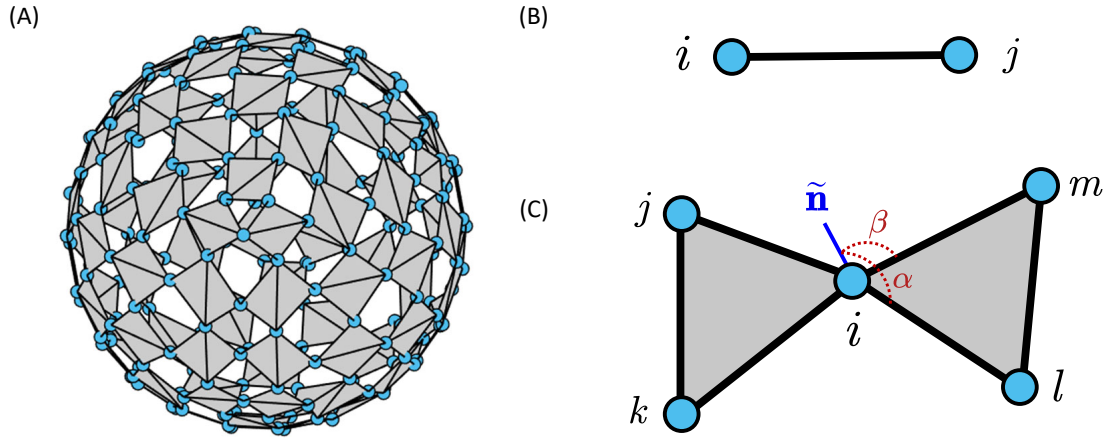


Figure S11: Kinematic constraints of the truss model. (A) The truss model of an ori-kiri assemblage. For better visual effects, the panel faces enclosed by the bars are highlighted in gray. (B) The constraint for inextensible bars. (C) The constraint for axial rotation.

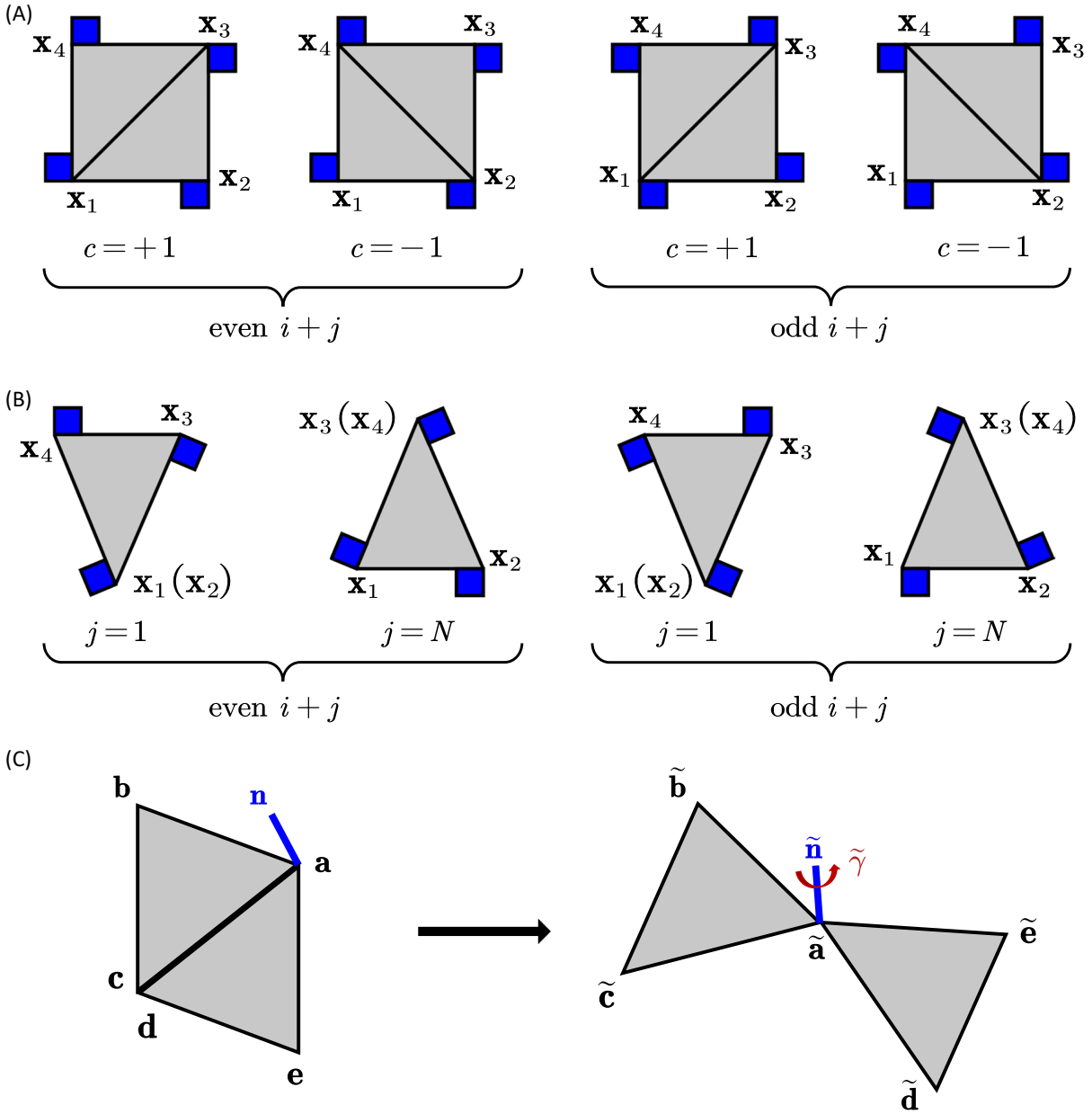


Figure S12: Hinges that are attached to the panels. (A) Quadrilateral panels (divided into two triangles by a crease) and the locations where the hinges are attached. (B) Triangular panels and the locations where the hinges are attached. (C) The hinge guides the relative rotation between two panels in such a way that its orientation is fixed with respect to the frame of reference of each panel.

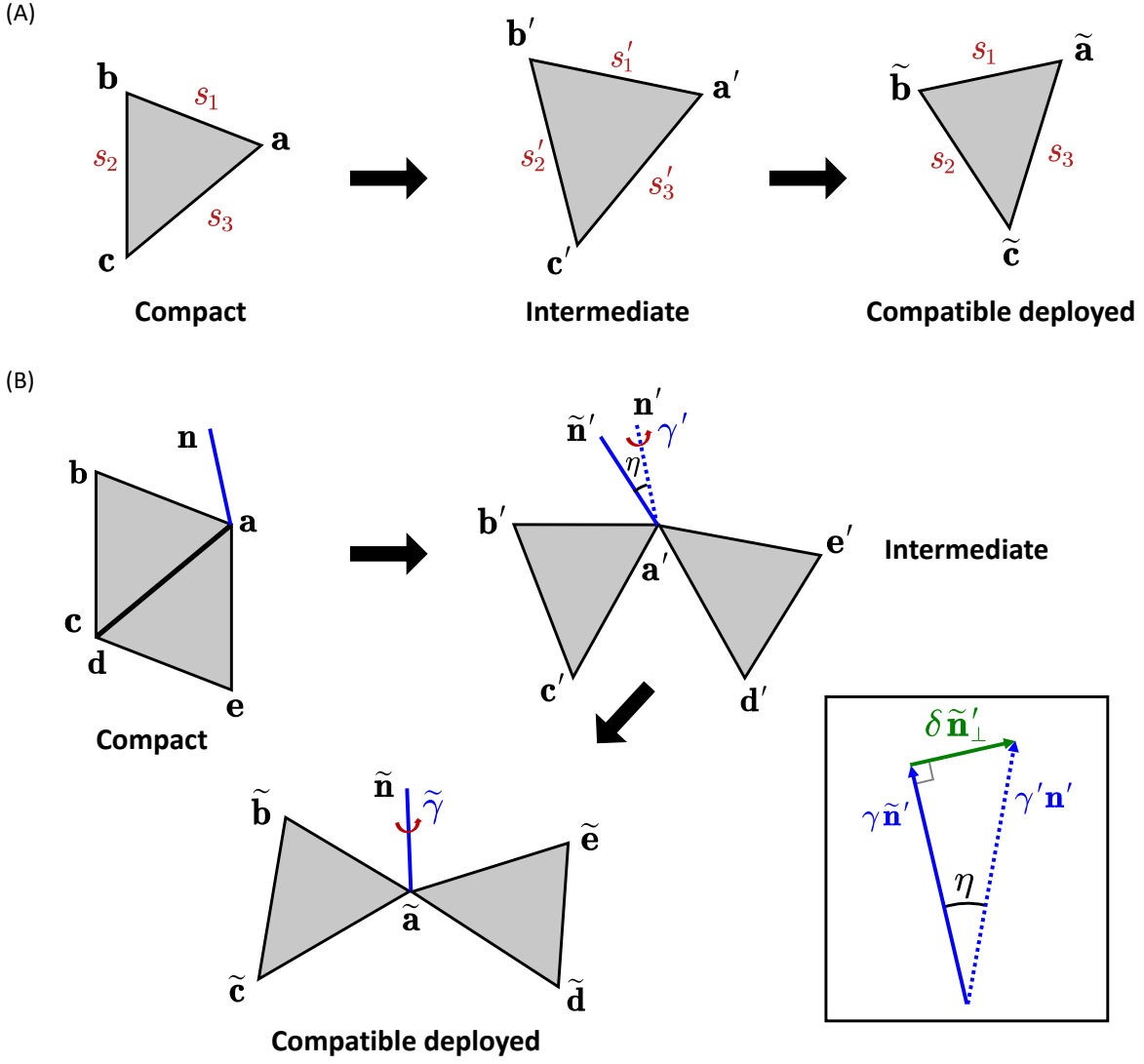


Figure S13: Two sources of elastic energy of the truss model. (A) The stretching energy of bars.

(B) The off-axial-rotation energy at hinges.

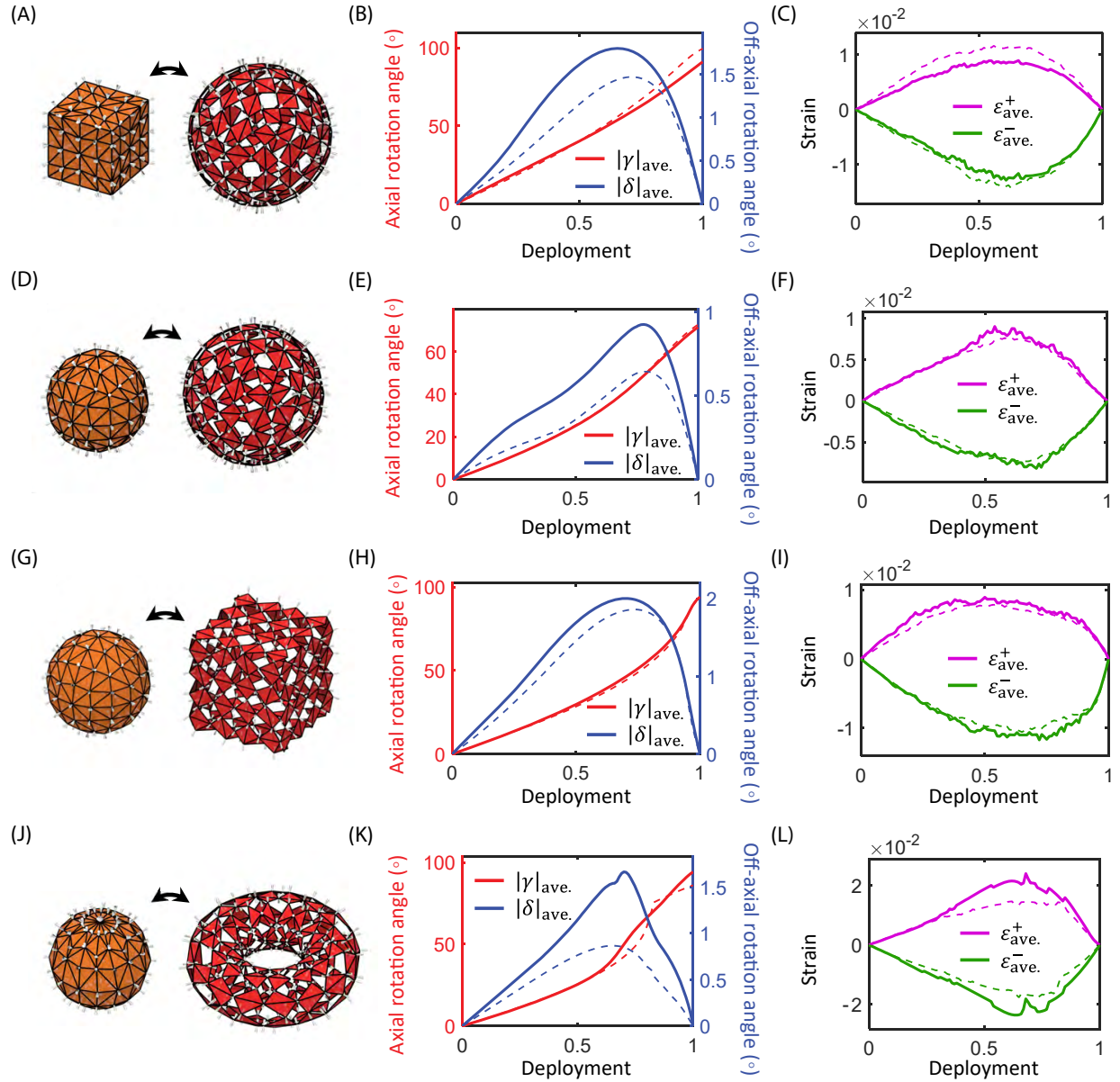


Figure S14: Geometric metrics evolving along the deployment path of the bistable ori-kiri assemblages. (A–C) Shape morphing between a compact cube and a deployed sphere. (D–F) Scaling between two spheres. (G–I) Shape morphing between a compact sphere and a deployed cube. (J–L) Topology morphing between a compact sphere and a deployed torus. Each morphing case includes: axial-rotation angle $|\gamma|_{\text{ave.}}$ and off-axial-rotation angle $|\delta|_{\text{ave.}}$, absolute and averaged for all hinges (B, E, H, K); positive bar strain $\varepsilon_{\text{ave.}}^+$ and negative bar strain $\varepsilon_{\text{ave.}}^-$, averaged for all bars (C, F, I, L). Dashed lines represent data from models with increased number of panels, i.e., $6 \times 6 \times 6$ for (A–I) and 24×12 for (J–L).

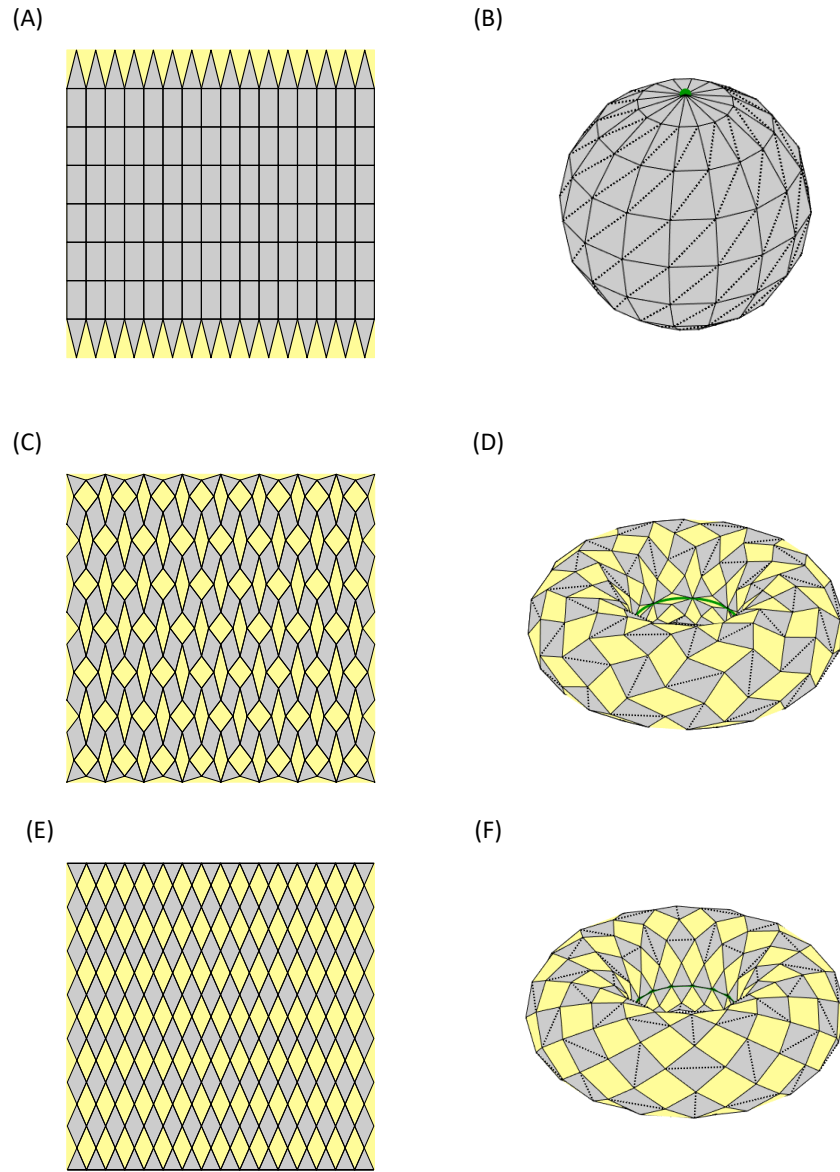


Figure S15: Regular UV-parameter meshes and the corresponding UV meshes. (A) Compact UV-parameter mesh, which can generate (B) the compact spherical UV mesh. (C) Deployed UV-parameter mesh which can generate (D) the deployed toric UV mesh. (E) Deployed UV-parameter mesh with which can generate (F) another deployed toric UV mesh with different opening angles.

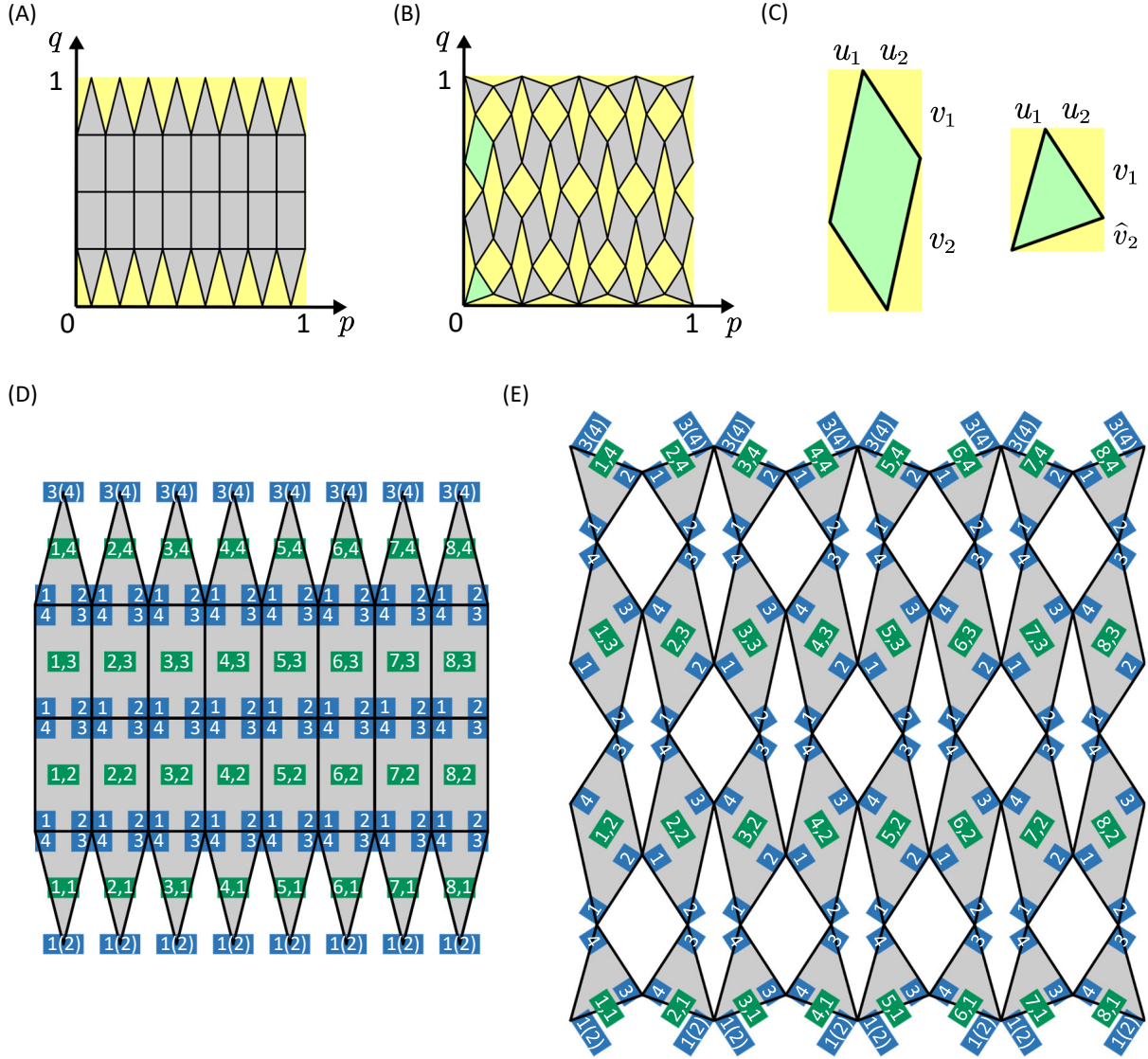


Figure S16: Regular UV-parameter mesh. (A) Compact UV-parameter mesh. (B) Deployed UV-parameter mesh. (C) Geometric notations of a quadrilateral cell and a triangle cell on the deployed UV-parameter mesh. (D) Indices of the cells and nodes on the compact UV-parameter mesh. (E) Indices of the cells and nodes on the deployed UV-parameter mesh.

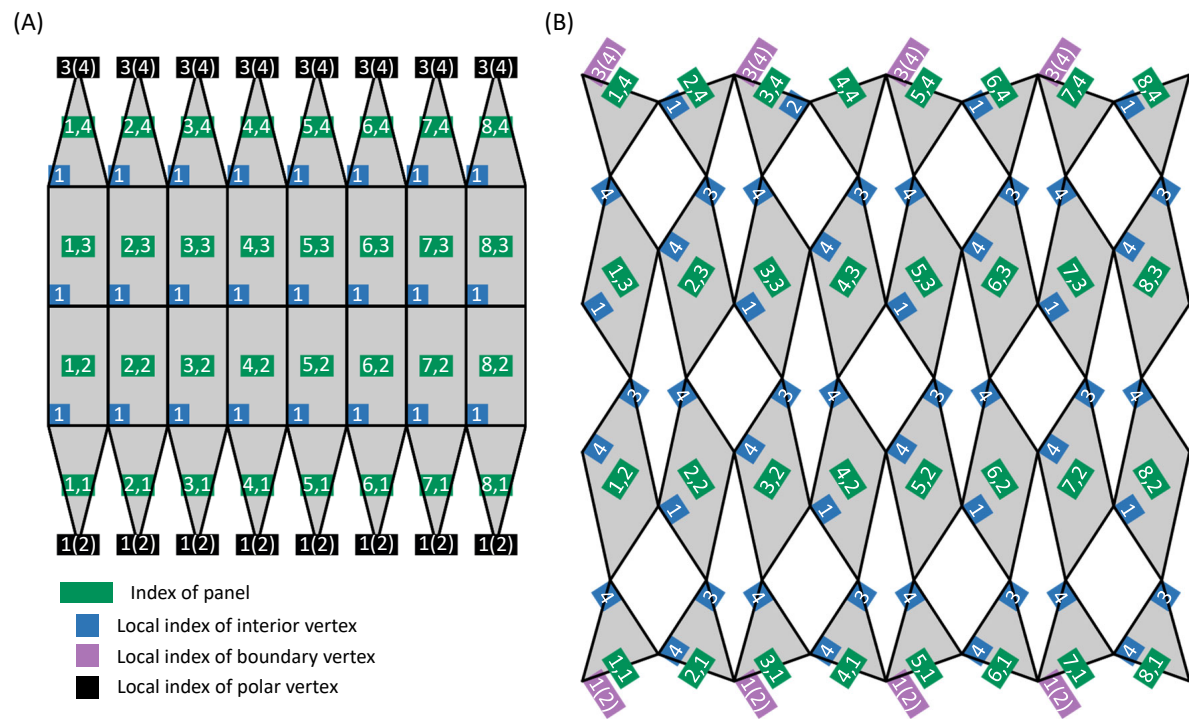


Figure S17: Interior vertices, boundary vertices, and corner vertices in the parameter space.

(A) The compact UV-parameter mesh. (B) The deployed UV-parameter mesh.

Movie S1. The deployment and energy landscape of bistable ori-kiri assemblages.

Movie S2. Shape morphing between a cube and a sphere.

Movie S3. Topology morphing between a sphere and a torus.

Movie S4. The deployment and energy landscape of bistable ori-kiri assemblages with increased number of panels.

Movie S5. The deployment and energy landscape of bistable ori-kiri assemblages with irregular shapes.