Jigsaw puzzle design of pluripotent origami

Peter Dieleman,[1,2] Niek Vasmel,[1] Scott Waitukaitis,[2,1] and Martin van Hecke[1,2]

[1]*Huygens-Kamerling Onnes Laboratories, University of Leiden, Postbus 9504, 2300 RA, The Netherlands*
[2]*AMOLF, Science Park 104, 1098 XG Amsterdam, The Netherlands*

Origami is rapidly transforming the design of robots[1,2], deployable structures[3–6], and metamaterials[7–14]. However, as foldability requires a large number of complex compatibility conditions that are difficult to satisfy, the design of crease patterns is limited to heuristics and computer optimization. Here we introduce a systematic strategy that enables intuitive and effective design of complex crease patterns that are guaranteed to fold. First, we exploit symmetries to construct 140 distinct foldable motifs, and represent these as jigsaw puzzle pieces. We then show that when these pieces are fitted together they encode foldable crease patterns. This maps origami design to solving combinatorial problems, which allows us to systematically create, count, and classify a vast number of crease patterns. We show that all these crease patterns are *pluripotent*—capable of folding into multiple shapes—and solve exactly for the number of possible shapes for each pattern. Finally, we employ our framework to rationally design a crease pattern that folds into two independently defined target shapes, and fabricate such pluripotent origami. Our results provide physicists, mathematicians and engineers a powerful new design strategy.

Traditional origami is the artistic pursuit of folding two-dimensional paper into intricate, three-dimensional structures[15]. In recent years, physicists and engineers have leveraged origami as a powerful design tool, leading to vigorous activities to capture the physics of folding. In particular, origami-based mechanical metamaterials have generated intense interest by displaying exotic properties such as reprogrammability[8], multistability[9,10], and topological protection[11]. Such pursuits typically focus on *rigid origami*, which concerns perfectly stiff plates connected by flexible hinges that are agnostic as to their mountain-valley (MV) assignment. The absence of a presupposed MV pattern opens up the possibility of pluripotent origami—crease patterns that can fold into multiple 3D target shapes[16–19]. However, the assumption of rigid plates leads to complex compatibility conditions that make designing foldable patterns notoriously difficult. As a result, many studies are constrained to a severely limited set of known solutions, such as the Miura-ori[3,7,8,12,16,25]. Moreover, a design framework for systematically obtaining or characterizing pluripotent origami is lacking, and the design of crease patterns that rigidly fold, let alone into multiple target shapes, remains a formidable challenge.

We address this challenge with symmetry-based groups of 4-vertices—*i.e.*, units where four folds (or hinges) separated by four plates meet at a point (Fig. 1a). The underlying geometry of a 4-vertex is defined by its sector angles α_j. Folded states are characterized by the fold angles $\rho_{j,j+1}$, which are defined by the out of plane deviation between plates j and $j+1$ (Fig. 1a). As the simplest non-trivial structures that rigidly fold, these 'atoms' of origami form the basis of many well-known crease patterns[3,9,20,22,23]. We generate a group of 4-vertices by first selecting four generic sector angles $\{\alpha_j\}$ with $\Sigma\alpha_j = 2\pi$ (see the Supplementary Information for a discussion of which sector angles are sufficiently generic to work with our scheme). We then defining a **B**ase vertex with **a**nticlockwise ordered sector angles (denoted **Ba**), a clockwise-ordered ordered copy of this vertex (**Bc**), a supplemented vertex with anticlockwise ordered sector angles $\alpha'_j := \pi - \alpha_j$ (**Sa**), and a supplemented-clockwise vertex (**Sc**) (Fig. 1a). The design space we consider consists of crease patterns made of quadrilateral meshes composed exclusively from these four vertices (Fig. 1b).

For such crease patterns to be foldable, each set of vertices around each quadrilateral plate (*i.e.*, each 'Kokotsakis mesh') must satisfy two compatibility conditions. Labeling the vertices around a plate as W-Z (Fig. 1b), the 'sum' condition simply requires that the interior angles add to 2π, *i.e.*,

$$\alpha^W + \beta^X + \gamma^Y + \delta^Z = 2\pi \ . \tag{1}$$

The assumption of rigidity demands compatible evolution of the folding angles of the Kokotsakis mesh[25,26]. Mathematically, this can be captured by considering 'fold operators', $\mathbf{P_j}$, which for a given vertex map the fold angles adjacent to sector angle j in an anticlockwise manner: $\mathbf{P}_j(\rho_{j-1,j}) = \rho_{j,j+1}$. The demand that the sequential execution of operators on the folds around the quadrilateral yields the identity[27] leads to the 'loop' condition (Fig. 1b):

$$\mathbf{P}^Z_\delta \cdot \mathbf{P}^Y_\gamma \cdot \mathbf{P}^X_\beta \cdot \mathbf{P}^W_\alpha = \mathbf{I}. \tag{2}$$

Finding combinations of vertices that satisfy this condition is notoriously difficult: foldable Kokotsakis meshes have only recently been mathematically classified[28], and practical approaches for their generation have so far relied on heavily restricted cases[3,20,22], approximations[26] or computer optimization[23]. However, for our group of symmetry-related vertices, the folding operators

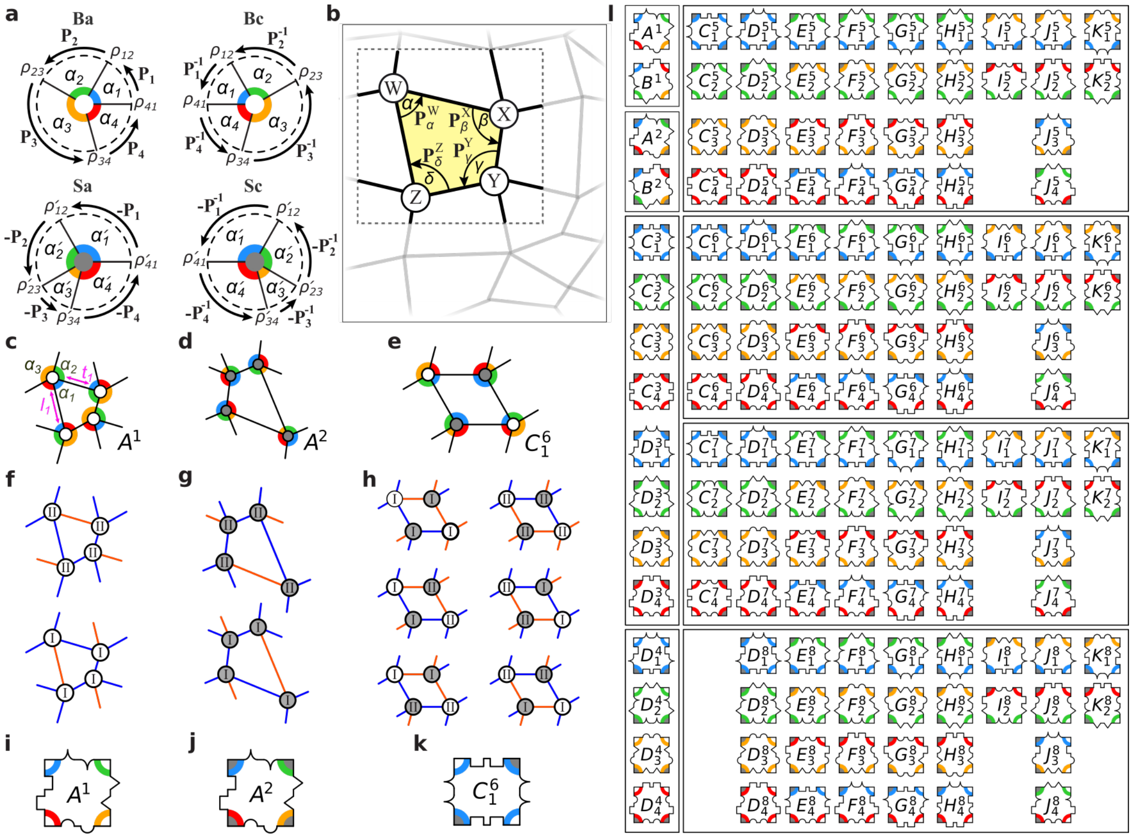

FIG. 1. **Rigidly foldable tiles.** **a,** Group of symmetry-related 4-vertices; color indicates sector angle, and center grey circle indicates supplementation. **Ba**: anti-clockwise ordered base vertex with sector angles α_j, fold angles $\rho_{j,j+1}$ and folding operators P_j. **Bc**: clockwise ordered base vertex, with inverse fold operators. **Sa**: supplemented vertex with anticlockwise ordered sector angles $\alpha_j' := \pi - \alpha_j$ and negated fold operators. **Sc**: supplemented-clockwise vertex, with inverse negated operators. **b,** A quadrilateral crease pattern (grey lines) composed of 4-vertices (intersections) where four creases meet. A Kokotsakis mesh (black lines) consists of four vertices grouped around a central quadrilateral plate (yellow). In order for such a mesh to fold, Eqs. 1 & 2 must be satisfied. **c,** Example crease pattern for tile A^1 with choice of sector angles $\{\alpha_j\} = \{60°, 90°, 135°, 75°\}$; note that a different choice of sector angles results in this same tile so long as the arrangement of the generating vertices around the central plate is the same. This leaves any given tile with three sector angles $(\alpha_1, \alpha_2, \alpha_3)$ and two crease lengths (l_1 and t_1) that can be adjusted. **d,** Example crease pattern for tile A^2. **e,** Example crease pattern for tile C_1^6. **f-h,** The folding branches of A^1, A^2 and C_1^6, with the folding branch (I or II) of each vertex indicated. Red (blue) lines correspond to mountain (valley) folds (with the convention that the left-top vertex has one mountain fold). **i-k,** Jigsaw puzzle piece representation. The colored rings correspond to the sector angles on the central plate, intruding/extruding notches encode the orientation of each corner vertex (cusp→plate 1 to 2, triangle→2 to 3, semicircle→3 to 4, square→4 to 1), and grey colouring indicates supplementation. **l,** All 140 compatible tiles, grouped by their supplementation patterns 1-8.

are connected by simple inverse and minus relations: ${}^{\mathbf{Bc}}\mathbf{P}_j = {}^{\mathbf{Ba}}\mathbf{P}_j^{-1}$, ${}^{\mathbf{Sa}}\mathbf{P}_j = -{}^{\mathbf{Ba}}\mathbf{P}_j$, and ${}^{\mathbf{Sc}}\mathbf{P}_j = -{}^{\mathbf{Ba}}\mathbf{P}_j^{-1}$. The simple expressions that relate both the operators and the sector angles of **Ba**, **Bc**, **Sa** and **Sc** allow to transform the compatibility conditions into to a fully solvable combinatorial problem (see Methods). Out of the $16^4 = 65536$ possible combinations that can be considered by placing one of these vertices in one of its four orientations at each corner of a mesh, we obtain 140 distinct motifs that are rigidly foldable (see Methods).

All foldable meshes represent unique combinations of the four vertices that we refer to as tiles, with names such as A^1, B^2 and C_1^6 (see following paragraphs for naming conventions). For example, tile A^1 combines four copies of the vertex **Ba**, tile A^2 combines four copies of the vertex **Sa**, and the tile C_1^6 combines vertices **Bc** and **Sa** (Fig. 1c-e). Crucially, all tiles are pluripotent and allow for two, four, or six independent folding branches (Fig. 1f-h—see Methods). This pluripotency arises from the fact that all 4-vertices have *two* independent folding motions[9], which can be accessed self-consistently in multiple ways for each tile. As one can vary the underlying sector angles and the horizontal and vertical spacing between vertices, each tile corresponds to a five degree-of-freedom family of crease patterns (Fig. 1c).

Tiles can be placed adjacent to encode larger foldable

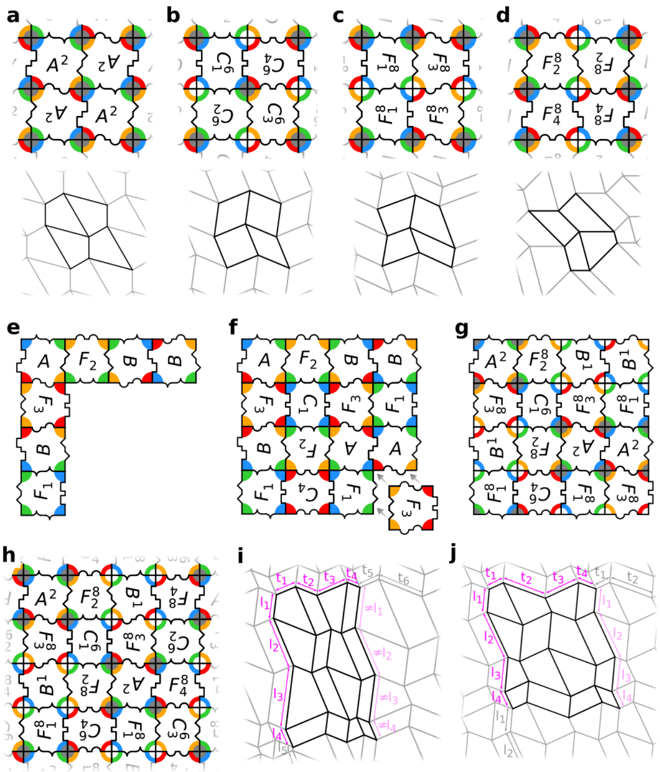

FIG. 2. **Jigsaw origami tilings.** **a,** A-tiling and corresponding Huffman crease pattern[22]. **b,** C-tiling and corresponding Barreto's MARS crease pattern[9,20]. **c-d,** F-tilings and corresponding trapezoidal crease patterns. **e,** A seed column and row of prototiles. **f,** The remaining prototiles in the bulk can be uniquely filled in. **g,** Assigning one of two possible supplementation patterns. **h,** Periodic 4×4 class 1 tiling. **i-j,** By choosing sector angles and adjusting crease lengths (t_k and l_k), crease patterns generated from the tiling in panel **h** can be aperiodic (**i**) or periodic (**j**).

structures if and only if their shared vertices are consistently defined. We implement this condition by representing tiles as jigsaw-shaped 'puzzle pieces', where the orientations and types of constituent vertices are represented by jigsaw edges and colored arcs and circles. Importantly, this representation is independent of the generating angles $\{\alpha_j\}$—the puzzle pieces capture the symmetry relationships, not the particular geometry. Consistency between pieces is then precisely equivalent to demanding that their notches and supplementations match (Fig. 1i-l—see Methods). Based on this intuitive rule one can immediately begin designing foldable crease patterns. Remarkably, many of the most widely-known and intensely studied patterns casually emerge. For example, fitting A tiles creates the Huffman crease pattern[22], combining C tiles yields a generalization of the ubiquitous Miura-ori[3] known as Barreto's Mars pattern[20], and combining F tiles yields lesser known trapezoidal patterns (Fig. 2a-d). Such crease patterns are also inherently pluripotent, and the multiplicity of their branches is exactly countable (see Table I for expressions and the Supplementary Information for exact counting arguments). For example, the Huffman pattern features two folding branches regardless of its $m \times n$ size[21], whereas the number of branches in the Mars pattern grows as $2^{m+1} + 2^{n+1} - 2$. In Supplementary Movies 1 & 2, we physically fold all 14 branches of a 3D printed, 3×3 tiling (additional details in Supplementary Information). We note that our branch counting argument requires the base vertex to not be collinear; in particular, counting the branches of the Miura-ori pattern, where all vertices are collinear, is a much more complex problem[16].

Beyond these elementary examples, we have devised a procedure to systematically create all possible $m \times n$ tilings. This begins by representing each tile as a combination of one of 34 'prototiles' ($A - K_2$), which encode the orientation and handedness of each vertex, and one of 8 supplementation patterns that encode the supplementation. This representation underlies the super/subscripts of our tiles; for example C_1^6 combines prototile C_1 and supplementation pattern 6. We group prototilings into four classes $(1 - 4)$, such that within each class, three prototiles placed into an L-shape admit a unique fourth fitting prototile. We refer to this as the *triplet completion rule*, which provides the key simplification in the construction and enumeration of prototilings. In each class, at least one prototile out of a subset of 'necessary' prototiles has to be present; in addition, some classes contain a group of 'optional' prototiles, which may or may not be present. Each prototile is a necessary prototile in precisely one class (Table I).

The possible number of prototilings in each class follow from the combinatorics of planting at most one 'seed' row and one 'seed' column, since these uniquely define the bulk due to triplet completion. For class 1, the seed row and column can be chosen independently; for class 2, only one periodic prototiling exists (up to permutations); for class 3 and 4, either a seed row or seed column can be freely chosen. Working out the combinatorics, we obtain exact expressions for the number of $m \times n$ prototilings, N_p (Table. I; note class 4 counting is slightly more complicated and depends on the even/oddness of the number of rows/columns, as we explain in the Supplementary Information). To extend this counting to tilings, we enumerate the number of allowed supplementation patterns, N_s, that are compatible with each prototiling. Our exact counting argument for the number of tilings is applicable when the base vertex is not collinear, not flat-foldable, and does not have an equal pair of opposing angles (see Supplementary Information). We find that in all classes the number of tilings, given by the product of N_p and N_s, grows exponentially with (linear) system size (Table I), and in total already exceeds 2 million for all 4×4 tilings.

We illustrate our design procedure for class 1 crease patterns, which are based on A, B, C and F tiles. First, we design one seed column and one seed row of prototiles (Fig. 2e); second we fill in the bulk (Fig. 2f); third, we assign a compatible supplementation pattern (Fig. 2g). The seed column and row determine whether these tilings are periodic or aperiodic (Fig. 2g-h). To translate tilings to crease patterns, we pick one set of generating sector angles, and then the linear dimensions of each seed row and column. Each $m \times n$ tiling therefore corresponds to a family of crease patterns with $(m+n+3)$ independently tunable parameters. Choosing generic crease dimensions for a periodic tiling will yield an aperiodic crease pattern (Fig. 2i), but optimized dimensions can yield periodic crease patterns (Fig. 2j). This illustrates that first designing tilings and then choosing sector angles and linear dimensions opens up a vast design space.

To illustrate the combinatorics that underlies the design of class 1 tilings, we choose one of two possible supplementation patterns by focussing on A^2 tiles and assume without loss of generality that these are rotated upright or upside-down. We then summarize the potential juxtapositions of class 1 tiles in adjacency diagrams (Fig. 3a). These adjacencies stipulate that: *(i)* A and B strips need to be separated by an F patch of odd length; *(ii)* An F strip of even length separates either two A, or two B strips. Using these rules, we can systematically construct 2^{m+1} seed rows of length m and 2^{n+1} seed columns of length n, where each row and column either have one or more necessary A or B tiles (and no C tiles), or have C tiles (but not A or B tiles); these seed columns and rows span the full space of class 1 tilings. (Similar adjacency diagrams can be constructed for the other classes and underly our counting procedures; see Methods and Supplementary Information.)

Naturally, the choice of seed rows and columns must also determine the ultimate folded shape(s). To demonstrate how, we consider an 11-tile strip consisting of patches of A, B and F tiles, and vertically extend this to an 11×6 tiling (Fig. 3b). The corresponding crease pattern admits two folding branches (Fig. 3c-d). In the first, the horizontal folds are all valleys while the vertical folds alternate between mountain and valley, leading to a cylindrical folded shape (Fig. 3e). In the second, the horizontal folds alternate, while the vertical folds are all valleys in A patches, mountains in B patches, and alternating mountains and valleys in F patches, leading to a folded state that, when viewed edge on, juxtaposes negative, positive and zero curvature in the A, B and F patches, respectively (Fig. 3f). This illustrates that the sign of the curvature in the folded shapes is encoded by the seed rows and columns.

We can harness this link between tile choice and curvature to design a single crease patterns that folds into two target shapes. To demonstrate this for two shapes involving markedly distinct curvatures, we choose the Greek letters α and ω. Working once more with class 1 tiles, we embed the curvatures for these symbols along the left-most seed column and upper seed row of a 36x36 tiling (Fig. 3g-h). By adjusting the crease lengths to modify the magnitude of the local curvature and avoid intersections in the flat crease pattern, we constructed a crease pattern whose two folded states closely approximate our desired shapes (Fig. 3i-j; to see this simulated pattern rigidly folding between the two shapes, see Supplementary Movie 3, and for more details see Methods). We note that here we use a base vertex that is flat foldable, as we observed that crease patterns based on generic vertices form 3D folded shapes which can exhibit 'torsion' when viewed edge on. To realize a physical manifestation, we laser-score two 50 cm x 60 cm x 0.20 mm Mylar sheets with this same pattern and manually fold them, pinching each crease according to the corresponding MV designation. The folded specimens closely match the simulated shapes, fleshing out the notion of tiling-based design of pluripotent origami (Fig. 3k-l).

Exploiting symmetries to ensure foldability, our approach can be easily implemented by scientists, engineers and designers alike, with potential applications in mechanical metamaterials, robotics, and deployable structures. Particularly for the case of metamaterials, our work opens the door to creating systems that are far more complex than standard patterns such as the Miura-ori, involving for example multi-vertex unit cells or globally disordered patterns. Our method encompasses restricted tiling problems for (existing) crease patterns with additional relations between the sector angles of the base vertex, such as flat foldability[3,18,29]. While we have presented an intuitive design approach, it is possible a more rigorous computational framework could be incorporated to design complex target shapes[30]. Our results provide starting point for constructing and designing non-rigid and multi-stable folding structures[8,10].

METHODS

4-Vertices

A 4-vertex consists of four rigid plates connected by four flexible hinges that meet in a central point. In all cases, we consider sector angles that add to 2π, with each angle $< \pi$. In origami, one often encounters non-generic vertices, where the sector angles are related in some manner[18]. In our work we need to consider three types of such non-generic vertices: collinear vertices, for which $\alpha_j + \alpha_{j+1} = \pi$ for at least two values of j, flat-foldable vertices for which $\alpha_j + \alpha_{j+2} = \pi$ for all j, and vertices that have an opposing pair of sector angles equal ($\alpha_j = \alpha_{j+2}$ for a least one value of j). We note in passing that the Miuri-ori fold is both collinear and flat-foldable, and thus highly non-generic.

All our results are immediately applicable to generic vertices, that we define as not collinear, not flat-foldable,

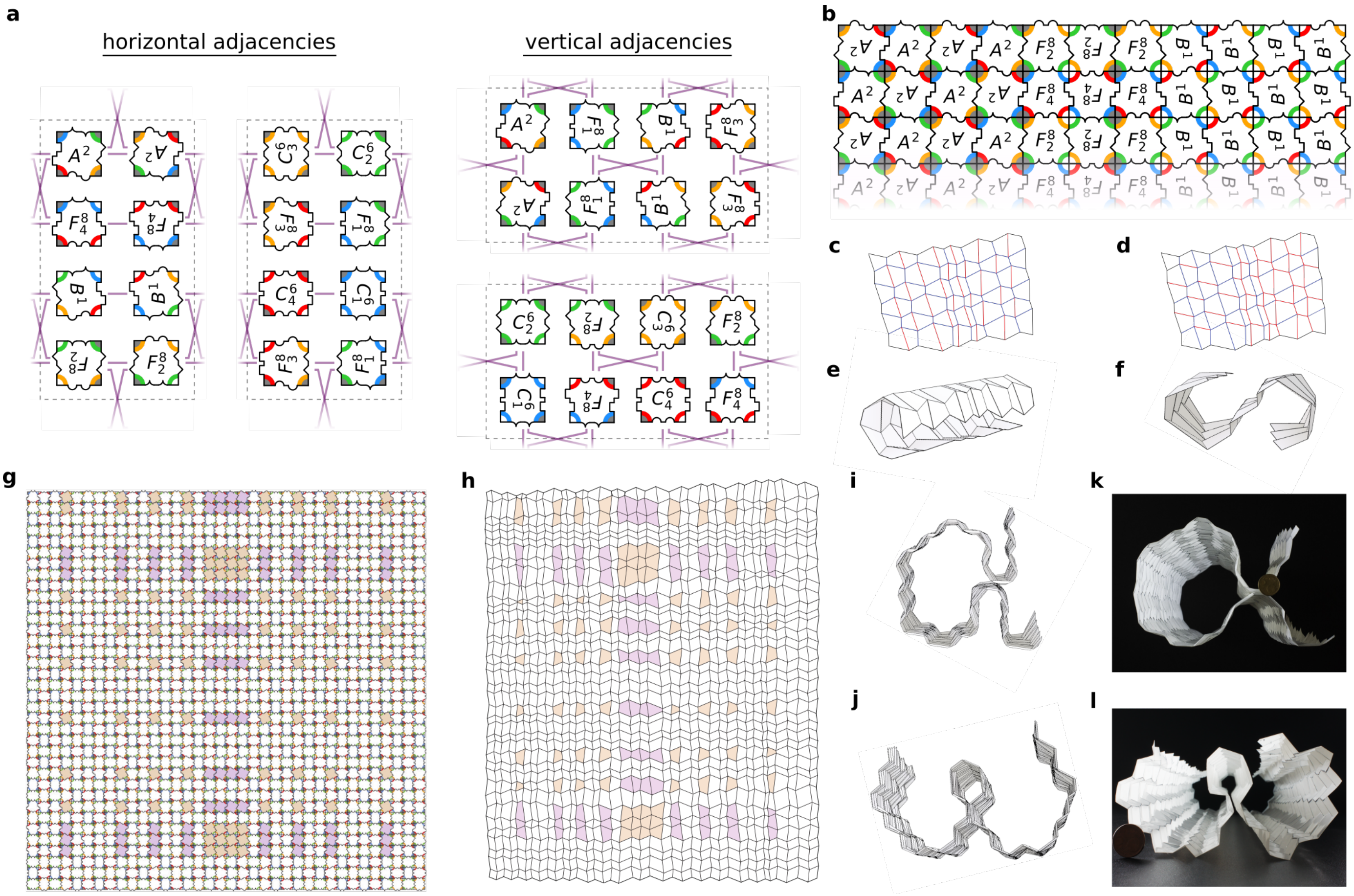

FIG. 3. **Rational design of pluripotent crease patterns.** **a** Adjacencies in class 1 patterns. The purple lines connect pairs of tiles that can be joined along the indicated edge; the grey box indicates a periodic boundary. **b** A 1×11 row combining A, B and F-strips forms a tiling when extended with alternating periodic rows. **c,d** Corresponding crease pattern using a base vertex of $\alpha_j = 60^\circ, 75^\circ, 120^\circ, 105^\circ$ and folding branches with mountains (valleys) colored red (blue). (Note that the special choice of a flat-foldable sector base vertex does not affect the branching or folding behavior relevant to the construction of the tilings/crease patterns shown in this figure—see Supplementary Information for a detailed discussion of this point.) **e,f** Folded states according to the branches shown in (b-c). **g,h** Independent design of a row and column leads to a 36×36 tiling (37×37 vertices) that is programmed to fold into either (**i**) the letter α, or (**j**) the letter ω. The purple/beige colouring indicates A and B tiles, which are important for the curvature. **k,l** Mylar sheet, laser scored with the crease pattern shown in **h** and folded into 3D shapes—the 5 euro cent coin (diameter 21.3 mm) indicates the scale. The rigid-folding simulations and images for panels **e,f,i,j** were produced with the Rigid Origami Simulator by T. Tachi[23].

and without pairs of equal opposing sector angles. However, many of our results require weaker restrictions. In particular, our results for the number of folding branches only requires the vertices not to be collinear; our tiling creation remains valid for non-generic tilings, but the counting of the number of tilings in a class may be affected by all non-genericities. For details, see the Supplementary Information.

The folded configurations are characterized by the folding angles $\rho_{j,j+1}$, defined as the deviation from in-plane alignment between adjacent plates j and $j+1$ (modulo 4). A folded generic 4-vertex always has one fold whose folding angle is opposite in sign from the others[9,18,22]. We call the two folds that are *capable* of having the opposite sign *unique folds*, and these folds straddle a common unique plate[9,18]. Without loss of generality we define our sector angles such that ρ_{41} and ρ_{12} are the unique folds, with plate 1 the unique plate. This is equivalent to demanding that[9,18]:

$$\alpha_1 + \alpha_2 < \alpha_3 + \alpha_4 \ , \tag{3}$$

$$\alpha_4 + \alpha_1 < \alpha_2 + \alpha_3 \ . \tag{4}$$

Vertices can be flipped 'upside-down', and to break this symmetry we assume for the base vertex (without loss of generality)[18]:

$$\alpha_2 > \alpha_4 \ . \tag{5}$$

The two branches, together with the $\{\rho_{j,j+1}\} \leftrightarrow$

$\{-\rho_{j,j+1}\}$ symmetry, yield four distinct mountain-valley patterns for vertex **Ba** (Suppl. Fig. 1a). We denote the folding branches where ρ_{41} or ρ_{12} has the opposite sign as branches I and II, respectively. For the supplemented vertices **Sa** and **Sc**, for which we have $\alpha'_j = \pi - \alpha_j$, inequalities equivalent to Eqs. (3-4) specify that α_3 is the unique plate (Suppl. Fig. 1b)[9].

Folding operators: On a given branch, 4-vertices have one continuous degree of freedom, and it is therefore possible to determine operators (functions) that map the value of any given fold to any other. For the vertex **Ba**, we define the folding operators, ${}^{\mathbf{Ba}}\mathbf{P}_j^{\mathrm{I,II}}$, which map the fold angles adjacent to plate j in an anticlockwise manner: ${}^{\mathbf{Ba}}\mathbf{P}_j^{\mathrm{I,II}}(\rho_{j-1,j}) = \rho_{j,j+1}$ (Suppl. Fig. 1c). The superscripts I and II reflect the fact that these operators are different for each branch of the vertex, but to avoid clutter we suppress these when possible. Explicit expressions for these operators can be readily derived, which shows that they are bijective and anti-symmetric: $\mathbf{P}_j(-\rho) = -\mathbf{P}_j(\rho)$ (see Supplementary Information). As the mirrored vertex **Bc** interchanges clockwise and anticlockwise orientations, it follows that ${}^{\mathbf{Bc}}\mathbf{P}_j = {}^{\mathbf{Ba}}\mathbf{P}_j^{-1}$. In addition, it can be shown that ${}^{\mathbf{Sa}}\boldsymbol{\rho}_j = -{}^{\mathbf{Ba}}\boldsymbol{\rho}_j$, and ${}^{\mathbf{Sc}}\boldsymbol{\rho}_j = {}^{\mathbf{Sa}}\boldsymbol{\rho}_j^{-1} = -{}^{\mathbf{Ba}}\boldsymbol{\rho}_j^{-1}$ (Suppl. Fig. 1c). These expressions can be derived from the explicit expressions for the folding operators, and have an elegant interpretation in terms of spherical mechanisms (see Supplementary Information).

Tiles

Constructing Tiles: To obtain foldable multi-vertex structures, we imagine placing one of the vertices, **Ba**, **Bc**, **Sa**, or **Sc**, in one of its four orientations, at each corner W-Z of a quadrilateral (Suppl. Fig. 1d), resulting in 65536 candidate meshes. The candidate meshes are only foldable when they satisfy two compatibility conditions. First, the angles around the central plate must obey the 'sum condition':

$$\alpha^{\mathrm{W}} + \beta^{\mathrm{X}} + \gamma^{\mathrm{Y}} + \delta^{\mathrm{Z}} = 2\pi \ . \tag{6}$$

Second, the sequential execution of operators around the central plate must yield the identity operation. This leads to the non-linear 'loop condition' expressed as an identity of the 'operator quad':

$$\mathbf{P}_\delta^{\mathrm{Z}} \cdot \mathbf{P}_\gamma^{\mathrm{Y}} \cdot \mathbf{P}_\beta^{\mathrm{X}} \cdot \mathbf{P}_\alpha^{\mathrm{W}} = \mathbf{I}, \tag{7}$$

By checking these compatibility conditions we find that 544 of the 65536 candidates rigidly fold. Several of these are related by tile rotations, leaving the 140 distinct tiles corresponding to Fig. 1 of the main paper. In Suppl. Fig. 2, we show real space versions of these tiles for a particular choice of generating sector angles.

Representation: We represent foldable Kokotsakis meshes, regardless of their generating angles $\{\alpha_i\}$, with puzzle pieces that encode all relevant information *independent* of the generating angles. White circles indicate unsupplemented vertices (**Ba** & **Bc**), grey circles indicate supplemented vertices (**Sa** & **Sc**). Notches pointing from plate j to $j+1$ distinguish between anti-clockwise ordered vertices (**Ba** & **Sa**) and clockwise vertices (**Bc** & **Sc**). The shape of the notches encodes the corresponding fold angles: cusp$\rightarrow\rho_{12}$, triangle$\rightarrow\rho_{23}$, semicircle$\rightarrow\rho_{34}$, square$\rightarrow\rho_{41}$. The coloured rings, which correspond to the sector angles on the central plate, also encode orientation and aid visualization. The combination of white/grey circles and notches completely specifies each of the 65536 candidate meshes. Finally, we note that tiles that have a geometric fit and matching grey/white colouring consistently define their shared vertices, which turns crease pattern design into solving a tiling problem.

Prototiles: Here we explain the rationale behind the prototile representation. First, the only combinations of generic sector angles $\{\alpha_j\}$ that add up to 2π (*i.e.*, satisfy the sum rule) are permutations of $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$, $\{\alpha'_1, \alpha'_2, \alpha'_3, \alpha'_4\}$, and $\{\alpha_j, \alpha'_j, \alpha_k, \alpha'_k\}$, where $j = k$ is allowed. In all such combinations, the number of supplementations per tile is even. However, an even number of supplementations does not change the operator quad, as $(-1)^2 = 1$ and all operators are anti-symmetric. Therefore one can determine all compatible tiles by first establishing all 'primitive' operator quads which combine $\boldsymbol{\rho}$ and $\boldsymbol{\rho}^{-1}$ to satisfy the loop condition, and then applying appropriate pairs of permitted supplementations. The prototile representation therefore delineates between vertex type and orientation with supplementation (Suppl. Fig. 3). We find that there are 34 of these prototile combinations of primitive operators $\mathbf{P}$ and $\mathbf{P}^{-1}$ that (up to cyclic permutations) satisfy the loop rule. These can be further organized into 11 groups (responsible for the *A*-*K* lettering and subscripts) as determined by the form of the operator quads (Suppl. Table 1; Suppl. Fig. 3a).

Each prototile allows two, four or six supplementation patterns, labeled 1–8 (and responsible for superscripts in our names—see Suppl. Fig. 3b). The number of prototiles per group, n_p, the number of supplementation patterns per prototile, n_s, the number of tiles per group, n_t, and the number of operator quads per group, n_q, as well as the allowed supplementation patterns within each group are summarized in Table I.

Folding branches:. Each tile can be folded along multiple branches. Explicitly denoting the branch for the folding operators, we can determine the number of branches for a given tile by counting the number of combinations of I and II labels in the operator quads that lead to identities. All tiles allow for at least two folding branches, where all vertices are on either branch I or branch II, but some tiles further allow for four or six branches. We explicitly present the possible branches for all tiles in the Supplementary Information.

Further considering the patterns that emerge in counting tile branches, we see that the numbers two, four, or six arise because we can organize all operator quads into three groups. First, for tiles $\{A, B\}$, all operators need to be on the same folding branch, yielding two different folding branches. Second, the operator quad of the C_k-prototiles, $\boldsymbol{\rho}_k{}^{\mathrm{I,II}} \cdot (\boldsymbol{\rho}_k{}^{\mathrm{I,II}})^{-1} \cdot \boldsymbol{\rho}_k{}^{\mathrm{I,II}} \cdot (\boldsymbol{\rho}_k{}^{\mathrm{I,II}})^{-1}$, yields identity when adjacent pairs of operators are on the same branch (*e.g.*, {I,I,II,II}, {II,I,I,II}, {I,I,I,I}, *etc.*, yielding six branches). Third, all other tiles contain pairs of distinct operators, and as both pairs need to be on the same branch, this yields four folding branches (Suppl. Fig. 3).

Solving of combinatorial problems

Using the classification, supplementation patterns, and folding branches of individual (proto)tiles, we have determined the number of $m \times n$ prototilings, and for each, the number of supplementation patterns and folding branches, by solving a slew of combinatorial tiling problems. A full explanation of our approach can be found in the Supplementary Information, but here we lay out the main steps of our arguments.

Tiling Classes: We have found that it is possible to group prototiles into four classes based on considering whether an L-shaped triplet of prototiles admits a fourth compatible prototile (Suppl. Fig. 4a). The fourth prototile consists of four vertices, three of which are directly specified by the prototile triplet, and the fourth of which follows (up to supplementation) from the sum rule and observation that each compatible tile has an even number of clockwise ordered vertices. If the fourth prototile does not occur within the set of 34 compatible prototiles, the triplet of prototiles are not in the same class (Suppl. Fig. 4b). If the fourth prototile is one of the 34 compatible prototiles, all four prototiles are in the same class (Suppl. Fig. 4c).

We note that as the number of supplementations per tile is even, the supplementation patterns also satisfy triplet completion, and consequently, tiles can be grouped in four classes and also satisfy triplet completion. Finally we note that we put C tiles in a separate class 2, even though these are optional tiles in all other classes. Similarly, D and J tiles (class 4) appear as optional tiles in class 3. The reason for this is that we recognize that the counting of the number of supplementation patterns or branches is easier within the four classes we define.

Counting prototilings: To calculate the number of $m \times n$ prototilings, N_p, in each class, we determine which pairs of necessary and optional prototiles can be fitted together, leading to 'connection numbers' for each edge of each prototile. In all classes, we find that the numbers of potential fits at the north/south side and east/west sides of each prototile are equal, which means that these connection numbers are preserved in rows and columns. The combination of triplet completion and the conservation of connection numbers greatly simplifies the explicit construction of all $m \times n$ prototilings: triplet completion implies that we, at most, need to construct one seed row and one seed column to uniquely determine each prototiling, and the conservation of connection number facilitates the construction of these rows and columns. The only remaining problem is the placement of necessary and optional prototiles, but this turns out to be solvable (in different ways) for each class, once one realizes that the orientation of prototiles (*i.e.*, 90° rotations) needs to be considered. Summing over all allowed placements and orientations and then constructing the number of corresponding edge columns and rows allows us to obtain exact expressions for N_p in each class. In the Supplementary Information, we derive all expressions in full detail.

Counting supplementation patterns: To calculate N_s, the number of supplementation patterns for $m \times n$ prototilings in each class, we focus on vertices rather than tiles. This transforms the counting of supplementation patterns to a relatively simple two-colouring problem, where we designate each vertex as supplemented or not, subject to the constraints set by the allowed supplementation patterns of each prototile (Suppl. Fig. 3). We find that class 1 tilings admit two supplementation patterns, irrespective of size, while for tilings in the other classes the number of supplementation patterns grows exponentially with system size.

Counting tilings: The results for the number of prototilings, N_p, and the number of allowed supplementation patterns, N_s, can be multiplied to obtain the number of possible $m \times n$ tilings in each class. As shown in Table 1 of the main paper, this leads to complicated yet tractable expressions. An important byproduct of solving the technical details of the counting is obtaining practical procedures to explicitly construct tilings in each class. We demonstrate this below in the context of class 1 patterns.

Counting the number of branches: To calculate N_b, the number of folding branches for $m \times n$ tilings in each class, we first note that while the MV pattern depends on the supplementation, the assignment of branches I or II to each vertex are independent from the supplementation. In fact, vertices can be 'coloured' as branch I or II in an analogous way to supplementation. Hence, counting branches is again a two-colouring problem, where we now assign each vertex in a prototiling to be on branch I or II, subjected to constraints set by the allowed branch patterns of each prototile (Suppl. Fig. 3). We find that class 1 tilings admit two branches, irrespective of size, while for tilings in the other classes, the number of branches grows exponentially with system size (For exact expressions, see Table 1 of the main text; for exact counting see the Supplementary Information.)

Design of class 1 crease patterns

In the main text we consider a tiling specified by a row consisting of 3×1 A, B and F patches, and columns consisting of A, B or F tiles only (Fig. 3 of the main text). Here we consider a more complex tiling, where the top row and left column consist of 3×1, respectively 1×3, strip of A, B and F tiles, leading to a tiling consisting of 3×3 patches of A, B, F and C tiles (Suppl. Fig. 5). The tiles in the interior follow from the tiles in the top row and left column; for example, C tiles are located wherever we find F tiles on the top row and left column.

The two folding branches of class 1 patterns follow from fixing all vertices on branch I or branch II (see Supplementary Information). To determine the corresponding mountain-valley (MV) patterns, we first summarize the MV patterns for each tile, and from this determine the MV patterns for each of the respective A, B, F and C patches (Suppl. Fig. 6a-c). Combining these, we can construct the two distinct MV patterns of the complex tiling (Suppl. Fig. 6d-e). For folding branch I, the horizontal folds have a single value along a connected line of folds: valleys for patches 1-3, alternating valleys and mountains for patches 4-6, and mountains for patches 7-9 (Suppl. Fig. 6d-e). The vertical folds alternate and do not lead to any overall curvature. Hence, once folded, the top and bottom patches exhibit opposite curvatures, while the middle patch forms a corrugated sheet that is flat at large scales—the precise curvature pattern is thus set by the left column. For folding branch II, the curvature pattern is set by the top row: the vertical folds are valleys in patches 1, 4 and 7, alternating valleys and mountains for patches 2, 5 and 8, and mountains for patches 3, 5 and 9. This illustrates that the patterns of A, F and B tiles in the top row and left column can be used to arbitrarily control two independent curvature patterns for the two folding branches of the corresponding crease patterns.

Rational design: While the sign of the curvature is directly controlled by the tilings, the magnitude can be controlled by the 'concentration' of A and B tiles in a background of F tiles, and by the linear spacing associated with each tile. This allows the straightforward design of crease patterns that fold into complex, targeted curvature patterns. Using these principles, the design of tilings and corresponding crease patterns that fold into two independently defined shapes is straightforward (see Fig. 3 main text).

Owing to our conventions for defining the ordering and orientation of the base vertex **Ba**, the qualitative behaviors described in the previous paragraph are *independent* of any generic choice of generating angles. Nonetheless, there are some practical considerations in choosing these. We use two sets of sector angles $\{\alpha_j\}$ in our crease patterns. Most cases correspond to $\{\alpha_j\} = \{60°, 90°, 135°, 75°\}$ which is a generic vertex (not flat-foldable) following the conventions Eq. (3-5). For the design of the β-crease pattern and the $\alpha-\omega$ crease pattern, we choose $\{\alpha_j\} = \{60°, 105°, 120°, 75°\}$ which is flat-foldable[29] ($\alpha_1 + \alpha_3 = \alpha_2 + \alpha_4$). While our designs also work for generic vertices, self-intersecting ('bowtie') quadrilaterals are harder to avoid there. The linear crease dimensions are chosen such that vertices are reasonably spaced, without self-intersections of creases—for finite patterns, this is always possible. Finally we note that an $m \times n$ tiling determines an $m+1 \times n+1$ vertex pattern. Edges to these patterns are in principle arbitrary; we defined the truncated vertex shapes at the edge by extending the tilings with additional rows and columns.

Sample fabrication

To create the folded specimen in Fig. 3 of the main text, we laser score two 50 cm by 60 cm Mylar sheets with a thickness of 0.2 mm. We program the laser cutter to burn the crease pattern 0.1 mm deep into the sheet, where all cuts are on one side, to make the sheet more easily bent along the locations of the folds. We then draw the crease patterns onto both sheets with permanent marker. After manipulating all folds, we can then fold the two sheets into their final shapes, shown in Fig. 3i-j of the main text. The scored lines on the sheet correspond exactly to the crease pattern in Fig. 3f except with one edge of plates corresponding to the edge of the 'omega' side deleted, resulting in 37×38-plate pattern. This is done in consideration of the fact that edges of the pattern can be extended or retracted with no effect on the foldability and minimal effect on the shape.

Acknowledgments—We thank B. G.-g Chen, C. Coulais, Y. Shokef and P.-R. ten Wolde for fruitful discussions, D. Ursem and R. Struik for technical support, and the Netherlands Organization for Scientific Research for funding through Grants NWO 680-47-609, NWO-680-47-453 and FOM-12CMA02.

Author contributions—M.v.H. conceived of the project. P.D. carried out the experiments. P.D., N.V., S.W. and M.v.H. developed the theoretical framework. P.D., S.W., and M.v.H., wrote the manuscript.

Data availability—The data that support the plots within this paper and other findings of this study are available from the corresponding author upon request.

[1] Felton, S., Tolley, M., Demaine, E., Rus, D., & Wood, R. A method for building self-folding machines. Science **345**, 644-646 (2014).

[2] Miskin, M.Z., Dorsey, K.J., Bircan, B., Han, Y., Muller, D.A., McEuen, P.L., & Cohen, I. Graphene-based bimorphs for micron-sized, autonomous origami machines. Proc. Natl. Acad. Sci. U.S.A. **115**, 466-470 (2018).

[3] Miura, K. Method of packaging and deployment

of large membranes. In Space. Inst. Space Astronaut. Sci. Rep. **618**, 1-9 (1985).

[4] Kuribayashi, K., Tsuchiya, K., You, Z., Tomus, D., Umemoto, M., Ito, T., & Sasaki, M. Self-deployable origami stent grafts as a biomedical application of Ni-rich TiNi shape memory alloy foil. Mat. Sci. Eng A-Struct **419**, 131-137 (2006).

[5] Wilson, L., Pellegrino, S., & Danner, R. Origami inspired concepts for space telescopes. In 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference. (Boston, MA 2015).

[6] Evgueni, T.P., Tachi, T., & Paulino, G.H. Origami tubes assembled into stiff, yet reconfigurable structures and metamaterials. Proc. Natl. Acad. Sci. U.S.A. **112**, 12321-12326 (2015).

[7] Schenk, M. & Guest, S.D. Geometry of Miura-folded metamaterials. Proc. Natl. Acad. Sci. U.S.A. **110**, 3276-3281 (2013).

[8] Silverberg, J.L., Evans, A.A., McLeod, L, & Hayward, R.C. Using origami design principles to fold reprogrammable mechanical metamaterials. Science **345**, 647-650 (2014).

[9] Waitukaitis, S., Menaut, R., Chen, B.G., & van Hecke, M. Origami Multistability: From Single Vertices to Metasheets. Phys. Rev. Lett. **114**, 055503 (2015).

[10] Silverberg, J.L., Na, J.-H., Evans, A.A., Liu, B., Hull, T.C., Santangelo, C.D., Lang, R.J., Hayward, R.C., & Cohen, I. Origami structures with a critical transition to bistability arising from hidden degrees of freedom. Nat. Mater. **14**, 389-393 (2015).

[11] Chen, B.G., Liu, B., Evans, A.A., Paulose, J., Cohen, I., Vitelli, V., & Santangelo, C.D. Topological Mechanics of Origami and Kirigami. Phys. Rev. Lett. **116**, 113501 (2016).

[12] Dudte, L. H., Vouga, E., Tachi, T., & Mahadevan, L. Programming curvature using origami tessellations. Nat. Mater. **15**, 583-588 (2016).

[13] Overvelde, J.T.B., Weaver, J.C., Hoberman, C., & Bertoldi, K. Rational design of reconfigurable prismatic architected materials. Nature **541**, 347-352 (2017).

[14] Bertoldi, K., Vitelli, V., Christensen, J., & van Hecke, M. Flexible mechanical metamaterials. Nat. Rev. Mat. **2**, 17066 (2017).

[15] Lang, R.J. *Origami Design Secrets: Mathematical Methods for an Ancient Art.* Taylor and Francis, 2011, Boca Raton, Florida, second edition (2011).

[16] Ginepro, J., & Hull, T.C. Counting Miura-ori Foldings. J. Int. Seq. **17**, 14108 (2014).

[17] Arkin, E.M., Bender, M.A., Demaine, E.D., Demaine, M.L., Mitchell, J.S.B., Sethia, S., & Skiena, S.S. When can you fold a map? Comp. Geom. **29**, 23-46 (2004).

[18] Waitukaitis, S. & van Hecke, M. Origami building blocks: Generic and special four-vertices. Phys. Rev. E **93**, 023003 (2016).

[19] Chen, B.G. & Santangelo, C.D. Branches of Triangulated Origami Near the Unfolded State. Phys. Rev. X **8**, 011034 (2018).

[20] Barreto, P. T. Lines meeting on a surface: the 'Mars' paperfolding. In Proc. 2nd International Meeting of Origami Science and Scientific Origami 323-331 (ed. Miura, K.) (Sein Univ. Art and Design, Otsu, 1997).

[21] Stachel, H. Flexible polyhedral surfaces with two flat poses. Symmetry **7**, 774-787 (2015).

[22] Huffman, D.A. Curvature and creases: A primer on paper. IEEE Trans. Comp. **C-25**, 1010-1019 (1976).

[23] Tachi, T. Generalization of Rigid Foldable Quadrilateral Mesh Origami. J. Int. Assoc. Shell Spat. Struct. **50**, 173-179 (2009).

[24] Kokotsakis, A. Uber bewegliche Polyeder. Math. Ann. **107**, 627-647 (1933).

[25] Evans, A.A., Silverberg, J.L., & Santangelo, C.D. Lattice mechanics of origami tessellations. Phys. Rev. E **92**, 013205 (2015).

[26] Stern, M., Pinson, M.B., & Murugan, A. The Complexity of Folding Self-Folding Origami. Phys. Rev. X **7**, 041070 (2017).

[27] belcastro, S.-M. & Hull, T.C. Modeling the folding of paper into three dimensions using affine transformations. Lin. Alg. Apps. **348**, 273-282 (2002).

[28] Izmestiev, I. Classification of Flexible Kokotsakis Polyhedra with Quadrangular Base. Int. Math. Res. Notices **3**, 715-808 (2017).

[29] Demaine, E., & Orourke, J. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra.* Cambridge University Press, New York, New York (2007).

[30] He, Z. & Guest, S.D. Approximating a Target Surface with 1-DOF Rigid Origami. In Origami 7: Seventh International Meeting of Origami Science, Mathematics, and Education 505-520 (ed. Lang, R.) (Oxford University, Oxford UK 2018).

Class	Necessary Prototiles	Optional Prototiles	# Prototilings (N_p)	# Sup.Patterns (N_s)	# Branches (N_b)
1	$\{A, B\}$	$\{C_k, F_k\}$	$8(2^m-1)(2^n-1)$	2	2
2	$\{C_k\}$	-	8	$2^{m+1}+2^{n+1}-2$	$2^{m+1}+2^{n+1}-2$
$3^\dagger$	$\{E_k, F_k, G_k, H_k, I_k, K_k\}$	$\{C_k, D_k, J_k\}$	$8(8^n-3^n)$	2^{n+1}	2^{n+1}
4-1 (m even)†	$\{D_k, J_k\}$	$\{C_k\}$	$8(3^n+1-2^{n+1})$	2^{n+1}	2^{n+1}
4-1 (m odd)†	$\{D_k, J_k\}$	$\{C_k\}$	$8(3^n+1-2^{n+1})$	2^{n+1}	2^{n+1}
4-2 (m even)†	$\{D_k, J_k\}$	$\{C_k\}$	$16(2^n-1)$	$2^{n+1}-2+2^{\frac{m+2}{2}}$	2^{n+1}
4-3 (m odd)†	$\{D_k, J_k\}$	$\{C_k\}$	$8(2^n-1)$	$2^{n+1}-2+2^{\frac{m+3}{2}}$	2^{n+1}
4-4 (m odd)†	$\{D_k, J_k\}$	$\{C_k\}$	$8(2^n-1)$	$2^{n+1}-2+2^{\frac{m+1}{2}}$	2^{n+1}

TABLE I. **Classification, counting and combinatorics**. Necessary and optional prototiles for each class, as well as the number of $m \times n$ prototilings, N_p, number of associated supplementation patterns, N_s, and number of branches, N_b. †Tilings in classes 3-4 are either row-specified or column specified. Expressions are for row-specified tilings; expressions for column-specified tilings follow by $m \leftrightarrow n$. The subclasses of class 4 depend on the occurrence of J tiles in odd and even columns. For subclass 4-1, J tiles are present in all columns, for the other subclasses, J tiles are present in alternating columns. Subclass 4-2 corresponds to even m, where J prototiles occur in either the leftmost or rightmost column; subclass 4-3 and 4-4 correspond to odd m, with J prototiles occurring in neither the leftmost nor rightmost column (subclass 4-3) or in both these columns (subclass 4-4). For details, see Supplementary Information.