Double-Curved Spin-Valence

Geometric and Computational Basis

Michael Ramirez[1], Emily Baker[2,*], Edmund Harriss[2], Gordana Herning[1]

[1] Massachusetts Institute of Technology, Cambridge, MA, USA

[2] University of Arkansas, Fayetteville, AR, USA

[*] Corresponding author e-mail: erb02@uark.edu

Abstract

This paper describes the geometric and computational basis for creating double-curved space frame configurations from the Spin-Valence deployable kirigami construction system. The goal was to produce space frames that match non-flat target surfaces to be cut from flat parts and deployed into form.

The Spin-Valence system can be geometrically described as two offset surfaces created from tiled units—the primary surface, which can be seen as the original flat sheet of material, and the secondary surface, which emerges from the primary through spin-folds and reconnections between neighbouring unit hubs. The surfaces are offset from each other and connected through triangulating legs inherent in the cut patterns, thus producing a rigid structural configuration.

The construction must take into account the curvature and non-trivial depth of the space frame, as well as the geometric deployment constraints of Spin-Valence units. The frame is created in two stages. From an arbitrary curved target surface, a planar and conical quad mesh is produced, giving the proposed configuration for the more constrained secondary surface. Each unique unit has a specific deployment space, which is applied to the secondary surface tiles in order to compute a primary surface configuration and inherent triangulating legs, completing the space frame geometry.

Keywords: Space Frame, Kirigami, Origami, Structure, Digital Fabrication, Computational Design, Deployable Structure, Discrete Differential Geometry, Python, Kangaroo

1 Introduction

The Spin-Valence kirigami space frame system is, at its core, exceedingly simple. Roughly L-shaped cuts are arrayed around the edges of a polygon to create the geometry of a unit (fig. 1). This geometry, when cut into a sheet of material, allows a spin-fold which extends a central hub out of the original sheet of material, similar to the RES system defined by Miyamoto (2014), and seen in fig. 2. Unlike RES, Spin-Valence relies on points of reduced cross-sectional area that act like pinned connections, rather than multiple folds to create hinges at each end of unit legs. Neighbouring units are placed onto a sheet of material such that their hubs can touch each other once deployed, as seen in fig. 3.

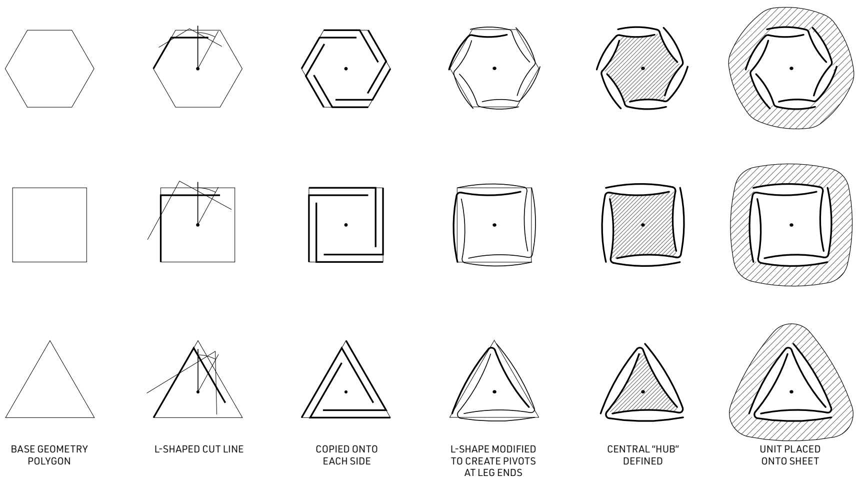

Figure 1: Spin-Valence unit geometry (abstracted pattern on first three columns and actual cut pattern on second three columns).

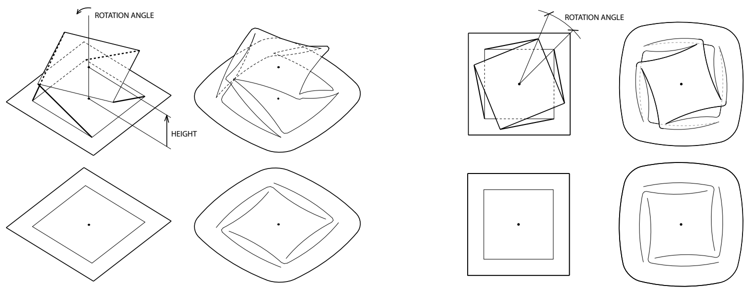

Figure 2: Spin-Valence unit deployment shown as simplified square tile and actual rounded cut pattern designed for steel in (a) axon view and (b) plan view.

This group of connected hubs generates a new surface offset some distance from the original surface and interconnected through triangulating legs, which are inherent in the pattern. The specific relationship between hubs resists their ability to fold back into the original sheet. The original sheet is referred to as the primary surface, and the connected hubs create the offset secondary surface, as described in Baker (2018) and shown in **fig. 3**. Initially designed in cut and folded paper, then explored more broadly in digitally-cut steel, the system aims at quickly deployed structural space frames that, rather than being formed of many parts with complex connections, are made of a single part reconnected to itself. Both the overall form of the space frame and individual connections are encoded in the cut patterns.

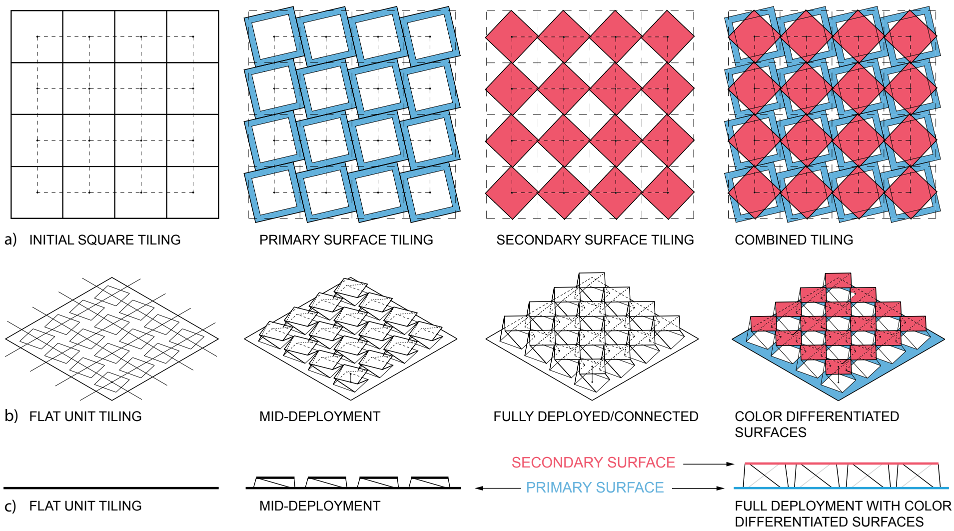

Figure 3: Aggregation of Spin-Valence units, (a) plan view showing base tiling and primary and secondary surface tiling, (b) axon view of unit deployment from a flat sheet into a space frame configuration, (c) section view showing the depth of space frame based on offset of primary and secondary surfaces.

The system was developed through iterative making, where useful relationships were discovered in the physical design space, and the specific cut line geometry of the system was refined for increased strength and constructibility. This is significant, because, as detailed in Baker (2014), the argument is made that the system would never have been conceived if it began with computational design, despite its seeming simplicity. Because of the difficulty of modelling 3D linkages with multiple degrees of freedom like those that occur within each unit in the system, there is greater barrier to creating this system in a computational space than to physically making it. However, it is a system that once developed, tuned to the specifics of fabrication method and material, is then more meaningfully translated into a computationally controllable model. While initially it may appear that making a double-curved version of Spin-Valence would only entail arraying individual units over a curved surface, the real problem is much more complex. For example, the system is superficially very similar in appearance to the tensegrity aggregations shown in Charalambides and Liapi (2005), as they both show double layers of quadrilateral tiling connected by straight legs. However, these systems have very different geometric constraints and obvious differences in structural behaviour. Spin-Valence must: 1) deal with creating double-curved surfaces from flat tessellations, 2) recognize the precise configuration space of the deployment of a single unit, and 3) maintain specific geometric relationships between neighbouring units, while also ensuring that 3D geometry can be nested back into workable 2D cut files for fabrication. The tensegrity systems must deal with the first of these, but they are not constrained by the latter two.

Previously, the Spin-Valence system has been tested as a structural system in Sahuc et al. (2019), though the structural ramifications of curving or bending the system have not yet been explored. A team was assembled to create a pavilion using a double-curved Spin-Valence space frame concept in order to explore both the computational and structural ramifications of this design. This paper will detail the approach to the geometric and computational problem of producing this initial design, show the first physical prototype, and describe future work.

2 Geometric challenge

The Spin-Valence system described above with two offset surfaces of square tiling is shown in flat Euclidean geometry. Adapting it to non-flat target surfaces introduces two problems. The first comes from any change in the shape of the target surface, and the second from the relationship between the two surfaces, which can also be considered as adding depth to the target surface.

The first challenge is described mathematically by Gaussian curvature, which defines the geometry of a surface from the point of view of observers on the surface. A flat sheet can only be bent in one direction and then becomes rigid in the other. Surfaces with non-zero Gaussian curvature are therefore often described as "double curved." (Sullivan 2008; Harriss 2020). From the point of view of the surface, this curvature can only be achieved by a deformation of the shapes involved, giving different lengths and angles to the flat case (as seen in Liapi et al. 2017). This changes the way individual units in the Spin Valence system can connect together. Creating double-curved geometry, such as the target surfaces in fig. 4, at an architectural scale can be accomplished in many ways, such as faceting a surface out of planar parts, as in a soccer ball, using curving seams, such as in clothing patterns, or strategically cutting the surface so that it may be expanded, as in auxetic surfaces. In this case we have chosen a method of creating strips of units that can be cut from flat sheets and assembled to produce double-curved surfaces

— a similar method to that described by Kilian (2003).

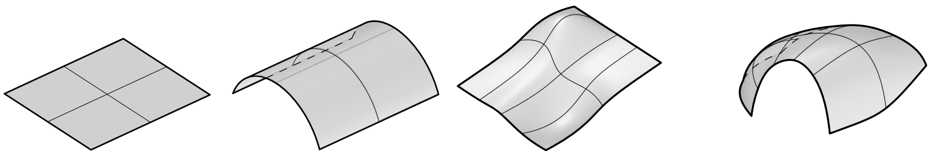

Figure 4: Target surfaces: (a) flat, (b) vault (Gaussian curvature 0), (c) double-curved (Gaussian curvature non-0), and (d) target surface for pavilion design (Gaussian curvature non-0).

The Spin-Valence configuration described in the above figures, with tiled units of equal size and rotation arrayed uniformly on a flat surface, can only be assembled into a flat space frame. Pulling identical units farther apart from each other on the primary surface in one direction can produce single-curved or vaulting space frames that maintain zero Gaussian curvature. The move to double-curved forms could be conceived as distorting the regular tiling pattern, necessitating that each unit be unique.

The second challenge comes from the offset of the primary and secondary surfaces. The geometric properties described above work on mathematical surfaces and so do not consider the material thickness, which adds a distinct geometric challenge discussed, for example in the thin shells (surfaces with non-zero thickness small in comparison to the other dimensions of the structure (Grinspun et al. 2003). For a space frame this creates two notions of material thickness, the thickness of the material used in manufacturing of elements and the thickness, or depth, of the frame structure itself. Both have importance for the structural properties of the resulting frame, but the depth given by the separation of the two surfaces must here be taken into account in computationally deriving the geometry of the space frame. In the completely flat case, all units must be deployed from the primary surface in the same direction, though the system may be deployed to either side of the original sheet. In other words, the units are deployed in the direction of the surface normal, which is constant for the flat case. For a curved surface the surface normal will shift around, creating differences in how individual units must be deployed in order to connect on the secondary surface. Beyond this direction the units will all deploy in the same way, so changing coordinates so that the surface normal is vertical allows for a detailed consideration of the configuration space in **sec. 3.2**.

What this geometric picture conceals, however, is the ability to vary the size and shape of individual units. For curved surfaces, therefore what is required is a system that allows for the changes in geometry within the surfaces and between surfaces.

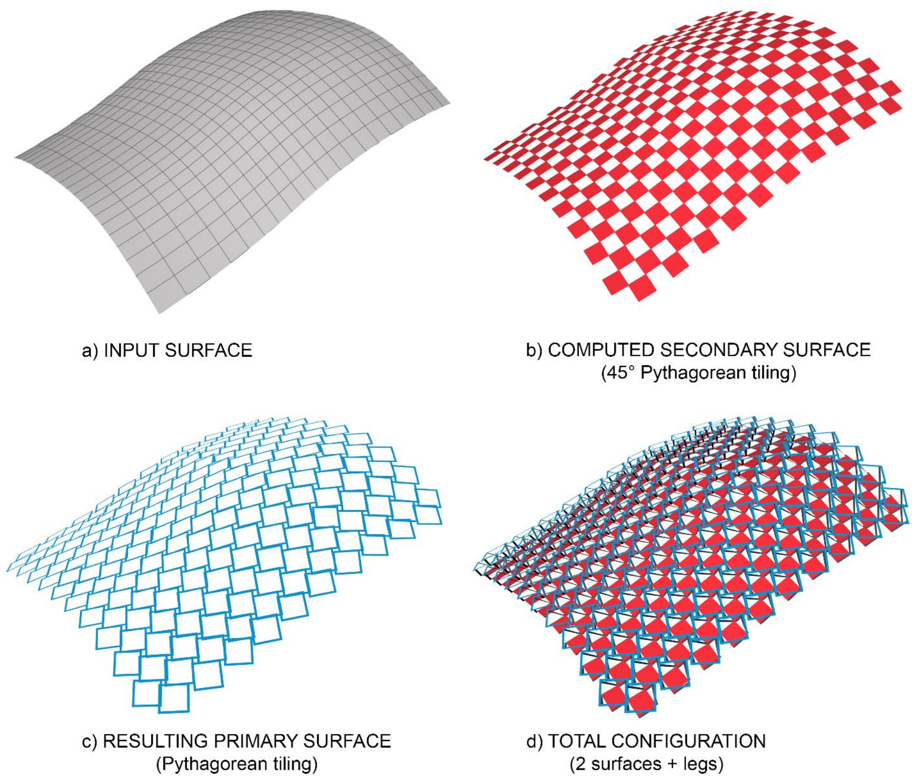

Figure 5: Computational strategy sequence: (a) target surface, (b) secondary surface optimization, (c) primary surface deployed off of secondary surface, and (d) complete space frame configuration.

3 Solution in two parts

Our approach to the problem described above, was to solve each of the two problems separately. We first created a suitable tiling for the unit hubs of the secondary surface, dealing with the problem of Gaussian curvature, and then used it to find an optimal primary surface, working backward to compute the primary surface from the secondary, as described in the set of diagrams in **fig. 5**, from an early iteration of this approach. This approach is mapped out in the flowchart in **fig. 6**.

3.1 Creating a double-curved secondary surface

The problem of creating a collection of flat quadrilateral units close to some given surface, is perhaps the more technical of the two challenges. It is also a classical problem from both CAD and CGI where meshes made of mostly flat quadrilaterals (quad mesh) are preferable to meshes made solely from triangles (for example Pottmann et al. 2007; Crane et al. 2013).

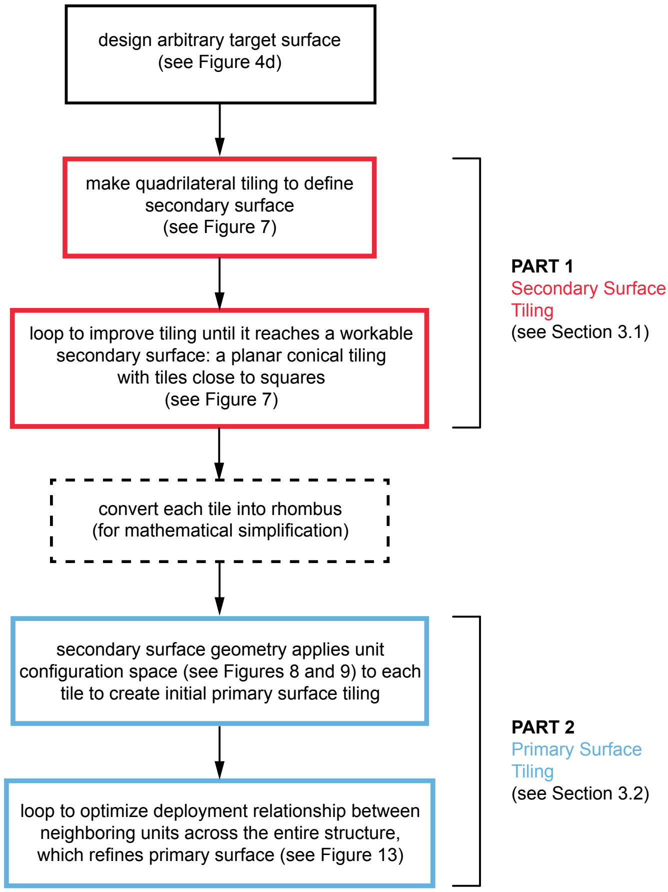

Figure 6: Flowchart describing the process of computing space frame geometry for a given target surface.

Our first requirement was to produce a quad mesh with each face planar and roughly square. This mesh can be considered as a tiling of the surface, producing a proposed secondary surface by twisting each unit in its plane, as described above for the flat Spin-Valence system. To ensure that the resulting pieces would still meet up, additional constraints were placed on the vertices. These constraints are equivalent to the mesh being conical and every face at a vertex being tangent to the same cone, as in Pottmann and Wallner (2008).

The conical mesh has the additional property that the curves on the surface following the edges of the mesh will be "torsion free". In other words, they will not twist around the edge as they move along it. This ensures that the relationship between the primary and secondary surface will be one of purely bending the sheet, without twisting it. This property is useful in many applications. For example, Schling et al. (2018) use similar methods to avoid twisting in a different frame structure.

This method was implemented by numerical optimization using Kangaroo in Grasshopper (Robert McNeel & Associates 2019). The points of the mesh were constrained to stay close to the target surface, and mesh boundary points were further constrained to stay on the boundary curve of the target surface. The individual quads were optimized for planarity and the vertices were optimized to be conical. In addition, bending along edges was discouraged, and for every face the four edges were kept to nearly equal lengths. The initial grid and the resulting optimized grid are shown in **fig. 7**.

Figure 7: Initial grid on surface and optimised candidate for the secondary surface, showing the underlying grid and the secondary surface hubs.

3.2 Creating the primary surface

Finding the optimal positions for the primary surface units based on the previously described optimized tiling configuration at the secondary surface is resolved through first solving the geometry of a single unit, and then performing optimization to minimize distances between deployed primary surface tiles.

Solving the geometry of Spin-Valence

The first step was to create a geometric model of a quadrilateral Spin-Valence unit. The two faces are related by a screw motion, but not all such motions satisfy the constraints given by the lengths of the legs. Solving for the constraints gives a parametrization of the configuration space for a single unit.

For this, the leg thickness (seen in **fig. 11**) was ignored, making the faces for the primary and secondary surface the same size. The two surfaces are joined by linear legs with length equal to adjacent sides as shown in **fig. 8**. This geometric model follows the constrains of the kirigami system by holding leg lengths and surface sizes constant and only varying their positions relative to each other, as a mechanical linkage. There is one configuration where primary and secondary surfaces fall in the same position — the flat or undeployed position. All other possible configurations that would be produced physically by deploying the Spin-Valence unit can be found as a two-parameter configuration space, derived below, using the relationships between the inscribed kirigami constraints and the isometry from the primary to secondary surface tile.

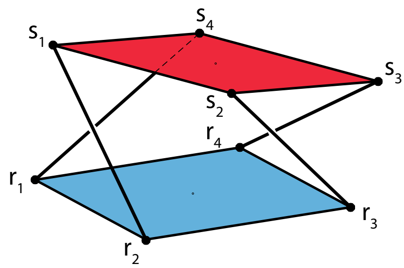

Figure 8: Primary and secondary surface notation.

The initial inputs for the algorithm are the coordinates of the primary surface, denoted as: r_1,r_2,r_3, and r_4 and the isometry R, that represents the deployment of the unit from the primary surface to its position in the secondary surface. Figure 8 illustrates the model. The corners of the deployed unit are therefore, $s_i = R[r_i]$. We will later make an assumption that the primary surface forms a rhombus to obtain the following parametrization:

Theorem. *Every rotation around an axis perpendicular to the diagonal of a rhombic spin-valence unit determines a unique deployment for that unit.*

The proof and explicit parametrization are given below.

The kirigami constraints on this system are the lengths of the four legs; for example, l_1 is the length from r_2 to s_1. This is fixed by eq. (1) and (2) which state that the edge lengths of the units and the distance between the primary surface corner (r_2) and the deploying secondary surface corner (s_1) all keep the same length. Symmetric equations constrain the lengths of the other three legs.

$$(r_1 - r_2).(r_1 - r_2) = (s_1 - s_2).(s_1 - s_2) = l_1 \tag{1}$$

$$(s_1 - r_2).(s_1 - r_2) = l_1 \tag{2}$$

Returning the transformation to eq. (2) gives

$$\begin{aligned}(s_1 - r_2).(s_1 - r_2) = (R[r_1] - r_2).(R[r_1] - r_2) = \\ R[r_1].R[r_1] - 2R[r_1].r_2 + r_2.r_2 = l_1\end{aligned} \tag{3}$$

Plugging (**eq. (1)**) into (**eq. (3)**) gives:

$$\begin{aligned} R[r_1].R[r_1]-2R[r_1].r_2+r_2.r_2 &= (r_1-r_2).(r_1-r_2) \\ R[r_1].R[r_1]-2R[r_1].r_2+r_2.r_2 &= r_1.r_1-2(r_2.r_1)+r_2.r_2 \\ R[r_1].R[r_1]-2R[r_1].r_2 &= r_1.r_1-2(r_2.r_1) \end{aligned} \tag{4}$$

In a similar manner each leg gives a distinct constraint:

$$\begin{aligned} &\text{Leg 1}: R[r_1].R[r_1]-2R[r_1].r_2 = r_1.r_1-2(r_2.r_1) \\ &\text{Leg 2}: R[r_2].R[r_2]-2R[r_2].r_3 = r_2.r_2-2(r_3.r_2) \\ &\text{Leg 3}: R[r_3].R[r_3]-2R[r_3].r_4 = r_3.r_3-2(r_4.r_3) \\ &\text{Leg 4}: R[r_4].R[r_4]-2R[r_4].r_1 = r_4.r_4-2(r_1.r_4) \end{aligned} \tag{5}$$

These equations are significantly simplified by the further assumption that the unit shape is a rhombus, and that its centroid is at the origin. This makes $r_3=-r_1$, $r_4=-r_2$, and $r_1.r_2=0$. The constraints from (**eq. (5)**) thus become:

$$\begin{aligned} &\text{Leg 1}: R[r_1].R[r_1]-2R[r_1].r_2 &&= r_1.r_1 \\ &\text{Leg 2}: R[r_2].R[r_2]+2R[r_2].r_1 &&= r_2.r_2 \\ &\text{Leg 3}: R[-r_1].R[-r_1]+2R[-r_1].r_2 &&= r_1.r_1 \\ &\text{Leg 4}: R[-r_2].R[-r_2]-2R[-r_2].r_1 &&= r_2.r_2 \end{aligned} \tag{6}$$

Now, as $R[x]$ is an isometry, it can be replaced with the function, $R.x+P$, where P is the vector between the centroids of the primary and secondary surfaces and R is a rotation matrix. This is illustrated in **fig. 9**. Moreover, $R[-x]=-R.x+P$, so:

$$\begin{aligned} &\text{Leg 1}: (R.r_1+P).(R.r_1+P) && -2(R.r_1+P).r_2 && = r_1.r_1 \\ &\text{Leg 2}: (R.r_2+P).(R.r_2+P) && +2(R.r_2+P).r_1 && = r_2.r_2 \\ &\text{Leg 3}: (-R.r_1+P).(-R.r_1+P) && +2(-R.r_1+P).r_2 && = r_1.r_1 \\ &\text{Leg 4}: (-R.r_2+P).(-R.r_2+P) && -2(-R.r_2+P).r_1 && = r_1.r_1 \end{aligned} \tag{7}$$

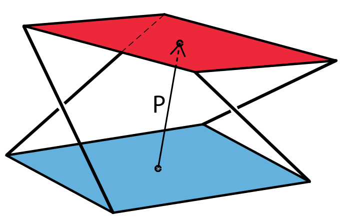

Figure 9: Vector P represents the offset between centroids of the primary and secondary surfaces.

As R is an isometry fixing the origin, it will not affect the distance from the origin, thus, $R.r_1.R.r_1 = r_1.r_1$, and similarly for the other corners. Combining this and simplifying the Leg constraints in (**eq. (7)**), using the sums and differences of Legs 1 and 3, and Legs 2 and 4 respectively, we obtain the following constraints on P and R given r_1 and r_2:

$$\begin{aligned} P.(R.r_1 - r_2) &= 0 \\ P.(R.r_2 + r_1) &= 0 \\ P.P - 2(R.r1.r2) &= 0 \\ P.P + 2(R.r2.r1) &= 0 \end{aligned} \tag{8}$$

The linear transformation, R, can be described using the Rodrigues' Rotation formula (Rodrigues 1840) using the axis of rotation, t, and angle of rotation, θ:

$$R.a = a\cos(\theta) + (t \times a)\sin(\theta) + t(t.a)(1 - \cos(\theta)) \tag{9}$$

Using this formula to parametrize rotations by axis t and angle θ, and again taking sum and difference for the last two constraints of (**eq. (8)**) we obtain the following constraints:

$$\begin{aligned} P.(r_1\cos(\theta) + (t \times r_1)\sin(\theta) + t(t.r_1)(1 - \cos(\theta)) - r_2) &= 0 \\ P.(r_2\cos(\theta) + (t \times r_2)\sin(\theta) + t(t.r_2)(1 - \cos(\theta)) + r_1) &= 0 \\ (\cos(\theta) - 1)\ t.r_1\ t.r_2 &= 0 \\ P.P - 2r_2.(t \times r_1\sin(\theta)) &= 0 \end{aligned} \tag{10}$$

Configuration space of the Spin-Valence unit

Given the constraints above we can now describe all possible configurations for this Spin-Valence unit, and prove the theorem given above. By the third constraint in (**eq. (10)**) one of $(\cos(\theta) - 1)$, $t.r_1$ and $t.r_2$ must be 0, so we can split into three cases, two of which are symmetric other than swapping r_1 and r_2. We will consider $t.r_1 = 0$.

1 ($\theta = 0$): If $\theta = 0$, then the fourth constraint of (**eq. (10)**) would give $P.P = 0$, this is therefore the undeployed case with no translation or rotation.

2 ($t.r_1 = 0$, and $\theta \neq 0$): The constraint that $t.r_1 = 0$ forces t to be orthogonal to r_1. As r_2 is also orthogonal to r_1 and t is a direction (so length does not matter) we can parametrize by ϕ with $t = \sin(\phi)r_2 + \cos(\phi)r_1 \times r_2$. Fixing these values takes the non-linearity out of the equations above, and setting $\sin(\theta) = s$ and $\cos(\theta) = c$ we obtain the system of three linear equations, with three unknowns (the coefficients of P):

$$
\begin{aligned}
P.(cr_1 + s(t \times r_1) - r_2) &= 0 \\
P.(cr_2 + s(t \times r_2) + (1-c)t(t.r_2) + r_1) &= 0 \\
P.P - 2sr_2.(t \times r_1) &= 0
\end{aligned}
\tag{11}
$$

that can be solved to find these unknowns, and thus P uniquely.

For example if P is the zero vector, then $r_2.(t \times r_1) = 0$ which is a triple product so can be rewritten as $t.(r_1 \times r_2) = 0$, so t lies on the plane generated by r_1 and r_2. As for a rhombus these are themselves orthogonal, the unit can deploy without translation by rotating around either diagonal as shown in fig. 10.

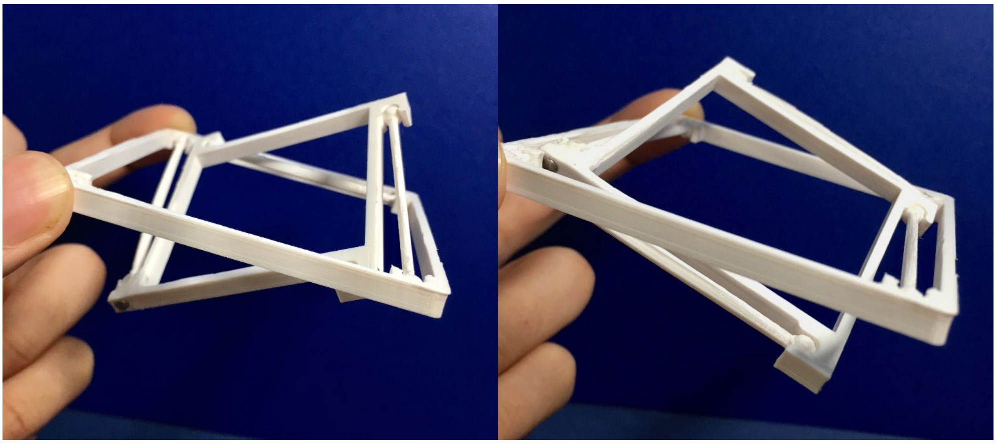

Figure 10: Unit deploying with rotation and no translation tilting in the secondary surface (shown with deployable 3D printed models).

Equating the primary surface geometry

To summarize sec. 3.2, solving the Spin-Valence geometry yields a two-dimensional configuration space, with parameters, θ and ϕ, for each unit; θ controls the rotation motion (vertical, or "up" displacement) of the deployed surface and ϕ dictates the degree of surface non-parallelism, or tilting.

A computationally produced primary surface is generated using Python (Van Rossum and Drake Jr 1995), using the SciPy optimization library (Oliphant 2007). First, the Python script receives the Grasshopper data that describes the secondary surface units. Then, it loops through all of the units and globally deploys their respective primary surfaces according to the equations in sec. 3.2 and each unit's prescribed θ and ϕ parameters. An expansion factor that enlarges the deployed surfaces by a set ratio is included to properly represent the inscribed size differences between the primary and secondary surfaces as seen in fig. 11. This accounts for the actual leg width in the physical construction.

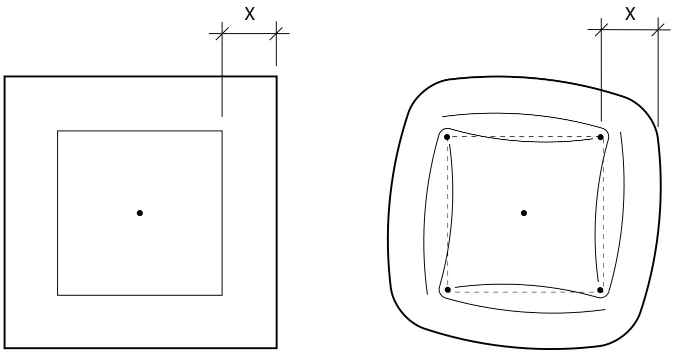

Figure 11: Inscribed size differences between primary and secondary surfaces on simplified and full patterns.

The objective value for optimization is the total summation of distances between adjacent deployed units, which is found by creating an algorithm to locate adjacent sides and calculating the smallest distance between the skew segments. The optimization is conducted by manipulating the θ and ϕ parameters for all units in the quad mesh, using SciPy's Sequential Least Squares Programming, or SLSQP, method. **Figure 12** illustrates the relationship between neighbouring units, and **fig. 13** shows the complete pavilion geometry after primary surface optimization.

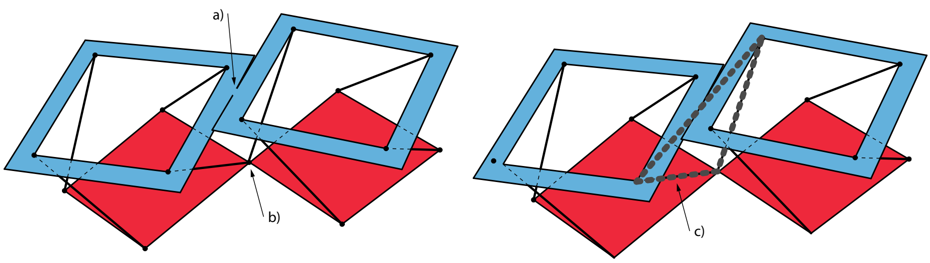

Figure 12: Relationship between neighbouring units, (a) point connection allows for primary surface tiling to be non-coplanar with neighbouring tiles, (b) new connection made between neighbouring secondary surface tiles, and (c) resulting triangular frame that repeats over the system to create a space frame.

4 Constructed prototype and observations

A physical prototype in plasma-cut steel was produced from a 12-unit portion of the pavilion design. The 3D geometry that was generated as described above was flattened and translated from straight line segments into the more nuanced curves that aid steel in folding at only specified locations. Three strips of four units each were drawn, cut out of steel and assembled through folding and welding, as seen in **fig. 14**. This act of deploying a small sample of double-curved Spin-Valence

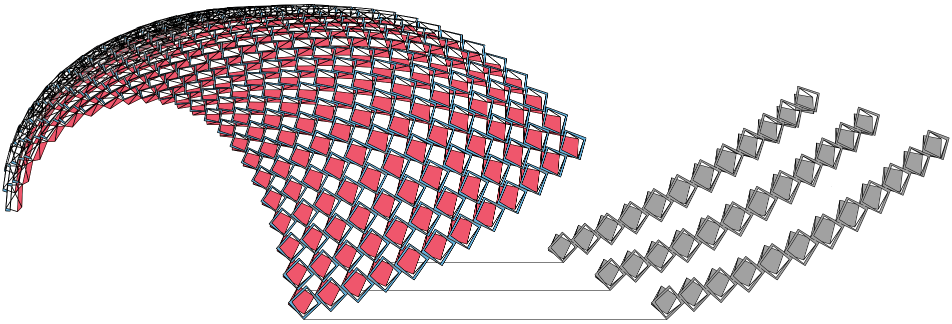

Figure 13: Pavilion design showing computationally formed primary surface after SciPy optimization. Physical assembly would be built up using strips of units.

space frame revealed the potential importance of holding a consistent dimension of overlap between unit hubs at their connections. This consistent dimension could act as a reference to the fabricator and would allow for assembly without the aid of 2D or 3D construction documents for the design, such as plans or digital models. Currently, the computational model does not allow for holding this dimension as a constant, but that is a future goal.

Figure 14: 12-unit prototypes of double-curved space frames in chipboard and steel.

Alternatively, an augmented reality platform such as Fologram could allow for precise deployment of units through projecting 3D data into the physical fabrication space and notifying the fabricator when the part is within tolerances for the design. There is still much to explore in tailoring the double-curved system to the specifics of fabrication.

5 Conclusions and future work

The methods used in this paper describe the optimization of a specific target surface, but they can be used to create Spin-Valence space frames in a wide variety of potential forms. These results open up many possibilities for future work to refine and extend what is presented here.

At the moment, the surface used must be covered by a grid of quadrilaterals meeting four to a corner, as this version is based on the square grid. This can accommodate many target surfaces, but introduces a topological constraint, because such a grid cannot cover a sphere or any other closed surface without a hole. This limitation can be released by allowing three or five quadrilaterals to meet at a small number of vertices, or alternatively, a small number of triangles or pentagons could be introduced within the mesh.

The deployment geometry described above is limited in that each unit is assumed to be a rhombus. We require all of the edges of the secondary surface quad mesh to be close to the same length, which significantly simplifies the geometry. In subsequent iterations, less constrained tile geometry could be used to allow any number of modifications to the system, such as constraining overlaps at hubs to be the same length for ease of fabrication. Constraints like those described in the previous section that lend to greater constructibility need to be addressed in subsequent iterations.

Currently, the system assumes that the individual units in the primary surface will be connected at points (essentially pinned connections on each of the four unit edges as shown in **fig. 12**). This provides freedom in the relationship among primary surface tiles, which makes the primary surface optimization much easier. In moving toward full fabrication of structurally capable space frames, these point connections may need revision for strength. Structural analysis of this configuration will soon reveal more about its behaviour and potential structural optimizations that have geometric bearing.

As seen in **fig. 15**, many other tiling patterns may be employed using the Spin-Valence system. Some target surface forms may lend themselves to better geometric and structural optimizations using alternatives to the square grid shown in this paper.

Another aspect of the system to explore further is the relationship between the primary and secondary surfaces. In addition to the closeness of neighbours considered here, the distance between the two surfaces could be controlled. In fact, the two surfaces could actually pass through each other as shown in **fig. 16**. This would be of importance for general surfaces as the distance between units of the secondary surface is bounded by a multiple of the distance between the same units in the primary surface. For example, if the surface constructed is a sphere, the secondary surface should lie on the inside where distances are shorter. Allowing the secondary surface to pass through the primary would therefore permit continuous surfaces with peaks (where the secondary should be below the primary) and valleys (where the secondary should be above the primary). It is worth noting that we became aware of this possibility thanks to the behaviour of the optimization tools described above, so they were an effective research tool, not just a problem-solving technique.

Figure 15: Examples of Spin-Valence space frames constructed of chip board in alternative tiling patterns.

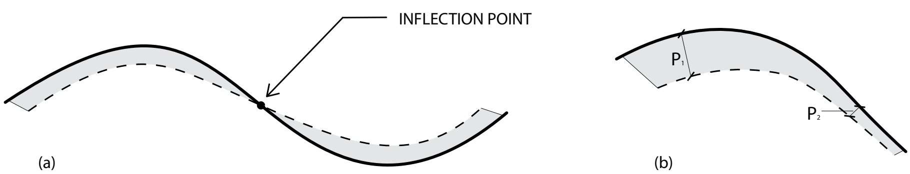

Figure 16: Sectional representation of (a) the secondary surface passing through the primary surface to seamlessly accommodate changes from positive to negative curvature, and b) potential to control the offset between the two surfaces, changing the depth of the space frame.

Acknowledgements

We would like to acknowledge Nick Sherrow-Groves for contributions to the structural aspects of this project, and we would like to thank Mohamed Ismail and Yijiang Huang for their input.

References

Baker, E. (2014). Search for a rooted aesthetic: Study in spin-valence. In F. Gramazio, M. Kohler, and S. Langenberg (Eds.), *Fabricate: Negotiating Design & Making*, pp. 128–135. Zurich, Switzerland: gta Verlag.

Baker, E. (2018). Spin-valence: Kirigami space frame. In *IASS 2018: Creativity in Structural Design annual symposium*. IASS: MIT, Cambridge, MA.

Charalambides, J. E. and K. A. Liapi (2005, September). Implementation of a computer algorithm for an interactive 3d cad generation of tensegrity structures. In *Proceedings of the 22nd International Symposium on Automation and Robotics in Construction*, Ferrara, Italy. International Association for Automation and Robotics in Construction (IAARC).

Crane, K., F. de Goes, M. Desbrun, and P. Schröder (2013). Digital geometry processing with discrete exterior calculus. In *ACM SIGGRAPH 2013 Courses*, SIGGRAPH '13, New York, NY, USA. Association for Computing Machinery.

Grinspun, E., A. N. Hirani, M. Desbrun, and P. Schröder (2003). Discrete shells. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, Goslar, DEU, pp. 62–67. Eurographics Association.

Harriss, E. (2020). Gauss-bonnet sculpting. In C. Yackel, R. Bosch, E. Torrence, and K. Fenyvesi (Eds.), *Proceedings of Bridges 2020: Mathematics, Art, Music, Architecture, Education, Culture*, Phoenix, Arizona, pp. 137–144. Tessellations Publishing.

Kilian, A. (2003, October). Fabrication of partially double-curved surfaces out of flat sheet material through a 3d puzzle approach. In *Connecting Crossroads of Digital Discourse*, Indianapolis (Indiana), pp. 75–83. Annual Conference of the Association for Computer Aided Design In Architecture.

Liapi, K., A. Papantoniou, and C. Nousias (2017). Square tessellation patterns on curved surfaces : In search of a parametric design method. In *Parametric and Generative Design, eCAADe*, pp. 371–378. eCAADe: eCAADe.

Miyamoto, Y. (2014, August). Rotational erection system (res): Origami extended with cuts. In M. K (Ed.), *Origami6 II: Technology Art, Education*, pp. 537–44. AMS.

Oliphant, T. E. (2007, May). Python for scientific computing. *Computing in Science and Engg. 9*(3), 10–20.

Pottmann, H., S. Brell-Cokcan, and J. Wallner (2007). Discrete surfaces for architectural design. In P. Chenin, T. Lyche, and L. L. Schumaker (Eds.), *Curves and Surface Design: Avignon 2006*, pp. 213–234. Nashboro Press.

Pottmann, H. and J. Wallner (2008). The focal geometry of circular and conical meshes. *Adv. Comp. Math 29*, 249–268.

Robert McNeel & Associates (2019). Grasshopper.

Rodrigues, O. (1840). Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace: et de la variation des cordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire. *Journal de Mathématiques Pures et Appliquées 5*, 380–440.

Sahuc, J., G. Herning, and E. Baker (2019). Experimental and analytical investigation of the spin-valence kirigami space frame. In *IASS 2019: Form and Force symposium proceedings. Barcelona, Spain.*

Schling, E., M. Kilian, H. Wang, D. Schikore, and H. Pottmann (2018). Design and construction of curved support structures with repetitive parameters. In L. Hesselgren, A. Kilian, S. Malek, K.-G. Olsson, O. Sorkine-Hornung, and C. Williams (Eds.), *Advances in Architectural Geometry*, pp. 140–165. Klein Publishing Ltd.

Sullivan, J. M. (2008). Curves of finite total curvature. In A. I. Bobenko, J. M. Sullivan, P. Schröder, and G. M. Ziegler (Eds.), *Discrete Differential Geometry*, pp. 137–161. Basel: Birkhäuser Basel.

Van Rossum, G. and F. L. Drake Jr (1995). *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam.