RIGID-FOLDABLE ORIGAMI STRUCTURES: PARAMETRIC MODELLING WITH GRASSHOPPER

Geometric and structural issues

Giulia Curletto

University of Genoa

Genoa, Italy

Giulia.curletto@edu.unige.it

Abstract: Rigid-foldable systems inspired by Origami enable the development of complex structures only governed by few variables. A method to design deployable structures considering geometrical and structural issues is described. Drawing of the geometry and analysis of the structural behaviour are integrated in a continuous process to find the best configuration of deployment. Parametric design is proposed as the ideal powerful method for modelling Origami structures, applicable in both geometrical and analysis processes. The parametric software Grasshopper has been applied to draw the geometry and to control the leading configuration parameters, while FEA software has been used to study the structural behaviour. To connect parametric and structural analysis software, GeometryGym (Grasshopper plug-in) has been investigated as particularly interesting to define the structural features. The purpose of this paper is to provide a guideline to design Origami structures using parametric methodology and to describe drawing and structural analysis steps applied to two specific cases: Waterbomb and Yoshimura Origami.

Keywords*: Origami Modelling; Rigid-Foldable Systems; Parametric Design; Structural Analysis in Grasshopper; Yoshimura and Waterbomb Origami*.

1. Introduction

Many deployable structures, derived by Origami folding principle, are ever more experimented suggesting emergency shelters, building elements and solar sails. Folding is a simple and inexpensive process for transforming matter to fast and easy obtain three-dimensional shapes, and kinetic properties of folds make them extremely versatile suggesting a wide field of application. Despite these properties, the use of deployable structures is limited to prototypes and few products because of modelling and manufacturing issues. The complexity of the geometry and its mechanisms, discussed by some researchers (Tachi et al. 2011, Tachi 2013), has to be integrated with kinematics and structural analysis during the deployment, as presented by Schenk and Guest (2011). Again, problems related to connection by hinges, material, thickness, and forces required to deploy the system have to be considered. In this paper, two-dimensional foldable systems inspired by Origami are investigated, carrying out modelling principles.

2. Origami modelling: parametric process

A correct and detailed modelling is necessary to understand the behaviour of Origami models, and to simulate the folding process and the structural response. Many approaches for modelling and simulating folding motion of Origami have been proposed, starting from Miyazaki et al. (1996), to recent examples, as proposed by Tachi (2009). However, some tools are not suitable for complex Origami, limiting the simulation to models whose folding process is divided into simpler steps. Others are limited to the drawing and simulation of Origami, without considering technological and mechanical issues. Parametric methodology seems to be the ideal instrument, in which both geometric and technological features are integrated. Parametric process using the software Grasshopper is here investigated, where sequences of algorithms and graphic results are shown for each step in order to provide a method applicable to different Origami.

Rigid-Foldable Origami Structures: parametric modelling using Grasshopper

3. Origami geometry: drawing with Grasshopper

In this section, a method to draw Origami using parametric tools is described. First, we identify the Origami pattern and the base module to reproduce, described by the periodicity vectors. The base module is the start point to develop each algorithm, because it governs the length and the width of the pattern. Then, we distinguish independent and dependent geometric parameters that describe the Origami. One important parameter to consider is the angle of deployment φ that manages the deployment of the system. Here, the process is applied to two specific cases: Yoshimura and Waterbomb Origami (Figure 1). Both of them are two degree of freedom Origami, presenting a deployment in space, which increases the modelling complexity.

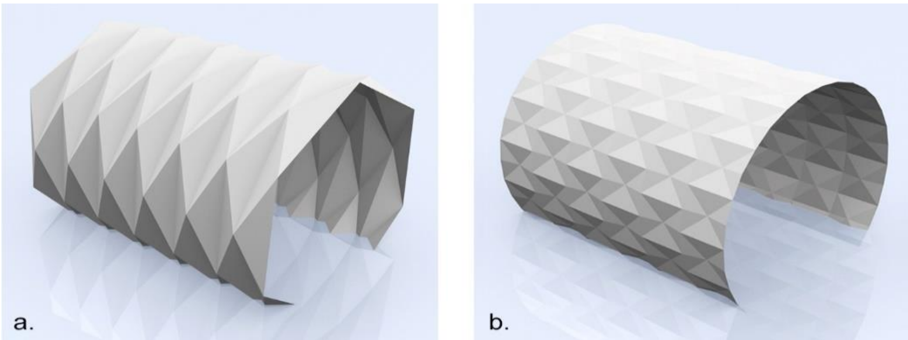

Fig.1: (a) Yoshimura Origami, (b) Waterbomb Origami.

3.1 Base Module: identification of Parameters and Drawing

The first step regards the identification of the parameters that describe a specific geometry. The dimensions of the base module – the length and the width – are independent parameters chosen by the designer. While the angle and the direction of rotation between adjacent faces has to be evaluated for each Origami. During the deployment, the faces exhibit rigid movements, resulting in free relative rotations between adjacent faces connected by edges. The angle of rotation δ between adjacent faces is obtained considering the rotation of the segments around the relative axis of an angle of deployment φ, $\varphi \in [0, \pi/2]$. Once are defined the dimensions and the angle of rotation δ, the base module and its deployment can be developed using parametric software Grasshopper. In the reference system (O, x, y, z), we define the points that describe the base module – the intersection points of the folds – and its central point on a plane x, y. The points with $x = 0$ rotate along x-axis of the angle of deployment φ, while the points with $x \neq 0$ rotate along y-axis of an angle $\delta = f(\varphi)$. The direction of rotation depends by the position of mountain and valley folds so it varies for each specific Origami. Then, the rotating points are linked to the central one creating the segments that represent the edges of the base module surfaces. Connecting the surfaces the base module and its deployment is obtained (Figure 2).

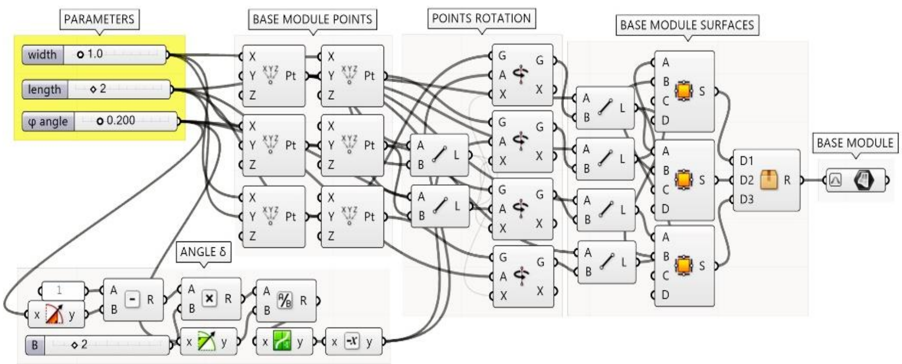

Fig 2: Yoshimura Origami algorithm developed on Grasshopper with indicated the main steps. In yellow independent parameters.

Now let us consider the geometry of two specific Origami, Yoshimura and Waterbomb, to exemplify the process previously described. These models present a similar pattern, but remarkable differences are evident in the deployment process. Yoshimura base module Origami (Figure 3a) consists of a rectangle $l \cdot w$ – geometric independent variables – divided into six triangles by a segment that vertically splits the rectangle into two equal parts and the diagonals of the rectangle. The edges of the triangles coincide with the folding lines of the module and indicate the direction of rotation (mountain folds in blue and valley folds in red). The relative rotation of these folds, measured by the angle of rotation δ and the angle of deployment φ, enable the deployment of the geometry. Considering φ as independent parameter ($\varphi \in [0, \pi/2]$), δ is expressed by the relation: $\delta = \arcsin(2 \cdot (1 - cos\varphi)/sin\varphi)$. While the most part of Origami present at least two geometric variables, Waterbomb base module Origami (Figure 3b) consists of a square $l \cdot l$ – unique geometric independent variable – divided into six triangles by two diagonals (valley folds), and a vertical segment located at half-length (mountain fold). In this case, the relative rotation angle δ is obtained by the following relation: $\delta = \arcsin((1 - cos\varphi)/sin\varphi), \varphi \in [0, \pi/2]$. The algorithmic process to draw the geometry follows the steps previously described, but in Waterbomb case the direction of points rotation is opposite of Yoshimura Origami.

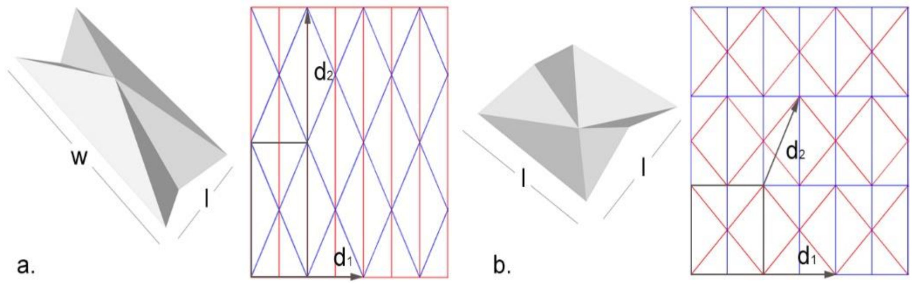

Fig 3: (a) Yoshimura Origami (b) Waterbomb Origami base module and pattern.

3.2 Origami pattern: compatible replication of the modules

On Origami modelling, problems concerning deployment mechanics and compatibility conditions to obtain a pattern have to be evaluated. In this section, we focus the attention on the periodicity vectors to replicate the base module and obtain an Origami pattern, considering compatibility conditions of adjacent modules during the deployment mechanism. The periodicity vectors are defined for specific Origami because they vary with the relative position of adjacent modules and the number of principal directions. For example, Yoshimura is characterized by a periodicity vector $\boldsymbol{d}_1 = l \cdot \boldsymbol{e}_1$ along direction $\boldsymbol{e}_1$, and a periodicity vector $\boldsymbol{d}_2 = w \cdot \boldsymbol{e}_2$ along direction $\boldsymbol{e}_2$ (Figure 3a). Waterbomb is characterized by a periodicity vector $\boldsymbol{d}_1 = l \cdot \boldsymbol{e}_1$ along direction $\boldsymbol{e}_1$, and a periodicity vector $\boldsymbol{d}_2 = {}^{l}/_{2}\, \boldsymbol{e}_1 + l\boldsymbol{e}_2$ along direction $\boldsymbol{e}_2$ (Figure 3b). Now it is assumed to have a second module with the same characteristics of the base module and with a side in common. If Origami have a deployment in plane, for example Miura Origami (Schenk M., Guest S. D., 2013), the modules can be copy and translated of the value given by the periodicity vectors. If Origami have a deployment in space, it is important to ensure the compatibility between adjacent modules, so the relative rotation angle has to be estimated. Let us suppose the second case, both Yoshimura and Waterbomb have a deployment in space. We consider a base module a and a second module b having a side in common with the first one, whose position is identified by the periodicity vectors. To evaluate the angle of compatibility between the two modules, we project a point A on a straight line s passing through the side in common, then we develop a circle C with center A' and radius $\overline{AA'}$, and we calculate the intersection between the circle C and a plane yz, so obtaining the point B. The angle $\widehat{AA'B}$ satisfy the compatibility condition and it lead to coincide A with B (Figure 4a). This geometric construction is developable using Grasshopper (Figure 4b) and it can be applied for Origami with a deployment in space.

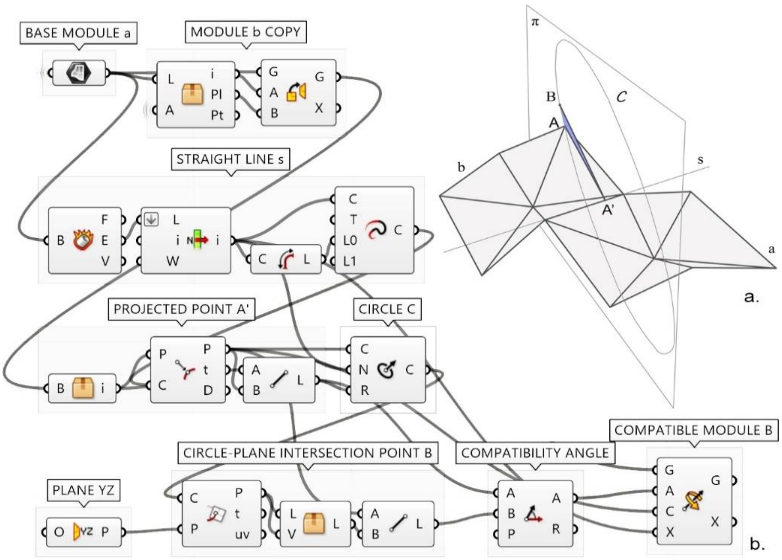

Fig 4: (a) Geometric construction to replicate modules, (b) Waterbomb Origami algorithm developed on Grasshopper with indicated the main steps.

4. Origami structural behaviour: data assignment and analysis

One of the purpose of this paper is to illustrate how to draw and analyse Origami in the same workflow. The first step of modelling was focused on the geometric drawing of Origami, now we are going to discuss the algorithmic sequence processed to define the structural features of a complex model, and to obtain its structural behaviour. The study of the behaviour for different statement of deployment is useful to identify the best operating configuration. In fact, varying the geometric parameters and their relation, different configurations can simultaneously be obtained extracting geometric, technological and structural data, integrated in the same process.

4.1 Focus on Origami Mesh: modelling conditions

The structural analysis using parametric methodology requires to convert a geometry in finite elements paying attention to some conditions that guarantee a correct process. The surfaces modelled on Grasshopper have to be converted to finite elements composed by triangular or quadrilateral mesh. Mesh surfaces, also called polygonal mesh, are defined as a set of adjacent polygons that determine the overall shape. In this section, we briefly investigate the conditions to develop a proper mesh system (to a detailed explanation about mesh see A. Tedeschi, 2014). A correct design method satisfies the following conditions (Figure 5):

1. Compatibility: coincidence between nodes of adjacent surfaces;
2. Orientation: same direction of surface vectors;
3. Discretization: the denser is the mesh the more detailed is the analysis.

In Origami modelling, the discretization in finite elements can be complicated to carry out because of problems related to compatibility and orientation, so the use of additional Grasshopper plug-ins, like Weaverbird – developed by Piacentino G. – is recommended. It includes some specific components to subdivide and join mesh, and to unify surface vectors. The system so acquired, composed by vertices, edges, faces, is the base point to process the analysis of the structural behaviour, in which vertices are converted to joints, edges to frames and faces to shells.

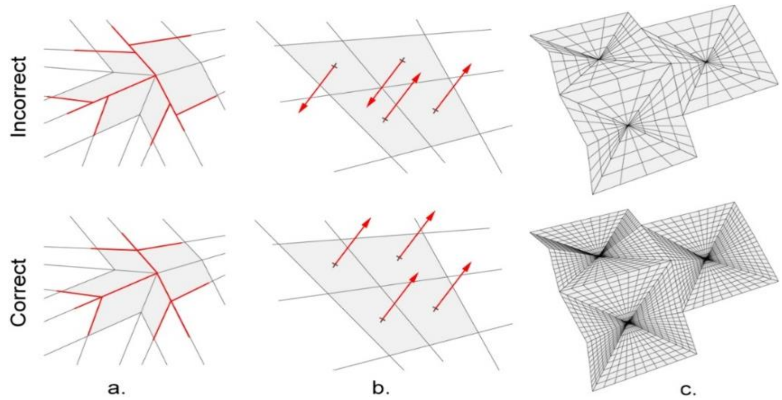

Fig 5: Incorrect (first row) and correct (second row) construction of mesh based on the conditions: (a) compatibility, (b) orientation, (c) discretization.

4.2 Structural behaviour: data assignment, analysis and results

Structural features enable the definition of Origami structures, including information about material, thickness, load conditions and restraints. In this step, structural parameters are integrated with geometric ones to describe an Origami model. Various Grasshopper plug-ins develop structural analysis all converting a geometry to a finite element system that enables an accurate representation of complex Origami and the relative results. Most part of plug-ins work inside Grasshopper, while Geometry-GymSap (developed by Mirtschin J.) allows to add structural features on Grasshopper, then the model is exported to FEA software Sap2000. It enables a more detailed analysis, in which stresses and membrane forces can be investigated carrying out the Origami behaviour.

Let us consider an Origami mesh geometry composed by $n \cdot n$ modules in a specific configuration of deployment $\varphi \in [0, \pi/2]$ (Figure 7a). The steps to obtain a structural model using Geometry-GymSap plug-in are described (Figure 6). First, mesh are converted to finite elements, whose material and thickness conditions are imposed as shell properties assigned to finite elements (a full database about material properties is included). Then, supports are defined identifying the vertices of the mesh to constrain and imposing restraint conditions. Finally, it is possible to assign different load conditions: for example temperature or pressure loads, mesh or point loads, static or modal cases, which can be combined.

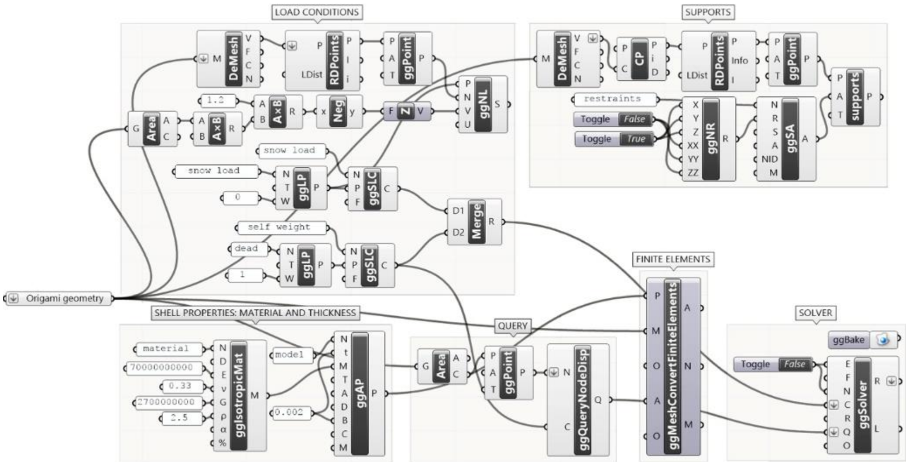

Fig 6: Algorithm to define structural features developed in Geometry-GymSap.

When the features are defined the model is assembled and exported automatically to Sap2000, where the analysis is made and the data are shown (Figure7b). Results can also be visualized on Grasshopper as values to combine with parameters to optimize the model and to find the best operating configuration.

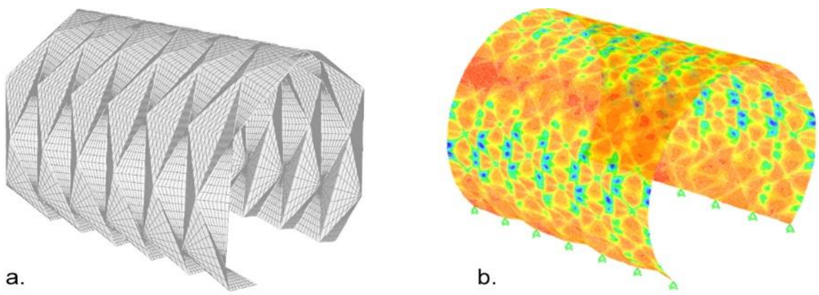

Fig 7: (a) Yoshimura Origami mesh surface, (b) Waterbomb Origami example of structural result: equivalent Von Mises Stress.

5. Conclusions

A method to design complex Origami was proposed, describing the geometric parameters that enable the definition of shapes, and showing the mechanism to correctly obtain the deployment and the structural behaviour of Origami models. As a result we get an interactive system that user can replicate to model different Origami and deployable-systems in general. Mechanical analysis results in detail and technological issues related to material and connection by edges are not described in this paper and remain to be discussed in future works.

References

Mirtschin J., GeometryGym, OpenBIM tools for architects, engineers and the construction industry, https://geometrygym.wordpress.com/, (accessed 27 April 2016).

Piacentino G., Weaverbird: topologic mesh editor, http://www.giuliopiacentino.com/weaverbird/, (accessed 27 April 2016).

Schenk M., Guest S. D., Origami folding: a structural engineering approach. Origami 5: 5th International Meeting of Origami Science, Mathematics and Education 5OSME, 291-303, 2011.

Schenk M., Guest S. D., Geometry of Miura-folded meta-materials, PNAS, 110, 3276-3281, 2013.

Tachi T., Rigid Origami Simulator, www.tsg.ne.jp/TT/software/

Tachi T., Simulation of Rigid Origami, in Origami^4, pp. 175-187, 2009.

Tachi T., Masubuchi M., Iwamoto M., Rigid Origami Structures with Vacuumatics: Geometric Considerations, IASS-APCS 2012, Seoul, Korea, 2011.

Tachi T., Freeform Origami Tessellations by generalizing Resch's patterns, Proceedings of ASME 2013 IDETC/CIE (Symposium on Origami-Based Engineering Design), DET2013-12326, Portland, USA, Aug. 4-7,2013.

Tedeschi A., AAD Algorithms-Aided Design. Parametric strategies using Grasshopper, Le Penseur, 2014.