

## MERLIN: A MATLAB implementation to capture highly nonlinear behavior of non-rigid origami

Ke LIU\*, Glaucio H. PAULINO<sup>a</sup>

\* Georgia Institute of Technology  
790 Atlantic Dr NW, Atlanta, GA, USA  
kliu89@gatech.edu

<sup>a</sup> Georgia Institute of Technology

### Abstract

Arisen from the geometric arrangements of panels and creases, unique mechanical properties such as foldability endow origami with promise for developing novel tunable and functional structural systems. To promote engineering applications of origami, a simplified but effective approach for investigation of the nonlinear mechanical behavior of non-rigid origami structures is essential. We propose a fully nonlinear, displacement-based formulation, for constructing quasi-static finite element analyses of origami structures based on a previously established bar-and-hinge simplification. The formulation leads to an efficient and robust numerical approach for predicting large displacements and large local deformations of origami structures. Comparison between actual paper-made models and numerical simulations hints the ability of the proposed approach in capturing key features of origami deformation. Thus the current work builds up a connection between theory and practice of origami structures, which has the potential to impact design, education, and applications of origami.

**Keywords:** origami, nonlinear analysis, bar-and-hinge model

### 1. Introduction

Currently, there are mainly two ways to analyze the deformation of origami structures. The first approach considers only the kinematics of the structure, for example, the rigid origami simulator by Tachi [12]. Some periodic origami patterns have closed-form expressions for their rigid mechanisms (Gattas *et al.* [2]). However, it is known that actual origami structures usually exhibit additional degrees of freedom beyond rigid folding mechanisms (Silverberg *et al.* [11], Wei *et al.* [13]). Due to the flexibility of the materials being used, the origami panels might be subjected to stretching, shearing and bending. Thus the rigid origami assumption does not reflect the true behaviors of a physical origami structure. The second approach is to build full finite element (FE) models using shell elements (Gattas and You [3]). It provides thorough information about the deformations of origami structures, including local stress distribution, but such a modeling requires a lot of detailed information of the design and the analysis is typically computationally intensive. Therefore, it is not suitable in many instances, for example, at the preliminary design stage, when detailed information is not available and iterations of designs are needed before a final decision. Sometimes predictions about the global mechanical behavior of origami, given just the crease pattern and general characteristics of the stiffness of panels, is what is really needed.

In 2011, Schenk and Guest [9] proposed a simplified bar-and-hinge model for the structural analysis of origami structures. This simplification of origami sheets also works as a foundation in our proposed analysis approach. In Schenk and Guest [9], the structural analysis formulation is only developed up to infinitesimal deformations and displacements. However, for many engineering applications, for

example, design of solar arrays (Zirbel *et al.* [16]), origami's ability to undergo large configurational transformations is essential. Therefore, a desired simplified analysis approach must also be able to capture the large displacements and local deformations of origami structures, which requires an appropriate nonlinear formulation.

*To promote engineering practice of origami, a simplified but effective analysis tool based on a fully nonlinear formulation is developed in this work.* This paper will focus on the computer implementation (in MATLAB code) of the proposed nonlinear analysis approach. The objective is to provide future researchers in origami engineering with an open-source and easy-to-use tool that serves as an assistant for research involving structural analysis of origami. The manuscript is organized as follows: the formulation will be introduced in Section 2. Section 3 explains details of the implementation. One example is provided in Section 4 to illustrate the capability of the tool.

## 2. Formulation basics

The bar-and-hinge simplification of origami is similar to the mass-spring system used in fields such as computer graphics for cloth animation (Bridson [1]), and chemistry for molecular dynamics simulation (van Schaik [8]). In structural analysis, under static and quasi-static loading, the mass is neglected (Wriggers [15]). The bar-and-hinge model was explicitly defined in the context of origami engineering as a simplification of origami sheets by Schenk and Guest [9]. The basic concept is illustrated in Figure 1, where a piece of simple fold is simplified to a bar frame with a rotational spring attached to the folding line (*ac*) to represent the folding stiffness of the origami shell. In case of the bending of panels, especially quadrilateral panels, it has been confirmed both theoretically and experimentally (Witten [14], Silverberg *et al.* [11]), that the bending curvature will mainly localize along a diagonal of the panel. Let us call such a diagonal a bending line. Thus, Figure 1 also applies to modeling of a bending deformation. Panels that have more than 4 nodes need further investigation to find proper triangulation schemes to represent their bending modes.

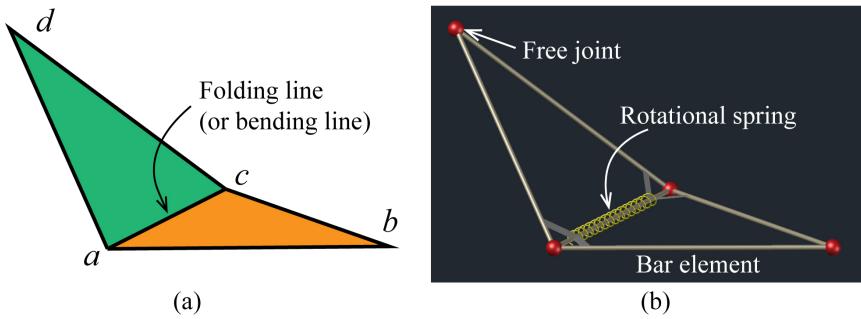


Figure 1: Illustration of the bar-and-hinge model for origami simulation.

Assuming that the structure is nonlinear elastic, we can write down the expression for the total potential energy of this simplified origami model:

$$\Pi = U_{bar} + U_{spr} - V_{ext} \quad (1)$$

The total potential energy contains 3 additively separable terms: (1) the strain energy stored in bars ( $U_{bar}$ ), accounting for the in-plane deformation strain energy of the origami sheet; (2) the strain energy stored in bending and folding deformations ( $U_{spr}$ ), simulating the out-of-plane deformation strain energy; and (3) the work ( $V_{ext}$ ) done by external loads. Thus the equilibrium of the system is given as:

$$\frac{\partial \Pi}{\partial \mathbf{u}} = \frac{\partial U_{bar}}{\partial \mathbf{u}} + \frac{\partial U_{spr}}{\partial \mathbf{u}} - \mathbf{F} = \mathbf{T}_{bar} + \mathbf{T}_{spr} - \mathbf{F} = 0 \quad (2)$$

where  $\mathbf{u}$  is the nodal displacement vector, and  $\mathbf{T}_{bar}$  and  $\mathbf{T}_{spr}$  denote internal forces. Accordingly, we have the tangent stiffness matrix as a summation of the contributions from both the bars and the rotational springs:

$$\mathbf{K}_T = \mathbf{K}_{bar} + \mathbf{K}_{spr} \quad (3)$$

Considering both material and geometric nonlinearities, for each bar element, we assume its stored energy density is  $W$ , as a function of the one dimensional Green-Lagrange strain  $E_x$ . Thus its internal force and tangent stiffness matrix can be expressed as (Wriggers [15]):

$$\mathbf{T}_{bar}^i = A_i L_i \frac{\partial W}{\partial E_x} \frac{\partial E_x}{\partial \mathbf{u}_i} = A_i L_i S_x \frac{\partial E_x}{\partial \mathbf{u}_i} \quad (4)$$

$$\mathbf{K}_{bar}^i = A_i L_i \left[ \frac{\partial W}{\partial E_x} \frac{\partial^2 E_x}{\partial \mathbf{u}_i^2} + \frac{\partial^2 W}{\partial E_x^2} \frac{\partial E_x}{\partial \mathbf{u}_i} \left( \frac{\partial E_x}{\partial \mathbf{u}_i} \right)^T \right] = A_i L_i \left[ S_x \frac{\partial^2 E_x}{\partial \mathbf{u}_i^2} + C \frac{\partial E_x}{\partial \mathbf{u}_i} \left( \frac{\partial E_x}{\partial \mathbf{u}_i} \right)^T \right] \quad (5)$$

where  $\mathbf{u}_i$  contains the associated degrees of freedom of a single bar element. The 2<sup>nd</sup> Piola-Kirchhoff stress and the tangent modulus (scalar) are denoted as  $S_x$  and  $C$ , respectively. The expression is for an individual bar element and there is no summation indicated by the repeated indices.

Similarly, assuming the stored energy in a rotational spring element is  $\psi$ , which is a function of the dihedral angle  $\theta$ , we obtain:

$$\mathbf{T}_{spr}^j = L_j \frac{\partial \psi}{\partial \theta} \frac{\partial \theta}{\partial \mathbf{u}_j} = L_j M \frac{\partial \theta}{\partial \mathbf{u}_j} \quad (6)$$

$$\mathbf{K}_{spr}^j = L_j \left[ \frac{\partial \psi}{\partial \theta} \frac{\partial^2 \theta}{\partial \mathbf{u}_j^2} + \frac{\partial^2 \psi}{\partial \theta^2} \frac{\partial \theta}{\partial \mathbf{u}_j} \left( \frac{\partial \theta}{\partial \mathbf{u}_j} \right)^T \right] = L_j \left[ M \frac{\partial^2 \theta}{\partial \mathbf{u}_j^2} + k \frac{\partial \theta}{\partial \mathbf{u}_j} \left( \frac{\partial \theta}{\partial \mathbf{u}_j} \right)^T \right] \quad (7)$$

where  $M$  denotes the resisting moment per unit length, and  $k$  denotes the tangent rotational stiffness. Compared to the linear elastic formulation of Schenk and Guest [9], the stiffness matrix of the entire structure is augmented with second-order terms, and it is a function of the displacements instead of a constant matrix.

### 3. Implementation

This section describes the main aspect of the MERLIN software and how it can be utilized to model general origami patterns. It includes information about the domain definition, model definition, constitutive relationships, nonlinear solution scheme and output data management.

#### 3.1. Domain definition

To define the geometry and topology of an origami structure, two variables must be specified: (1) the coordinates of nodes (vertices), and (2) the groups of nodes that defines the panels. The coordinates of the nodes are stored in a  $N_n \times 3$  matrix **Node**, of data type “array”, where  $N_n$  is the number of nodes. Each row  $p$  of the matrix has the nodal coordinates  $x$ ,  $y$  and  $z$  for node  $p$ . The topology information is stored in variable **Panel**, which is of data type “cell array”. This allows freedom on defining origami patterns that contain panels with different numbers of nodes, for example, quadrilaterals and triangles. Therefore, the variable **Panel** has  $N_p$  elements, and each element is a vector containing the indices of the nodes in a particular panel. The indices should appear following a counter clockwise order by geometric adjacency. For example, if Figure 1 shows a panel, it shall be labeled as  $[a, b, c, d]$ .

The boundary conditions are defined following a conventional manner in truss analysis. A matrix **Supp** contains the information about supports. Each row of **Supp** consists of a node index, and three

displacement fixities (0 or 1) in the  $x$ ,  $y$ , and  $z$  directions respectively. The loading information is collected in another matrix **Load**, each row of which consists a node index, and applied forces in  $x$ ,  $y$  and  $z$ . A zero specifies no force in that direction.

The conversion of the model from the original origami sheet to the simplified bar-and-hinge model is automated in the implementation. This feature brings a lot of convenience because then users just need to provide essential information about their origami sheets, and do not need to worry about the remodeling with bars and rotational springs. The procedure is elaborated in the following sections.

### 3.2. Model definition

With the input variables **Node** and **Panel**, the simplified bar-and-hinge model can be defined correspondingly. The data to define a bar-and-hinge model are stored in three variables: **Fold**, **Bend** and **Bars**, as listed in Table 1. The **Fold** and **Bend** variable have the same format. Following the nodal indices in Figure 1, each row of **Fold** (or **Bend**) contains a vector with four entries:  $[a, c, b, d]$ , where the link  $ac$  identifies the rotation axis. Such a group of four nodes forms a rotational spring element. Essentially, the variables **Fold** and **Bend** can be combined into one variable because they have information of the same type. The only difference is the corresponding rotational stiffness. For educational purpose, we keep them separate in this tool. The variable **Bars** contains all the pairs of nodes as edges in the triangulation, which defines the bar elements.

Bending lines are first defined. For triangular panels, the current setup requires no further discretization. For quadrilateral panels, each of them will be divided into two triangles by the shorter diagonal which will become a bending line. If the two diagonals are of equal length, depending on the order of indices listed in the variable **Panel**, the link connecting the first and third node will be assigned as a bending line. Panels that have more than 4 nodes are not automatically supported, however, one may divide such panels into triangles manually and assign the internal links as bending lines.

Table 1: Bar-and-hinge model definition variables

Variable Name	Size	Description
<b>Fold</b>	$N_f \times 4$	List of all the rotational spring elements representing folding action. Each row lists the indices of 4 nodes with the first two referring to the pair of nodes that form the rotation axis.
<b>Bend</b>	$N_d \times 4$	List of all the rotational spring elements representing bending action. Same structure as <b>Fold</b> .
<b>Bdry</b>	$N_r \times 2$	List of all boundary edges. Each row contains the indices of a pair of nodes. (Intermediate variable)
<b>Bars</b>	$N_b \times 2$	List of all bars (edges in the triangulation). Each row contains the indices of a pair of nodes. $N_b = N_f + N_d + N_r$ .

Each element for bending (e.g.,  $[a, c, b, d]$ ) contains two triangles, indexed by  $[a, c, b]$  and  $[a, c, d]$ . Combining with the triangular panels, we obtain a triangular mesh of the entire origami sheet as an immediate consequence. Based on this triangulation, we define an incidence matrix as

$$[\mathbf{C}]_{p,q} = \begin{cases} 1, & \text{if node } p \text{ is in triangle } q \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

Then the element-wise adjacency matrix can be obtained as

$$\mathbf{G}_E = \mathbf{C}^T \mathbf{C} \quad (9)$$

The off-diagonal entries of  $\mathbf{G}_E$  show how many nodes are shared by two triangles. If entry  $(m, n)$  equals 2, it means that triangle  $m$  and triangle  $n$  share an edge. Then there are four nodes contained in

the two adjacent triangles, completing a rotational spring element. Examining the nodes of triangles  $m$  and  $n$  by their indices, the two shared nodes identify the rotation axis, i.e. a bending line or folding line. If we avoid rotational spring elements that are already marked for bending action, we are left with only the elements for folding simulation.

The bar elements in the simplified model are the edges in the triangulation. The first two columns of the **Fold** and **Bend** variables, i.e., the folding lines and bending lines, defines all the internal bars. To complete the list of all bars, those that lies on the boundaries of the sheet need to be identified. Splitting all the triangles to edges (pairs of nodes) and collecting them together, the edges that only appear once are the boundary edges, because an internal edge must appear twice as it is shared by two triangles. These boundary edges are collected in the variable **Bdry**, as a two-column matrix containing indices of nodes (see Table. 1). Eventually, combining the first two columns of **Fold** and **Bend** and the entire **Bdry**, we can obtain the variable **Bars**.

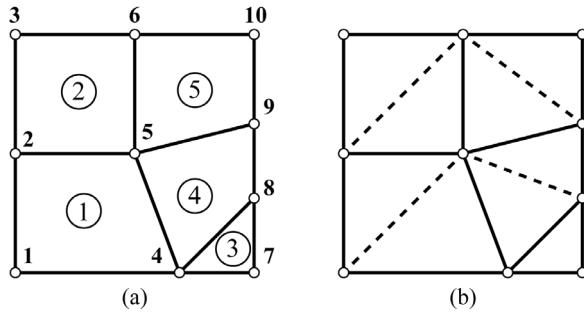


Figure 2: A sample origami pattern with 5 panels and 10 nodes. (a) The origami pattern. (b) The corresponding bar-and-hinge model, where the solid lines are folding lines and boundary edges, and the dashed lines are bending lines. All lines represent pin-jointed bars.

A simple example is provided in Figure 2. There are 5 panels in this origami pattern. It results in 4 bending lines, 5 folding lines, and 9 boundary edges. After triangulation, the variables **Fold**, **Bend** and **Bdry** are detailed in Eq. (3), (4) and (5). Eventually, there are 18 bars and 9 rotational spring elements in the corresponding bar-and-hinge model.

$$\mathbf{Bend} = \begin{bmatrix} 1 & 5 & 2 & 4 \\ 2 & 6 & 3 & 5 \\ 6 & 9 & 5 & 10 \\ 5 & 8 & 4 & 9 \end{bmatrix} \quad (10)$$

$$\mathbf{Fold} = \begin{bmatrix} 2 & 5 & 1 & 6 \\ 4 & 5 & 1 & 8 \\ 4 & 8 & 5 & 7 \\ 5 & 9 & 6 & 8 \\ 5 & 6 & 2 & 9 \end{bmatrix} \quad (11)$$

$$\mathbf{Bdry} = \begin{bmatrix} 1 & 2 & 3 & 6 & 9 & 8 & 7 & 4 & 1 \\ 2 & 3 & 6 & 10 & 10 & 9 & 8 & 7 & 4 \end{bmatrix}^T \quad (12)$$

### 3.3. Constitutive relationship definition

Nonlinear constitutive relationships are assigned to both the bars and rotational springs. The two functions “BarConst” and “RotConst” specify the relationships.

The function “BarConst” takes the one dimensional Green-Lagrange strain  $E_x$  of bars as input. The outputs are the 2<sup>nd</sup> Piola-Kirchhoff stress  $S_x$ , the tangent modulus  $C$ , and the stored energy density  $W$ .

The formulas describing the relationship between these quantities can be found in Wriggers [15], which are also hinted in Eq. (5). The current implementation provides a two-term Ogden model (Ogden [7]), which is a model for hyperelastic materials that offers rich tunability. The expression for the stress-strain relationship is:

$$S_x = \frac{C_0}{\alpha_1 - \alpha_2} \left[ \left( \sqrt{2E_x + 1} \right)^{\alpha_1 - 2} - \left( \sqrt{2E_x + 1} \right)^{\alpha_2 - 2} \right]$$

The tunable parameters are the initial modulus  $C_0$ , and the other two material constants  $\alpha_1$  and  $\alpha_2$ . The initial modulus describes how stiff the bars are when undeformed. Typically, we assign the bars a relatively stiff behavior so that they will only undergo small deformations, because based on observation, in general, the creases in real origami structures do not shrink or stretch significantly. However, the nonlinear material assumption leaves broad possibility for future development.

The function “RotConst” specifies the rotational stiffness at the folding lines and bending lines, which is also nonlinear. In the current implementation, the rotational springs for folding and bending actions are described by the same constitutive model with different parameters. The model describes a piecewise behavior for the rotational springs. The formula for the moment-angle relationship is given as follows:

$$M = \begin{cases} k_0(\theta_1 - \theta_0) + \frac{2k_0\theta_1}{\pi} \tan\left(\frac{\pi(\theta - \theta_1)}{2\theta}\right), & 0 < \theta < \theta_1 \\ k(\theta - \theta_0), & \theta_1 \leq \theta \leq \theta_2 \\ k_0(\theta_2 - \theta_0) + \frac{2k_0(2\pi - \theta_2)}{\pi} \tan\left(\frac{\pi(\theta - \theta_2)}{4\pi - 2\theta_2}\right), & \theta_2 < \theta < \pi \end{cases}$$

This model is an extension of the linear stiffness model for rotations (Schenk and Guest [9]). The excessive moment near 0 and  $2\pi$ , will prevent local penetration of the two jointed panels. The fold (or panel) is flat when  $\theta = \pi$ . The tunable parameters of this model are the linear rotational stiffness  $k_0$ . The role of parameters  $\theta_1$  and  $\theta_2$  is to create a smooth ramp as the stiffness becomes infinitely large. When  $\theta$  is between  $\theta_1$  and  $\theta_2$ , the behavior is linear. The linear rotational stiffness  $k_0$  distinguishes the folding and bending actions. We assign  $k_0 = k_b$  for the rotational spring elements of bending ( $k_{facet}$  in Schenk and Guest [9]), and  $k_0 = k_f$  for folding ( $k_{fold}$  in Schenk and Guest [9]). One can also define their own constitutive relationships by changing the formulas in the two functions.

### 3.4. Nonlinear solution scheme

The nonlinear equilibrium problem defined in Eq. (2) is solved by the algorithm called Modified Generalized Displacement Control Method (MGDCM) proposed by Leon *et al.* [6]. The method is essentially an arc-length type method, following an incremental-iterative manner. Compared to other types of nonlinear solver, for example, the popular Newton-Raphson method, arc-length methods allow us to track the whole equilibrium path of the system even with large nonlinearities.

Denote the load factor as  $\lambda$ , and the prescribed initial load factor as  $\overline{\Delta\lambda}$ . The algorithm is summarized as follows:

1.	Initialization: $\mathbf{u}_0^0 = \mathbf{0}$ and $\lambda_0^0 = \mathbf{0}$
2.	Start increments $i$ : $i = i + 1$ , $\mathbf{u}_0^i = \mathbf{u}_k^{i-1}$ , and $\lambda_0^i = \lambda_k^{i-1}$
3.	Start iterations $k$ : $k = 0$
4.	Compute internal forces, tangent stiffness matrix and residual vector: $\mathbf{T}_{k-1}^i = \mathbf{T}(\mathbf{u}_{k-1}^i)$ , $\mathbf{K}_{k-1}^i = \mathbf{K}(\mathbf{u}_{k-1}^i)$ , $\mathbf{R}_{k-1}^i = \lambda_{k-1}^i \mathbf{F} - \mathbf{T}_{k-1}^i$

5.	Solve linearized systems: $\mathbf{K}_{k-1}^i \Delta \hat{\mathbf{u}}_k^i = \mathbf{F}$ , $\mathbf{K}_{k-1}^i \Delta \tilde{\mathbf{u}}_k^i = \mathbf{R}_{k-1}^i$
6.	Determine increments: $\Delta \lambda_k^i = \begin{cases} \overline{\Delta \lambda}, & i=1, k=1 \\ -\frac{\Delta \hat{\mathbf{u}}_1^1 \cdot \Delta \tilde{\mathbf{u}}_k^1}{\Delta \hat{\mathbf{u}}_1^1 \cdot \Delta \hat{\mathbf{u}}_k^1}, & i=1, k>1 \\ \operatorname{sgn}(\Delta \hat{\mathbf{u}}_1^{i-1} \cdot \Delta \hat{\mathbf{u}}_1^i) \cdot \overline{\Delta \lambda} \left  \frac{\Delta \hat{\mathbf{u}}_1^1 \cdot \Delta \tilde{\mathbf{u}}_1^1}{\Delta \hat{\mathbf{u}}_1^{i-1} \cdot \Delta \hat{\mathbf{u}}_1^i} \right ^{1/2}, & i>1, k=1 \\ -\frac{\Delta \hat{\mathbf{u}}_1^i \cdot \Delta \tilde{\mathbf{u}}_k^i}{\Delta \hat{\mathbf{u}}_1^i \cdot \Delta \hat{\mathbf{u}}_k^i}, & i>1, k>1 \end{cases}$
7.	Update: $\Delta \mathbf{u} = \Delta \lambda_k^i \Delta \hat{\mathbf{u}}_k^i + \Delta \tilde{\mathbf{u}}_k^i$
8.	Convergence test: (1) If $\ \Delta \mathbf{u}_k^i\  \leq tol$ (typically, $tol = 10^{-6}$ ), stop iteration, and go to 2. (2) Otherwise, $k = k + 1$ , and go to 4.

### 3.5. Output data management

The primary outputs are the loading history and nodal displacement history, which contains the information about the equilibrium path of the structural system. In addition, other pieces of information about the folding angles, bending amplitudes (angles), and bar elongations/shrinkage are also collected, as listed in Table 2. These variables essentially allow recovery of all information involved in the simulation, for example, we can compute the strain energy distribution due to folding, bending and stretching of the panels along its deformation history. Visualization is also provided within the tool to show the animation of the simulation, which also provides functions to record the animations to videos or GIF images.

Table 2: Output variables

Variable Name	Size	Description
<b><i>U_his</i></b>	$N_{dof} \times N_{icr}$	History of nodal displacements. $N_{icr}$ denotes the number of increments, which is prescribed. $N_{dof}$ refers to the number of degrees of freedom, which equals to 3 times the number of nodes.
<b><i>LF_his</i></b>	$1 \times N_{icr}$	Variation history of the load factor.
<b><i>ExBar</i></b>	$N_b \times N_{icr}$	Green-Lagrange strain of bar elements.
<b><i>FdAngle</i></b>	$N_f \times N_{icr}$	Abosolute folding angles (dihedral angles).
<b><i>BdAngle</i></b>	$N_d \times N_{icr}$	Bending amplitudes measured in dihedral angles.

### 4. Example - Bending of the Eggbox origami

Using the proposed approach, we are able to numerically analyze various behaviors of origami structures. In this example, we are going to investigate the bending of the Eggbox sheet, which is a non-developable but two-way flat foldable origami structure, as shown in Figure 3(a). Based on rigid origami assumption, the Eggbox has only one degree of freedom, i.e., the folding mechanisms, categorized as planar kinematics. However, in reality, it can also undergo out-of-plane kinematics because the quadrilateral facets do not remain planar during global deformation. It is interesting that when subject to out-of-plane global deformations the Eggbox sheet exhibits a spherical deformation mode (Schenk [10]), as shown in Figure 3. The coupling ratio of the two principle curvatures of the deformed Eggbox sheet, was only studied for small global deformations numerically in Schenk [10].

Utilizing the ability of the proposed tool, we can investigate large global bending behavior of the Eggbox sheet.

The boundary conditions are illustrated in Figure 3, reproducing the situation as in Figure 3(b). Forces are applied with a unit magnitude, and the load factor reflects the magnitude of forces. The constitutive relationships are characterized by the parameters:  $C_0 = 10^8$ ,  $\alpha_1 = 5$ ,  $\alpha_2 = 1$ ,  $k_f = 0.1$ ,  $k_b = 2$ ,  $\theta_1 = \pi/8$  and  $\theta_2 = 15\pi/8$ . The cross-sectional areas of bars are assumed to be  $10^{-5}$ .

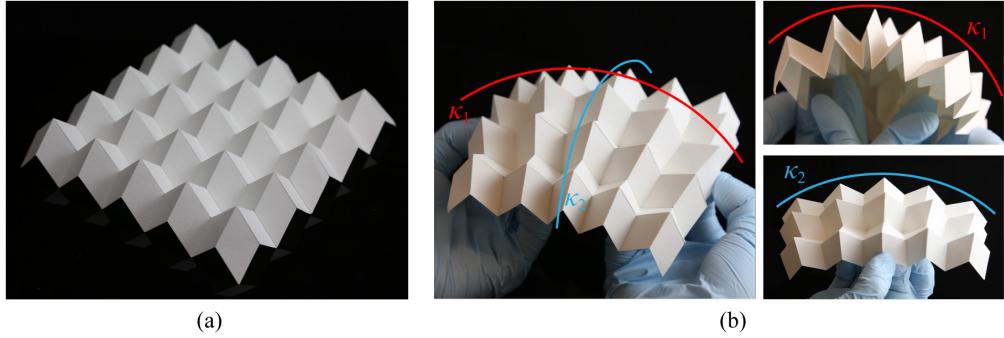


Figure 3: Eggbox origami (unit cells placed in a  $5 \times 5$  pattern). (a) Paper model. (b) Bending of the Eggbox sheet.

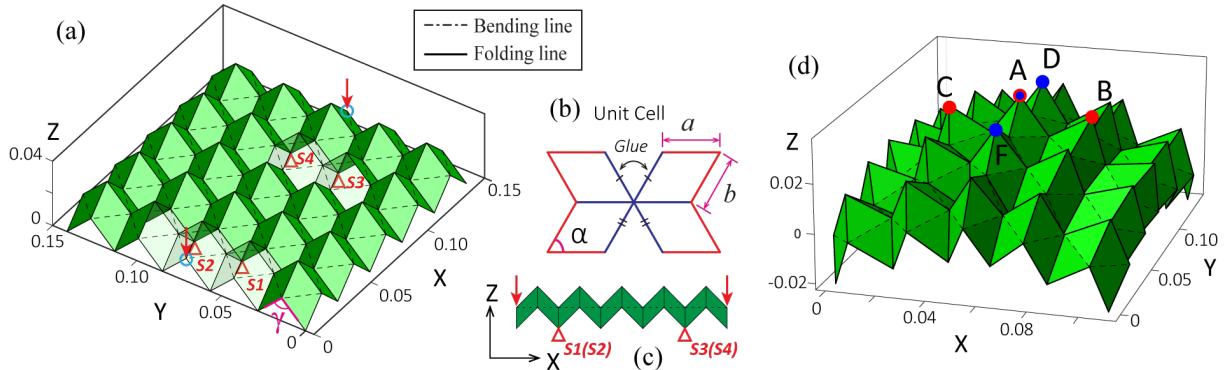


Figure 4: Numerical simulation setups and boundary conditions. (a) Red triangles indicate pin supports, detailed in Eq. (13). Blue circles show where the external forces are applied. The forces are applied vertically to the negative  $z$ -direction, as illustrated by the red arrows. The angle  $\gamma = 90^\circ$  describes the initial configuration. (b) A flattened unit cell of the Eggbox sheet, where  $a = b = 0.02$ ,  $\alpha = 60^\circ$ . (c) A side view of the boundary conditions. Clearly, the structure is subject to bending. (d) The deformed shape of the origami sheet by numerical analysis. Points A, B, C are used to calculate the curvature  $\kappa_1$  (A, D, F for  $\kappa_2$ ), by fitting the three points to a circle and taking the reciprocal of the radius.

$$\mathbf{Supp} = \begin{bmatrix} S1 & 1 & 1 & 1 \\ S2 & 1 & 0 & 1 \\ S3 & 0 & 1 & 1 \\ S4 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

The number of increments are specified to be 200 and the prescribed initial load factor  $\bar{\lambda}$  is set to be 0.03. The simulation result is presented in Figure 4(b). The bending creates a (distorted) spherical shape for the Eggbox sheet, similar to the configuration as suggested by the physical model (see Figure 3). The analysis of data is shown in Figure 5. One intermediate state (State (1)) and the final state (State (2)) of the deformation are plotted. The coupling ratio of the two principle curvatures at the two states are 1.019 and 1.090. Both are positive and around 1.0, which is the prediction from the

linear analysis (Schenk [10]). As the global deformation become large, the linear analysis based on infinitesimal deformation assumption is no longer valid. We also plot the stored energy distribution at these two states. As we can see, the bending deformation of panels is occupying the majority of the total stored energy in the (elastic) system. The energy stored in folding deformation increases its percentage as the forces become large.

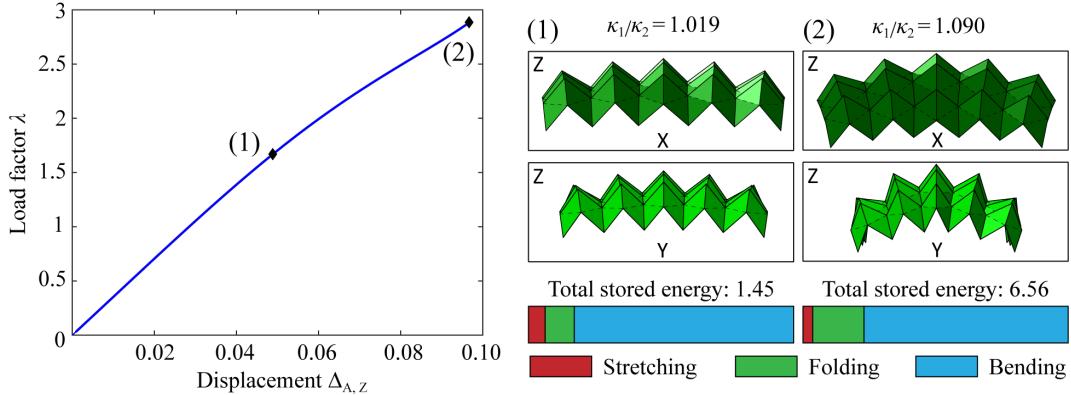


Figure 5: Simulation result. The diagram shows the load (factor) vs. displacement curve. The displacement is measured as the z-displacement of node A as marked in Figure 4. The occurrences of state (1) and state (2) are marked with black diamonds in the diagram. The corresponding deformations (each from two views) are shown on the right. The coupling ratio of curvatures and distribution of stored energy are also provided. The energy of panel stretching (and shearing) is represented by the shrinkage and elongation of bars.

## 5. Conclusion

The bar-and-hinge model, as a simplified model, provides insight into the essential global mechanical behavior of origami structures. The previously established linear formulation (Schenk and Guest [9]) has gained some popularity in the research field of origami engineering (Fuchi *et al.* [5]). The present work and implementation, extends the simplified model to a new nonlinear paradigm that handles large displacements and large deformations. This leads to a simple and efficient approach for understanding the mechanics of large global deformations of origami structures, which is the most attractive property of origami to many engineering applications.

The MERLIN MATLAB implementation is provided in reference [17] to encourage both research and education in the field by providing an easy-to-use access to the nonlinear mechanics of origami structures. The code was created with a balance of performance and readability in mind and the analysis procedure is maximally automated with minimized input information.

## Acknowledgements

We acknowledge support from the US NSF (National Science Foundation) through Grant 1538830. In addition, Ke Liu acknowledges support of the China Scholarship Council (CSC), and Glaucio H. Paulino acknowledges support of the Raymond Allen Jones Chair at the Georgia Institute of Technology.

## References

- [1] Bridson R., Marino S., Fedkiw R., Simulation of clothing with folds and wrinkles. In: Breen D., Lin M. (Eds.), *Proceedings of the Eurographics/SIGGRAPH Symposium on Computer Animation*, 2003; **21**, 28-36.
- [2] Gattas J.M., Wu W., You Z., Miura-base rigid origami: parameterizations of first-level derivative and piecewise geometries. *Journal of Mechanical Design*, 2013; **135**; 111011.

- [3] Gattas J.M., You Z., Quasi-static impact of indented foldcores. *International Journal of Impact Engineering*, 2014; **73**; 15-29.
- [4] Filipov E.T., Tachi T., Paulino G.H., Origami tubes assembled into stiff, yet reconfigurable structures and metamaterials. *Proceedings of the National Academy of Sciences*, 2015; 201509465.
- [5] Fuchi K., Buskohl P.R., Bazzan G., Durstock M.F., Reich G.W., Vaia R.A., Joo J.J., Origami actuator design and networking through crease topology optimization. *Journal of Mechanical Design*, 2015; **137**(9); 091401.
- [6] Leon S.E., Lages E.N., de Araújo C.N., Paulino G.H., On the effect of constraint parameters on the generalized displacement control method. *Mechanics Research Communications*, 2014; **56**; 123-129.
- [7] Ogden R.W., *Non-linear Elastic Deformations*. Dover Publications, 1997.
- [8] van Schaik R.C., Berendsen H.J.C., Torda A.E., van Gunsteren W.F., A structure refinement method based on molecular dynamics in four spatial dimensions. *Journal of Molecular Biology*, 1993; **234**; 751-762.
- [9] Schenk M., Guest S.D., Origami folding: A structural engineering approach. In: *Origami 5*, CRC, 2011; 293-305.
- [10] Schenk M., *Folded Shell Structures*. PhD Thesis, Clare college, University of Cambridge, 2011.
- [11] Silverberg J.L., Evans A.A., McLeod L., Hayward R.C., Hull T., Santangelo C.D., Cohen I., Using origami design principles to fold reprogrammable mechanical metamaterials. *Science*, 2014; **345**(6197); 647-650.
- [12] Tachi T., Simulation of rigid origami. In: *Origami 4*, CRC, 2006; 175-187.
- [13] Wei Z.Y., Guo Z.V., Dudte L., Liang H.Y., Mahadevan L., Geometric mechanics of periodic pleated origami. *Physical Review Letters*, 2013; **110**(21); 1-5.
- [14] Witten T.A., Stress focusing in elastic sheets. *Reviews of Modern Physics*, 2007; **79**(2); 643-675.
- [15] Wriggers P., *Nonlinear Finite Element Methods*. Springer, 2008.
- [16] Zirbel S.A., Trease B.P., Thomson M.W., Lang R.J., Magleby S.P., Howell L.H., HanaFlex: a large solar array for space applications. *Proc. SPIE 9467*, Micro- and Nanotechnology Sensors, Systems, and Applications VII, 94671C.
- [17] Website: [ghpaulino.com](http://ghpaulino.com).