About Freeform Origami

Tomohiro Tachi*

April 6, 2017

1 Overview

Freeform Origami is a software written by Tomohiro Tachi for enabling interactive design of three-dimensional origami based on continuously modifying origami shapes under several geometric constraints.

The now-released or lower version of the software is a proprietary freeware applying the following terms of use.

1.1 Terms of Use

1. Non-commercial redistribution is allowed, if and only if the software is distributed without any change and with this document.
2. The user is allowed to use the software if and only if the work achieved using the software including research publications, exhibitions, educational workshops, and architectural/product designs, (henceforth resulting works), satisfy either (or both) of the following conditions.

    1. (Personal Use): The resulting works are personal, i.e., neither public, educational, nor commercial.

    2. (Non-Personal Use with Attribution): The resulting works include a reference to the software "Freeform Origami by Tomohiro Tachi" or to the related paper publication listed in the section 1.3.
3. Any other type of usage of the software is by default prohibited, i.e., requires the author's explicit permission.

1.2 URLs

- The software is downloadable from `http://www.tsg.ne.jp/TT/software/#ffo`.
- Any questions and bug reports are planned to be responded at freeform origami group at curvedfolding.com (`http://www.curvedfolding.com/group/freeformorigami/`).

1.3 Papers

For technical details of the software, the following references can help.

*Department of General Systems Studies, The University of Tokyo, 3-8-1 Komaba, Meguro-Ku, Tokyo 153-8092, Japan, `tachi@idea.c.u-tokyo.ac.jp`

- Tomohiro Tachi, "Generalization of Rigid-Foldable Quadrilateral-Mesh Origami," Journal of the International Association for Shell and Spatial Structures (IASS), 50(3), pp. 173–179, December 2009.
- Tomohiro Tachi, "Freeform Variations of Origami", in Proceedings of The 14th International Conference on Geometry and Graphics (ICGG 2010), Kyoto, Japan, pp. 273–274, August 5-9, 2010.
- Tomohiro Tachi, "Freeform Rigid-Foldable Structure using Bidirectionally Flat-Foldable Planar Quadrilateral Mesh", Advances in Architectural Geometry 2010, pp. 87–102, September 2010.

2 Basic User Interface

2.1 Screens

FreeformOrigami.exe shows the screen like Figure 1.

Left pane 3D View. Shows 3D view of the model when the model is loaded. Most of manipulation is done on this screen.

Right pane optional graphics: this is shown when related constraints are activated. From top to bottom:

1. Crease pattern shows the developed pattern of the form when "developable" constraint is activated (check System→developable).
2. Flat-folded pattern shows the X-ray view of the form in the completely flat-folded state if "flat-foldable" constraint is activated (check System→flat-foldable).
3. Reciprocal Figure shows the reciprocal figure related to the first order folding mode when "shaky" constraint is activated (check System→shaky).

2.2 Coloring Scheme

The edges are colored according to the crease property. These assignment can be changed using Tool→assign.

Red	Mountain
Blue	Valley
Dark Gray	General Crease
Light Gray	Triangulation

2.3 Changing View

In order to change view, use mouse:

Rotate View : Use **Right Button Drag** for 3D rotation (for 3D view) and 2D rotation (for crease pattern and flat-folded pattern views).

Pan View : use **Middle Button Drag** or **Shift + Right Button Drag** for panning.

Zoom View : use **Wheel** to zoom up and down.

Most of other operations are done by left click/drag, selecting menu, or pressing key, and their combination depending which tool you are using.

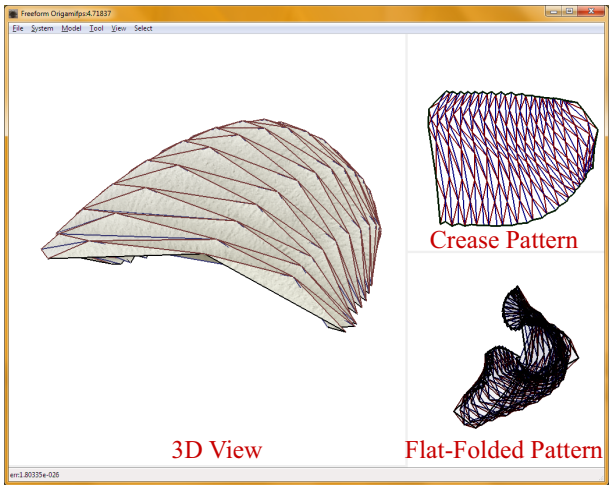

Figure 1: Screen shot of the software.

2.4 Simulation Mode vs. Edit Mode

The software runs in two different mode "Simulation Mode" and "Design Mode." The modes can be switched by "System → Simulation Mode" check list. Advanced: By pressing **Tab** key, the software temporarily runs in the opposite mode.

Simulation Mode When the background color of Crease Pattern and Flat-Folded Pattern windows is **gray**, CP and FF are fixed, and the system runs in simulation mode. In this mode, the software does a rigid origami simulation using truss elements.

Edit Mode When the background color of Crease Pattern and Flat-Folded Pattern windows is **white** (CP and FF are fixed), the system runs in simulation mode. In this mode, manipulation to the 3D object affects the crease pattern and flat-folded pattern. The deformation of the object follows the constraints checklist in System menu.

3 File Menu

3.1 Open/Save/Append

Freeform Origami supports files of ***.dxf (2d drawing, i.e., crease pattern), ***.obj (3d mesh folded form), or ***.fold (3d mesh folded form). You can drag and drop the files to the window to open.

The files must be prepared carefully so that the software can successufully read the files. Freeform Origami is currently very sensitive to the file formatting. DO NOT OVERESTIMATE. Like Spelunker, IT DIES of different easy reasons; it used to die even with extra white spaces. Once imported correctly, the software is quite robust.

You can drag and drop multiple files or open multiple files to manipulate more than one meshes at the same time. Also, you may append files in the same scene, so that you can apply geometric constraints between multiple pieces.

DXF Two dimensional crease pattern must be prepared in dxf format.

- Only LINES and POLYLINES in 2D are used. If you use Rhinoceros, "polyline" setting is preferred.
- Color scheme: Red=Mountain, Blue=Valley, Black=Crease. Layer color can be used instead of individual coloring.

- The software cannot recognize Group and Block Instance, thus all the line data should be ungrouped.
- The lines can intersect but cannot overlap (double line).

If chose to save in this format, its 2D crease pattern is saved.

OBJ Three-dimensional surface must be imported as a polyhedral mesh in obj file format.

- Surface must be singly covered and must be an orientable manifold, i.e., the edges must be shared by one or two facets with consistent orientation given by the counterclockwise order of vertices of the face.
- Be careful to "weld" the vertices. Usually, multiple vertices with the same coordinates are regarded as different vertices. The connectivity is given by vertex index of faces. For example, in **Rhinoceros**, every face must be joined, consistently oriented, and every edge should be welded.
- If there exist multiple vertices with the same coordinates (the distance is within certain threshold), the software asks you whether to join vertices. Joining vertices may result in non-manifold or non-orientable surfaces, and may lead to the software's death.
- imported OBJ file is usually without crease information. If you want to naturally assign Mountain and Valley according to the current state, use Model→Re-Assign MV.

The obj output from freeform origami includes comment lines with additional data used for freeform origami.

FOLD FOLD format by Erik Demaine, Jason Ku, and Robert Lang. `https://github.com/edemaine/fold` `http://jasonku.mit.edu/foldviewer/`

Currently, only 3D manifold folded form with single frame is supported. Face ordering is the key feature of this format, but freeform origami does not support (yet). Saving in FOLD format will put out additional data with namespace *ffo:*.

3.2 Add Reference

This operation appends point data from ***.dxf (2d drawing, i.e., crease pattern) or ***.obj (3d mesh folded form). File preparation is the same as the open command. This is intended to be used for reading reference geometry such as boundary condition and existing shapes, using *stitch command*.

3.3 Load Texture

For fun visualization, you can load an image of the texture of paper.

4 Constraints Menu

Constraints menu controls the constraints and how the software solves the constraints in the background.

Simulation Mode This check item set either the simulation mode or edit mode. See Section 2.4 for the difference between modes.

Constraints

Developable constraint forces the pattern developable, i.e., foldable from a piece of flat paper.

DevBoundaryNonOverlap constraint forces the boundary not to overlap

DevBoundaryAngle constraint forces the angles of the boundary unchanged.

DevBoundaryRigid constraint forces the model to be folded from the same size of piece of paper by preserving the length of edges of the paper boundary.

Globally Developable constraint forces the mesh to be developed to a piece of paper even if it is with holes.

Flat-Foldable constraint forces the pattern to be flat-foldable. Activate this constraints only after you assign proper Mountain and Valleys.

FlatBoundaryAngle constraint forces the angles of the boundary unchanged in the flat-folded state.

Globally Flat-foldable constraint forces the mesh to be flattened to a planar state even if it is with holes.

Avoid Collision (Edge) constraint forces mountain to be folded in mountain-wise, and valley in valley-wise.]

Avoid Collision (Vert) constraint avoids local collision between facets sharing a vertex]

Planar constraint forces each mesh to be planar (triangulation lines have the folding angle of 0). If this option is on when the 3D model is saved, the mesh is not triangulated, so quadrangles, pentagons, etc stay quadrangles, pentagons, etc.

Rigid Segment Rigidize the edges marked as rigid bars displayed as green segments. This constraint can work in the editing mode. Rigid bars are specified using Tool→Rigidize Edge.

Stitch This enables the stitch constraint (stitch constraints are created using Tool→Stitch)

Shaky This finds the form so that the first order folding mode exists. Can be used for generating shaky closed polyhedron.

Inequality Sector Angle Solves angle inequality condition for flat-foldability.

Triang Weight Defines the flexibility of the triangulating hinge.

Flat Limit This can limit the folding angle so that dihedral angle cannot be to small. This is used with the Avoid Collision option checked.

Set Iteration This defines the number of iteration of CG method in each projection step. If this is set to 1, the software works in steepest descent method, which is slow but sometimes stable.

Constraint On/Off Background calculation is enabled or disabled. Default: ON.

Adaptive Mesh Enables or disables adaptive mesh by merging close vertices.

5 ApplyToModel Menu

ApplyToModel menu is for performing action to the entire model. Undo/Redo is supported.

Set Length This command sets the length of the selected edges to the specified value, and make these edges rigid bars.

Add Ref Pt Create a point in 3D to which you can stitch vertices of mesh.

Stitch Overlapped Verts This command operates Tool→Stitch for close enough vertices and reference points so that they won't separate. Stitch constraint is activated by System→Stitch.

Auto Assign MV This command assigns Mountains and Valleys to the edges according to the current folding angles.

Pin Boundary Select all boundary points and set them fixed in space.

Pin Selected Pin fix selected vertices in space.

Origamize operation will insert "tuck" structure between faces to create an origami tessellations. The software automatically adds developability constraints, so that the shape may converge into a developable corrugated surfaces.

6 Fold/Unfold Menu

The menu have different tools of folding and unfolding shape change, controllable by "B" and SPACE key.

7 Tool

Select Used for selecting elements. Rectangle region selection is possible. (Left to Right and Right to Left has different selection, following conventional 3D CAD interface.)

Move This tool is for translating selected vertices by dragging. Following keyboard shortcut can be combined.

Shift Adding points to the selection.

x, y, z The transformation is constrained in x, y, and z directions.

Finger / Move Magnet This tool is similar to Move tool, but also affects neighbor vertices (in geodesic sense). Finger uses the dragging speed and Magnet uses the size of the window for the unit distance for defining the decay of effect. In the Finger mode, left dragging of the background rotates the whole object.

Assign This is for assigning mountain and valley to the edges. Chose the assignment from the palette and then click or region select edges that you want to "color."

Complementary foldlines try to unfold when foldlines fold, and fold when foldlines unfold. It is completely folded in the developed state, and is completely unfolded in flat-folded state. Using complementary foldlines along with normal foldlines with Planarity, Developability, Flat-foldability constraints is useful for making Bi-directionally Flat-foldable one-DOF rigid foldable surface such as egg-box surface or discrete Voss surface. See Tomohiro Tachi "Freeform Rigid-Foldable Structure using Bidirectionally Flat-Foldable Planar Quadrilateral Mesh" in Advances in Architectural Geometry 2010.

Flip Mesh Performs edge flipping.

Rigidize Edge This tool is for specifying which edges are rigid.

Stitch to Ground Set the selected vertices on the ground ($z = 0$ plane).

Non Dev Select singular points in which developability condition is not applied.

Non Flat Select singular points in which flat-foldability condition is not applied.

Stitch Vertices This tool is for "stitching" two vertices or reference points, so that their coordinates are the same. Select the first point and then select the second point. The segment between vertices represent the stitch constraint, clicking of which results in eliminating the constraint. Stitch constraint is activated by System→Stitch.

EqAngle Set selected edges to have the same folding angle.

EqLength Set selected edges to have the same length.

Unweld Selected edges will be unwelded, and the connectivity will change.

8 View Menu

This menu changes the visibility of elements.

9 Keyboard Shortcuts

This section lists the implicit keyboard commands.

Tab Key Temporarily switches Simulation mode and Edit Mode.

Control Key Temporarily makes the constraints soft (or slows down the solver), so that the shape acts softly.

Space Key Fold the model

'B' Unfold the model

'N' Add random white noise to vertices.

'F' Toggle full screen (Shift + F → to show in full screen with menu)

10 Acknowledgment

The development of the software is supported by a Grant in Aid for JSPS fellows (2008 April - 2010 March), and JST presto program (2010 October - 2016 March).

The software uses following open-source libraries:

wxWidgets `https://www.wxwidgets.org/`

wxWindows Library Licence, Version 3.1 `https://www.wxwidgets.org/about/licence/`

TinyXML2 by Lee Thomason (`www.grinninglizard.com`)

Original code by Lee Thomason (www.grinninglizard.com) This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software. Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions: 1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required. 2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software. 3. This notice may not be removed or altered from any source distribution.

JSON for Modern C++ by Niels Lohmann (`https://nlohmann.github.io/json/`)

MIT License

Copyright (c) 2013-2017 Niels Lohmann

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.