Proceedings of the International Association for Shell and Spatial Structures (IASS)
Symposium 2015, Amsterdam
Future Visions
17 - 20 August 2015, Amsterdam, The Netherlands

Optimizing the Steffen flexible polyhedron

Iila Lijingjiao [a]*, Tomohiro Tachi [b], Simon D. Guest [a]**

[a] University of Cambridge
Department of Engineering, Trumpington Street, Cambridge CB2 1PZ, UK
*iila.jingjiao.li@gmail.com
**sdg@eng.cam.ac.uk
[b] University of Tokyo
Department of General Systems Studies, 3-8-1 Komaba, Meguro-Ku, Tokyo 153-8902, Japan
tachi@idea.c.u-tokyo.ac.jp

Abstract

We revisit Steffen's known flexible polyhedron, originally described in 1978, and investigate whether we can increase its range of motion by varying his original dimensions. We also define the *regularity* of a polyhedron. Using a simulated annealing algorithm, we perform multi-objective optimization on the Steffen polyhedron to achieve both maximum regularity and range of motion. The results show that we are able to both increase the range of motion possible for the polyhedron, while still making the polyhedron more regular.

Keywords: polyhedron, rigidity, flexibility, optimization.

1. Introduction

Most polyhedra are rigid, but this paper considers an exception: the Steffen flexible polyhedron [9]. Here we show how its range of motion can be extended.

We use a concrete definition of polyhedra as closed surfaces consisting of a set of rigid polygons that are connected along their edges by hinges. Thus we consider a simply connected hollow object with an envelope of rigid faces. It was long conjectured (by Euler, among others) that all triangulated polyhedra are rigid (Connelly [3]). Such structures automatically satisfy Maxwell's equation for the rigidity of frames (Maxwell [6]), and it was proved by Cauchy [2] that all convex triangulated polyhedra are rigid. In 1897, Bricard [1] introduced three types of triangulated octahedra that are flexible; however, these are all self-intersecting, and hence are not 'polyhedra' by our definition. Later Gluck [4] showed that triangulated polyhedra are *generically* rigid, i.e., *almost all* such structures are rigid. However, Connelly showed, in the 1970s, by the simple expedient of a counterexample, that non-convex

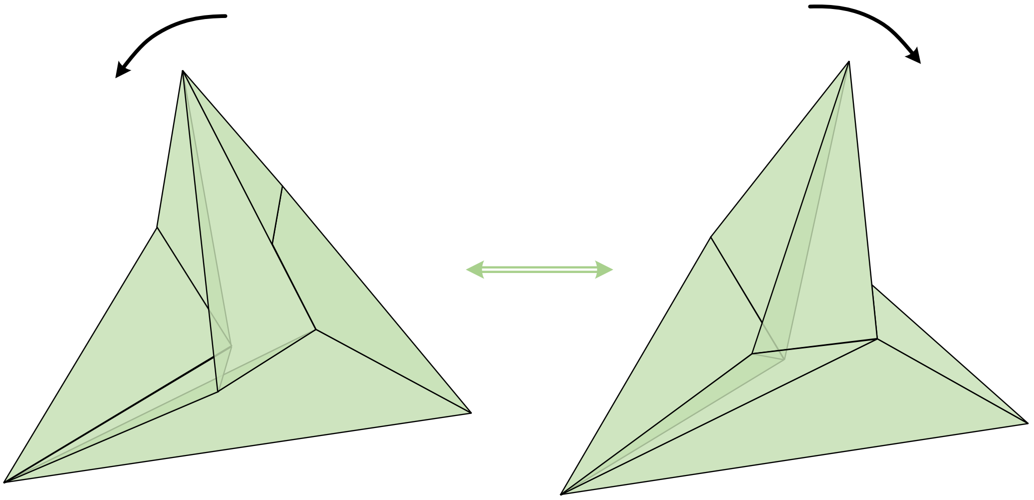

Figure 1: Two extreme positions during the flexing movement of the Steffen polyhedron. The faces are shown semi-opaque, so that hidden edges and vertices can be seen.

polyhedra *can* be flexible (Connelly [3]). Soon after, Steffen [9] described a much simpler flexible polyhedron consisting of 14 faces, 9 nodes and 21 edges, shown in Figure 1.

This paper revisits the construction of Steffen flexible polyhedron. It will consider how to optimize the range of motion for this polyhedron, while also considering its 'regularity'.

2. Composition of Steffen polyhedron

Figure 2 shows the composition of the Steffen flexible polyhedron. The feature of the polyhedron that allows flexibility is the use of 'crinkles', which were introduced by Connelly [3]. Figure 2(c,d) shows one such crinkle. The crinkle consists of 6 triangular faces joining 6 vertices. The crinkle has 11 edges, plus one *virtual* edge, whose length remains constant as the crinkle is flexed. The crinkle is formed by the elimination of two adjacent faces from a Bricard type I octahedron, which has a C_2 axis of symmetry. Since the crinkle is based on a Bricard octahedron, the crinkle can change configuration, or 'flex', without any changes in edge length – including the length of the virtual edge *EF*. The Steffen polyhedron has two crinkles that both share the same virtual edge, so the whole polyhedron can flex, as is shown in Figure 2(a). However, after some displacement, 'clashes' occur: constituent elements of the flexible polyhedron come into contact with one another. For instance, an edge within one crinkle might touch an edge in the other crinkle, or two faces of one crinkle might touch each other.

Figure 2(b) shows the net of the Steffen polyhedron, which can be used to make a model. The net shows five length parameters defined by Steffen. He suggested giving them the relative values $a = 6$, $b =$

5, $c = 2.5$, $d = 5.5$, $e = 8.5$. In Section 3, we will vary these values, but will use Steffen’s suggestion to provide us with a valid starting configuration.

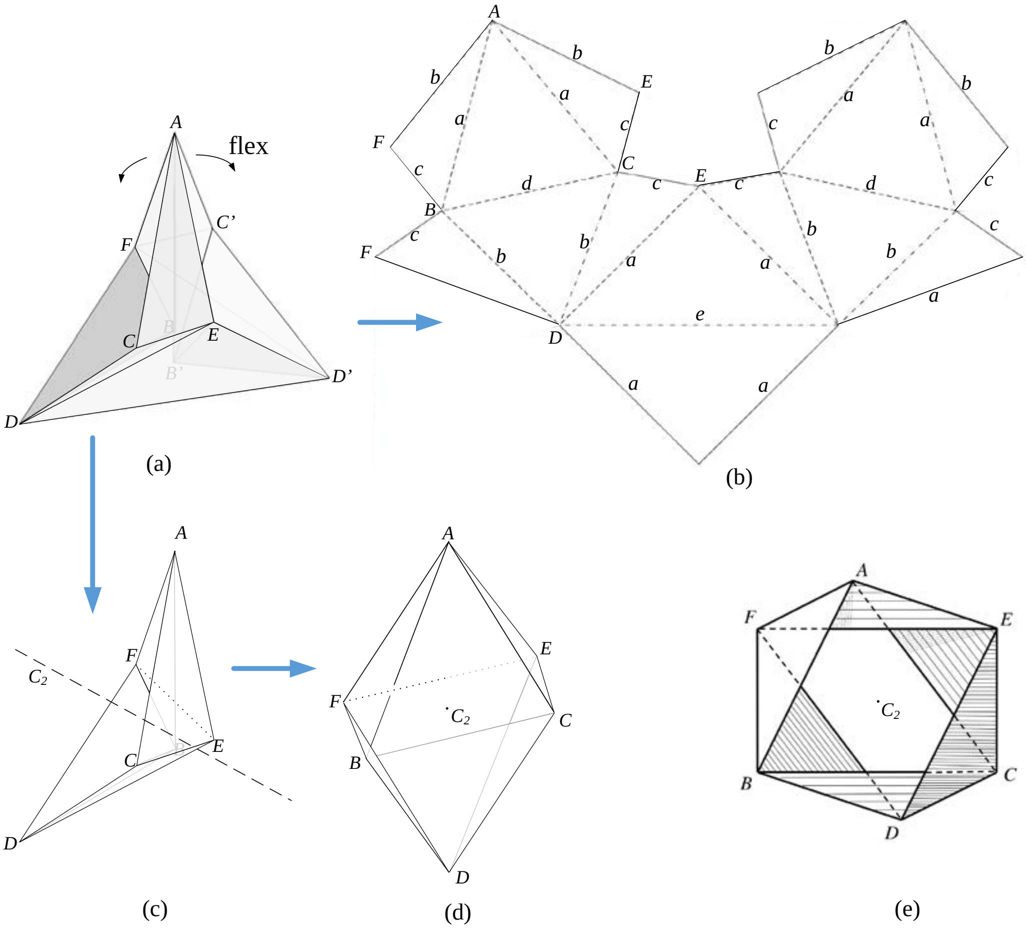

Figure 2: Construction of the Steffen flexible polyhedron. (a) A view of the Steffen polyhedron, showing the flexibility. Point *A* is able to rotate around the virtual hinge *EF*. (b) A net from which the Steffen polyhedron can be folded – 'mountain' (dashed) and 'valley' (dash-dot) lines are distinguished from each other. The net shows five length parameters *a*–*e*. (c, d) Two views of part of the Steffen polyhedron showing one 'crinkle' (Connelly [3]) which forms a Bricard Type I octahedron. The view point in (c) is the same as the view point in (a); the view in (d) is along the marked C_2 axis of the octahedron, which is to be compared with the Bricard octahedron Type I in (e). The virtual hinge *EF* is dotted. (e) Bricard's original figure of a Type I octahedron (taken from Bricard [1]), to which we have added the position of the C_2 axis which is perpendicular to the plane of the drawing. The octahedron in (d) is a Bricard octahedron (e), but with different dimensions.

3. Optimization

3.1. Optimization overview

We wish to adjust the values of the parameters *a–e* to give a polyhedron with as large a flex as possible. Specifically, we wish to find the Steffen polyhedron which allows the largest possible rotation of point *A* around the virtual hinge *EF*, which we define as the 'range of motion' Θ. Because we are not interested in the overall scale of the polyhedron, we fix the value of *e*, and thus have four parameters whose values we allow to vary. However, when we first considered this, we quickly discovered that an optimization process to increase the range of motion drove the crinkles to become smaller and flatter. To avoid generating polyhedra that are very distorted, we have defined an additional measure for the 'regularity' *R* of the polyhedron, which we define as being the ratio of the radius of the smallest inscribed circle (the inradius) for any face to the radius of the largest circumscribed circle (the circumradius) for any face – see Section 3.5. We then carry out a multi-objective optimization of both the range of motion and the regularity.

In the rest of this section, we provide essential technical details that enabled us to carry out the optimization.

3.2 Optimization method

We use the Simulated Annealing (SA) method (Kirkpatrick [5]) for the optimization. To do this, we defined a function in Matlab to be used within a SA algorithm. The variables passed to the function are *a–d* (with *e* fixed at 8.5) and a chosen maximum rotation of point *A* around the virtual hinge *EF*. Because the whole polyhedron is line symmetric, the range of motion Θ is twice this maximum rotation. The function finds the configuration of the polyhedron for the maximum rotation and detects any clash within the configuration. We define a penalty function to penalize any clash, and the function returns the range of motion penalized by any clash detected.

To carry out the simulated annealing, we used the 'General simulated annealing algorithm' available from the online resource Matlab Central (Vandekerckhove [11]). Each SA run returned a set of parameters that locally maximised Θ, and for which *R* could then be found.

3.3 Finding configurations

The function we use for optimization first finds a *configuration* of the polyhedron, where a configuration is defined by the position of all nine vertices. The parameters *a–e* define the positions of vertices *D*, *D'*, *E* and *F*, shown in Figure 2(a). Together with the maximum rotation parameter, the position of vertex *A* is also fixed. What is required to complete the configuration is the positions of four other vertices, *B* and *C* in one crinkle, and the corresponding nodes *B'* and *C'* in the other crinkle.

Initially the four vertices *B*, *C*, *B'*, *C'* are given a position that roughly approximates their likely position, and this position is then corrected to the actual configuration. To do this, the 21 edge lengths in the current iteration *i* are calculated and used to define a vector $\mathbf{L}_i$. The required correct lengths define a vector $\mathbf{L}_c$, each of whose entries are one of the values of the parameters *a–e*. The current *extension* is then given by $\mathbf{e}_i = \mathbf{L}_i - \mathbf{L}_c$. We then correct for this extension by using a linearized compatibility matrix (Pellegrino [7]) as follows.

At the current position, we consider the Steffen polyhedron to be a truss structure, in which nodes *A*, *D*, *E*, *F* and *D'* are fixed to a foundation, while four nodes *B*, *C*, *B'*, *C'* are able to be displaced. Thus we have a truss structure with has $j = 4$ non-foundation nodes and $b = 21$ bars, for which we can write a compatibility matrix $\mathbf{C}_i$ for the current iteration to correct for the extensions (i.e. we try to find **d** to impose a change of length –**e**).

$$b\left\{\vphantom{\begin{matrix}\\ \\ \\ \\ \\ \\ \end{matrix}}\right.\underbrace{\begin{bmatrix} & & \\ & & \\ & \mathbf{C} & \\ & & \\ & & \end{bmatrix}}_{3j}\cdot\left.\begin{bmatrix}\\ \mathbf{d}\\ \\ \end{bmatrix}\right\}3j=\left.\begin{bmatrix}\\ \\ -\mathbf{e}\\ \\ \\ \end{bmatrix}\right\}b \qquad (1)$$

In fact, seven of the truss members are guaranteed to have the correct length, because they span fixed nodes,

$$\mathbf{e}=\begin{bmatrix}\times\\ \times\\ \times\\ \times\\ \times\\ \times\\ \times\\ 0\\ \vdots\\ 0\end{bmatrix}\begin{matrix}\left.\vphantom{\begin{matrix}\times\\ \times\\ \times\\ \times\\ \times\\ \times\\ \times\end{matrix}}\right\}14\ \text{ extensional bars}\\ \left.\vphantom{\begin{matrix}0\\ \vdots\\ 0\end{matrix}}\right\}7\ \text{ nonextended bars}\end{matrix}$$

Equation (1) is overconstrained. We find a minimum length least squares solution using the pseudo-inverse $\mathbf{C}^{+}$ (Strang [10]) to give nodal corrections for iteration *i*, $\mathbf{d}_i$. The nodes are displaced by the calculated amount, and then iteration *i* is complete. If the extensions are then small enough then the nodal positions are taken to give the target configuration. In practice, it typically takes four iterations to give a configuration with resultant extensions of the order of 10^{-4}.

3.4 Clash detection

Once we know the configuration, we need to ensure that no 'clashes' have occurred, where, for instance, an edge within one crinkle has passed through an edge in the other crinkle, or two faces of one crinkle have passed through each other. In the Steffen polyhedron, six possible clashes can occur. Each of them can be detected by considering the relative orientation of the relevant line and face.

Consider, as an example, the potential clash between line *FB* and face *DD'F*, shown in Figure 3. We calculate a unit normal **n** to the face *DD'F*, and find the dot product with the vector $\overrightarrow{FB}$. If this is negative, a clash has occurred. The magnitude of the clash is defined as the angle β shown in Figure 3(a). We define the penalty *p* associated with a clash as

$$p(\beta) = \begin{cases} T\left(e^{-\frac{\beta}{T}} + \frac{\beta}{T} - 1\right), & \beta < 0 \\ 0, & \beta \geq 0 \end{cases}$$

where *T* is a parameter that can be varied. In fact, we used the 'temperature' parameter in the simulated annealing algorithm as the parameter *T* to ensure that the penalty became stricter as the simulated annealing converged on a final optimum. The penalty used in the optimization is the largest penalty associated with any of the potential clashes.

3.5 Definition of regularity

As well as optimizing the range of motion *Θ*, we also consider the *regularity R*, which we define as follows. For any polyhedron, *R* is the ratio of the radius of the smallest inscribed circle (the inradius) for any face to the radius of the largest circumscribed circle (the circumradius) for any face. Thus, if we number the faces 1 to *n*, and define the inradius and circumradius for face *j* to be $R_i(j)$ and $R_c(j)$ respectively, we have

$$R = \frac{\min\limits_{j=1\cdots n} R_i(j)}{\max\limits_{j=1\cdots n} R_c(j)}$$

Note that the maximum possible value of *R* for any triangulated polyhedron is 0.5, when all faces are equilateral triangles, for instance, for a regular tetrahedron.

As an example, Figure 4 shows the parameters $R_{i,\,\min} = \min\limits_{j=1\cdots 14} R_i(j)$ and $R_{c,\,\max} = \max\limits_{j=1\cdots 14} R_c(j)$ for the net of the Steffen polyhedron.

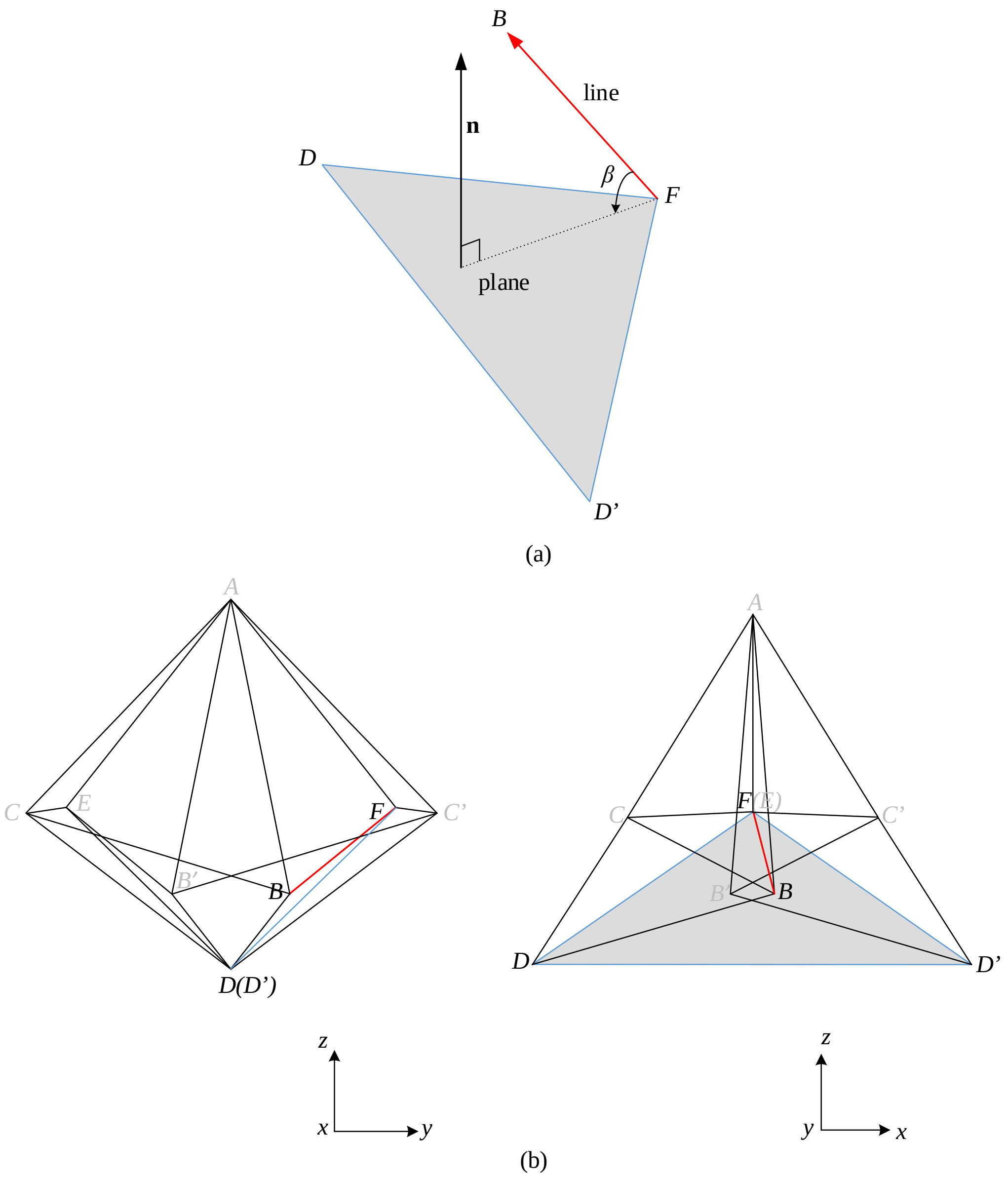

Figure 3: Clash detection between a line *FB* and face *DD'F*. (a) Schematic representation of the line, the face, and a normal to the face. (b) Two views of the Steffen polyhedron, showing the actual disposition of the line *FB* and face *DD'F*. The polyhedron is in neutral position, so there is no clash. But if the polyhedron is flexed in one direction, the angle between the line and the face decreases. If the angle becomes negative, a clash is said to have occurred.

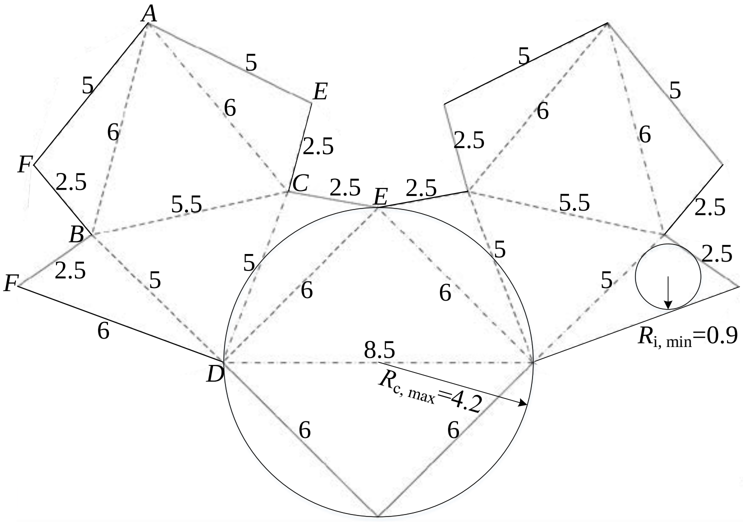

Figure 4: The regularity of the Steffen polyhedron constructed with Steffen's recommended parameter values is given by R=$R_{\text{i,min}}$/$R_{\text{c,max}}$=0.214

4. Multi-objective optimization result

Formally, we define our optimization process as follows. For four geometric parameters,

$$\mathbf{x} = [a\ b\ c\ d]^T$$

we wish to maximize

$$f(\mathbf{x}) = w_1\Theta(\mathbf{x}) + w_2R(\mathbf{x}) \qquad (2)$$

subject to the following constraints on the weights w_1 and w_2:

$$w_1 \geq 0;\ w_2 \geq 0;\ w_1 + w_2 = 1$$

For our results, we describe the Pareto front that captures all possible maximal values of $f(\mathbf{x})$ for any choice of the weights w_1 and w_2.

In practice, we generated results as follows. Starting from the initial parameters given by Steffen, the SA method was run a number of times to generate a sequence of results. The parameters within the SA method (e.g., the 'cooling' rate) were varied to give other sequences, and small random perturbations of parameters were used to give other starting points. Sometimes targeted changes in parameters were used to ensure an increase in the regularity, and sometimes new configurations were only accepted if they did not decrease the regularity. By this method, we generated a cloud of possible configurations, each of which has a range of motion Θ, and for which R is defined. This cloud of points, and the Pareto front that it defines, is given in Figure 5.

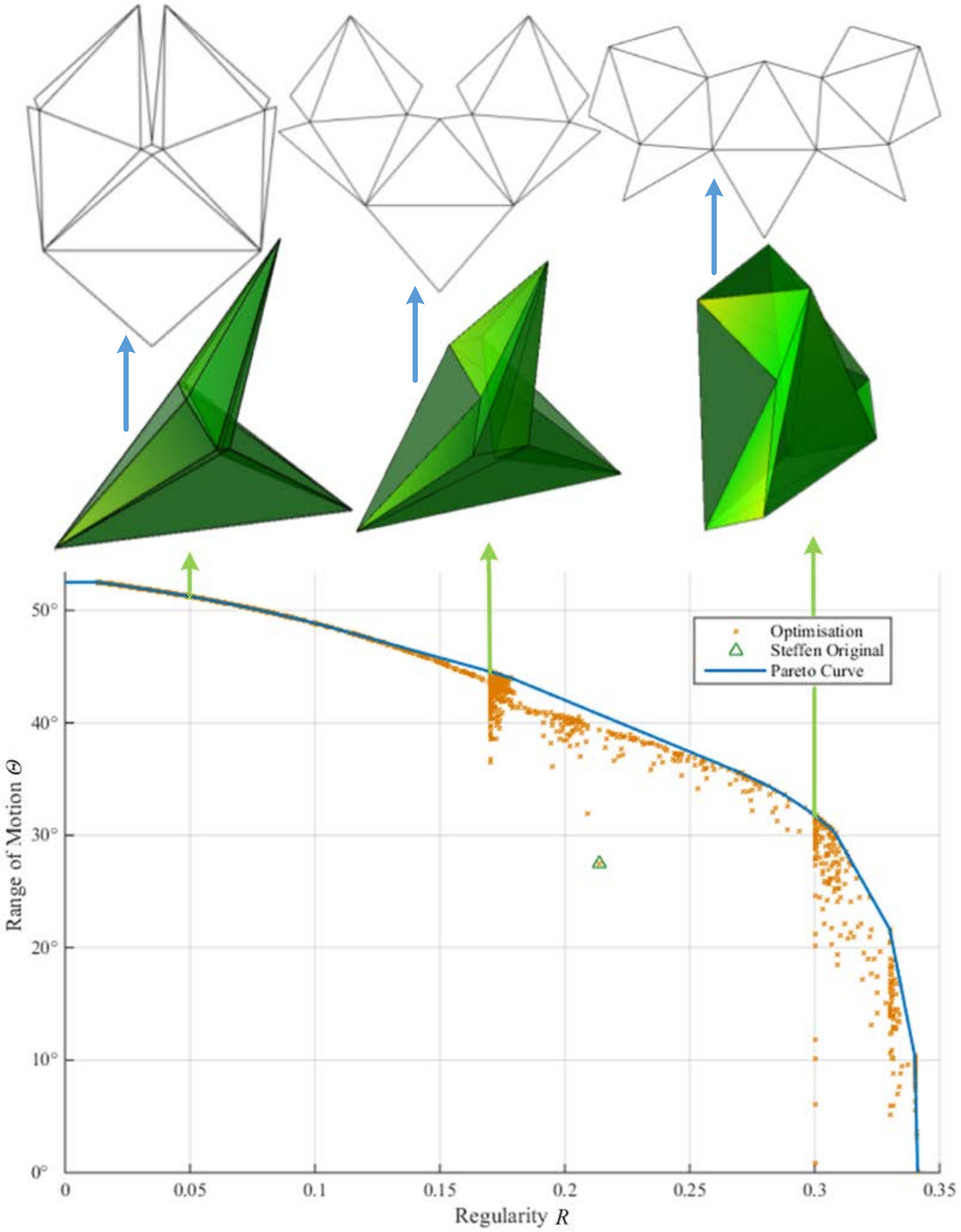

Figure 5: Multi-objective optimization of the Steffen polyhedron, simultaneously maximizing range of motion Θ and regularity R. Each cross gives one simulated annealing result. The green triangle shows Θ, R for the original Steffen polyhedron. Around the optimization results the convex Pareto front is plotted, giving the maximal values of the function $f(\mathbf{x})$ defined in equation (2). Three results that lie on the Pareto curve have been chosen to illustrate optimal polyhedra: for each of the results the net and a configuration are shown.

Comparing the results with the original Steffen parameters, we can see that it is possible to generate highly irregular polyhedron with twice the range of motion, and also a just flexible polyhedron which is 50% more regular. More importantly, we have shown that there are many sets of parameters which give an increased range of motion with little or no detriment to the regularity of the polyhedron, and indeed that it is possible to increase *both* the regularity *and* the range of motion simultaneously by around 25%.

4. Conclusion

This research has shown that with modern computational tools, we are able to increase the range of motion possible for the Steffen polyhedron as well as the regularity.

Further work will explore how the concept of a crinkle based on Bricard octahedra can be generalized to provide additional parameters for Steffen's polyhedron. We will also look to develop new examples of flexible polyhedra and consider engineering applications.

References

[1] Bricard, R., Memoir on the Theory of the Articulated Octahedron. English translation published as *arXiv preprint arXiv:1203.1286*. 2010, translated by Coutsias E.A from the French original Mémoire sur la théorie de l'octaèdre articulé, *J.Math.Pures Appl*. 1897, 3, 113-150.

[2] Cauchy A.L., Sur les polygones et les polyèdres, Second Mémoire, *Journal de l'Ecole Impériale Polytechnique*, XVI Cahier, 9 (1813) 87-99 (Oeuvres Complètes d'Augustin Cauchy, 2e série, Tome 1 (1905) 26-38), referenced by Connelly [3] and by Servatius [8].

[3] Connelly, R., The rigidity of polyhedral surfaces. *Mathematics Magazine*, 52(5), 1979, 275-283.

[4] Gluck H., Almost all simply connected closed surfaces are rigid, *Lecture Notes in Math*., 438, *Geometric Topology*, Springer-Verlag (1975) 225-239, referenced by Connelly [3] and by Servatius [8].

[5] Kirkpatrick, S., Gelatt C.D. and Vecchi, M.P. Optimization by Simulated Annealing. *Science*, 220, 1983, 671–680

[6] Maxwell J. C., On the calculation of the equilibrium and stiffness of frames. *The Philosophical Magazine*, 1864, 598–604.

[7] Pellegrino S., Structural computations with the singular value decomposition of the equilibrium matrix. *International Journal of Solids and Structures*, 30(21), 1993, 3025–3035.

[8] Servatius B., Combinatorics and the rigidity of frameworks. *Newsletter of the SIAM Activity Group on Discrete Math*, 4, 1993, (1), 1–5.

[9] Steffen, K. *A symmetric flexible Connelly sphere with only nine vertices*. Letter written to Institut des Hautes Études Scientifiques, 1978 (available online at http://www.math.cornell.edu/~connelly/Steffen.pdf, downloaded 30 March 2015).

[10] Strang G., Linear Algebra and Its Applications. (4th ed.), Thomson Higher Education, 1988.

[11] Vandekerckhove, J. *General simulated annealing algorithm*. A Matlab function downloaded 15 March 2014 from http://mathworks.com/matlabcentral (2008).