== Draft for Tech Report: 1/26/2017 ==

Modular Toroids Constructed from Nonahedra

Carlo Séquin, Yifat Amir, Ruta Jawale, Hong Jeon, Rohan Taori
CS Division, University of California, Berkeley
E-mail: sequin@cs.berkeley.edu

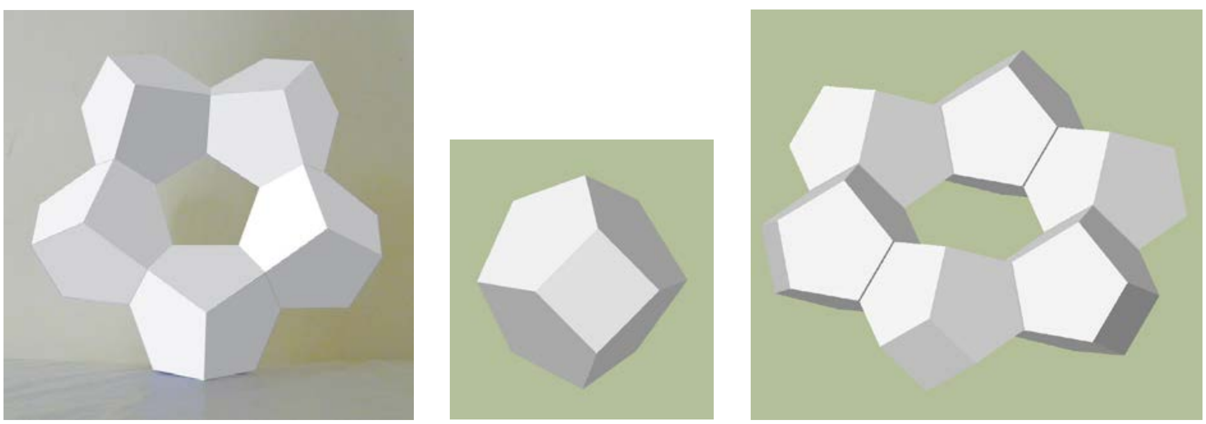
Abstract

Inspired by a piece of artwork at Bridges 2016, we were looking for various ways of making modular polyhedral toroids from the same basic nonahedron building block. Since the chosen building block cannot be an ideal, regular Johnson solid, we used greedy optimization to minimize the variations of edge lengths and the deviation from regular *n*-gons for the facets of these various constructions. In general, we tried to maximize overall symmetry.

1. An Intriguing Puzzle

Figure 1(a) shows a sculpture by Bente Simonsen that was exhibited at the Bridges 2016 Art Exhibit [1]. At first glance, it seems to be nothing very special. But it gets much more interesting after one takes a closer look and reads the accompanying cryptic description:

"*The sculpture is created from five nine-sided polygons, each with 3 square sides and 6 pentangle sides, all with the same side length, and fitted together to build a pentagonal torus.*"



(a) (b) (c)

Figure 1: *(a) Bente Simonsen:"Torus" [1]; (b) nonahedron building block; (c) 6-module toroid.*

To a geometrically astute observer, a couple of intriguing puzzles present themselves. Figure 1(b) shows what the implied nonahedron module might look like.

A check in Wikipedia [2] reveals that this *truncated triangular bipyramid* is a "near-miss" Johnson solid. Johnson solids are convex polyhedra made from all completely regular n-gons. But the shape in Figure 1(b) cannot be constructed from regular pentagons and square faces. Either the edges need to be of somewhat different lengths, or some faces need to be irregular or even non-planar.

Moreover, the building block seems to have 3-fold rotational symmetry, and this implies that the dihedral angle between pairs of square faces must be 60°. This further implies that six, rather than five, of those modules would readily form a modular, 6-fold symmetrical, toroidal ring (Fig.1c). This then raises the question, whether all faces in this toroidal shape can be made planar, when all the edges are constrained to be of equal length. We analyze the degrees of freedom (DOF) of the shapes shown in Figure 1 and find out, how close to a perfect Johnson solid the building-block polyhedron can be made.

2. Variations of the Nonahedron (Enneahedron) Module

Let's first explore some design options for the nonahedron (9-faced) module.

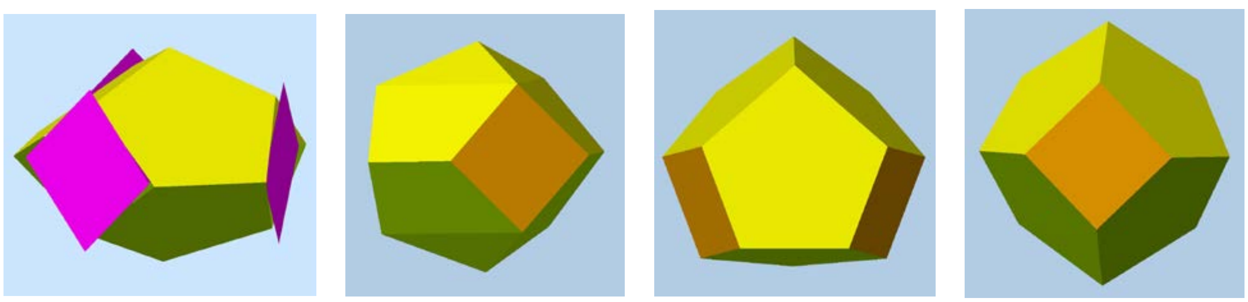
A: We could start with the corner of the pentagonal dodecahedron consisting of 3 regular pentagons, and put two of them together so that three edge pairs fuse (Fig.2a); but this results in non-planar quad faces.

B: Next we start with 3 perfect unit-square faces and place them symmetrically around the *z*-axis, so that the 3 pairs of closest corners are a unit length apart. Subsequently we choose a point on the *z*-axis that is exactly one unit-length away from the upward-pointing corners of the square faces, and we use this vertex to complete the 3 yellow pentagons. We also apply the same construction at the bottom (green pentagons). This construction leads to non-planar pentagons (Fig.2b).

C: A third option is to enforce planarity by raising the tips of the 3-sided top and bottom pyramids to an appropriate height. But in doing so, we have lengthened these pyramid edges considerably (Fig.2c).

D: Since the pentagons are no longer regular, what can we gain, by also allowing the quads to become rhombi? Can we now make all faces planar <u>and</u> keep all edges the same length?

E: Perhaps the best compromise lies in a mixture of all these approaches, where we allow some violation of all the constraints of a perfect Johnson solid. Hopefully, all these violations can then be kept small enough, so that they are not immediately visible in any display of this polyhedron (Fig.2d).



(a) (b) (c) (d)

Figure 2: *Various nonahedral modules: (a) using 6 regular pentagons, (b) using 3 planar squares and all equal edge lengths, (c) all planar faces, (d) a visually "optimized" module.*

To study what is possible and to find a combination of small violations that gives the best overall result, we need to introduce an efficient parameterization for the geometry of this solid and devise a tunable "cost"- (or "penalty"-) function that will allow us to measure the deviation from perfection.

Parameterization

Our building block has 14 vertices, and so we have to set the values of 42 coordinate components. But doing the needed optimization in 42-dimensional space would be inefficient. We can get a complete geometry description with much fewer parameters, if we maintain strict 12-fold "D_{3h} symmetry" (Schönflies notation), also known as "*223" symmetry (Conway notation), composed of a 3-fold rotation axis (the *z*-axis), and horizontal and vertical mirror planes.

Since our optimization is scale-independent, let's start by making the 3 equatorial edges of length 1.0 and placing them into the *x*-*y*-plane at a distance, *ex*, from the origin. The top and bottom pyramid tips lie on the *z*-axis, and they will be placed at a height of ± *tz*. The "corner vertices" at the "shoulders" need two parameters: a radial distance, *rc*, and a height, *hc*. These four parameters are sufficient to capture all variations of our 12-fold symmetrical building block; and any optimization can also be carried out in this 4-dimensional search space.

Cost Function

Our penalty- (cost-) function will have 3 major components, measuring separately the differences in edge length, the deviations from the ideal internal polygon angles, and the degree of non-planarity for all faces.

Edge-Length Differences (ELD)

We determine the average length, *avl*, of all 21 edges, and then calculate the sum of squares of all the deviations from this average length for all edges.

$avl = \sum (\text{edge-length}_i) / 21$;

$ELD = \sum (\text{length}_i - avl)^2$

Polygonal Angle Deviations (PAD)

We measure the internal angles at all corners in all faces, and then calculate the sum of squares of all the deviations from the corresponding angles in a regular *n*-gon.

$PAD\ (\text{squares}) = \sum (\text{angle}_{s4} - 90)^2$; $PAD\ (\text{pentagons}) = \sum (\text{angle}_{p5} - 108)^2$;

$PAD = PAD\ (\text{squares}) + PAD\ (\text{pentagons})$

Face Non-Planarity (FNP)

For each face we find the best-fitting plane, and then calculate the offset of the face-contour vertices from this plane, and sum the square of these distances.

Calculate the best fitting plane-normal for a polygon [3].

Move this plane parallel to itself, so that it passes through the centroid of all the face vertices.

dist = {insert vertex coordinates into above plane equations}

$FNP = \sum (dist_i)^2$

Weighted Combination

The above three terms can be combined into a single cost function, where each component is given some weighting coefficient that expresses the "visibility" of the corresponding type of deviation from the ideal. For instance, if we do not care about the regularity of our faces, we can simply set *beta* = 0.

$COST = alfa * ELD + beta * PAD + gamma * FNP$

Optimization

By gradient descent in the 4D space {*ex*, *tz*, *rc*, *hc*}, we determine parameter values that minimize *COST*, and thus find the "nicest" nonahedron building block. If the quad faces are kept planar, 6 units of this 3-fold symmetrical building block will readily assemble into a 6-module toroid with D_{6h} symmetry (Fig.1c).

Practical Considerations

To make the computation of the cost function as efficient as possible, we exploit the symmetry of this polyhedron.

Edge-Length Differences (ELD)

There are only three different types of edges. One type (lying in the *x*-*y*-plane) has been set to always be of length 1.0. Thus, rather than measuring the deviations of the edge lengths from their overall average, we might simply measure their deviation from the ideal 1.0 edge-length. We only have to do this calculation for two edges: an edge used by the quad faces, and one edge ending in a vertex lying on the *z*-axis. Since there are 12 edges of the first kind, but only 6 edges of the second kind, we will weigh the measured deviations from the ideal unit length in the ratio 2:1. However, this edge-length equalization does not yield the absolute minimal cost, when the unit-length edge does not have the median of the three edge types.

Polygonal Angle Deviations (PAD)

The ideal internal angle for the quad faces is 90°, and for the pentagons it is 108°. We can find the actual angles by forming a dot-product of two directional unit vectors for two edges sharing a vertex. But there is no need to actually convert the value of this dot product into an angle. The dot product of two vectors at right angles to one another is zero, and thus we can use the value of the dot product directly to serve as a measure of the deviation from 90°. We only have to determine this deviation for one corner of one quad face; because of the overall symmetry imposed, the absolute value of the deviations of the other three corners will be the same as long as the face is close to being planar.

Similarly, the dot product corresponding to an angle of 108° is 0.309; and (small) deviations from this value are good enough to measure the deviation from the ideal pentagon angle. Three different values for these angles may appear in each pentagon. They should be weighed in the ratios 2:2:1, with the last one being the value measured at the vertex lying on the z-axis. These deviation values are then combined with the value from the quad face to form the overall cost function component *PAD*. Since our visual system is more trained at recognizing right angles, we might weight the quad-face angle deviation a little more strongly than the deviations in the pentagonal face.

Face Non-Planarity (FNP)

For the individual nonahedral building block, the non-planarity test is applied to one quad face and to one pentagon, and these measures are weighed in the ratio 1:2. For some of the toroid assemblies, we may constrain certain faces to be planar by construction.

Display of the Current Penalty Value

To get an intuitive understanding of the dependency of the cost function on the four parameters of the nonahedron building block, we want to integrate a display of the current values of its three components into the display of the current model geometry. As we adjust the values of some of the defining parameters, we can then readily see its influence on the cost function. This should help us to find useful values for the three weighting factors, *alfa*, *beta*, and *gamma*, that yield a good balance between the three rather different measures. It may also be helpful to display this information not just as three numeric values, but also as three differently colored bars that provide immediate visual feedback.

3. Gradient Descent

With a good cost function in place, we are now ready to tackle the task of minimizing this cost function via gradient descent. The principle is simple. For each independent parameter, we look at the effect that a small change in this parameter value has on the overall cost function. The test step that we take to determine the sensitivity of a parameter should be small enough, so that we are still in a mostly linear range in our cost-function landscape. On the other hand, it should not be so small that the measurement becomes unreliable because of numerical inaccuracies and rounding errors.

Once we know the sensitivities for all the individual parameters, we combine them into a resultant vector that includes each parameter with a weight that is proportional to the contribution that this parameter can make to reduce the overall cost function. This defines the direction of the local gradient, which gives us the steepest down-hill vector in the landscape of the cost function in the multi-dimensional parameter space of our optimization problem. We now let our system take a small step in this gradient direction.

It is not immediately clear what is the best step size; so we want to make sure that this is easily adjustable. When the system just does not want to move, we double the step size; when it behaves erratically or oscillates back and forth, we reduce the step size. We may start the optimization with a relatively large step size, but then reduce it as we approach a (local) minimum, so as not to overshoot our target, which is the lowest point in this local cost-function bowl. It is desirable that these adjustments can be made readily from the keyboard while the program is running.

Eventually, one may want to add extra code to set this step size adaptively to a reasonable value. We may do this by running tentative optimization steps with different step sizes. We may increase the step size until we find that the response in the reduction of the cost function deviates strongly from a more or less linear behavior and then back up to a smaller, safe step size.

We also have to control the values for *alfa*, *beta*, and *gamma*. Originally these values may be set roughly to yield about equal sensitivities with respect to the three different components of the cost function. But in the end, they will have to be fine-tuned by the user to yield the most preferable aesthetic result. The tastes of different users may vary, and the sensitivities for different realizations of the nonahedron may be different. For instance, if the nonahedron has a mirror-like finish, then any non-planarity of its faces would be much more apparent.

Some Optimal Parameter Settings and Corresponding Results

Table 1 shows the parameter values for various "optimized" nonahedra. The first one (by Séquin) was optimized based on visual inspection of an interactive display, while adjusting the four defining parameters manually. The others were optimized by gradient descent with different settings for *alfa*, *beta*, and *gamma*. The main difference seems to be that in the visually optimized shape the value *tz* is considerably larger. This improves the planarity of the pentagons, but increases the variations in the edge lengths.

Table 1: *Optimized Parameter Values for the Isolated Nonahedron:*

param.	Séquin	Jeon	Taori	Jawale	Amir
ex	1.111	1.146	1.104	1.070	1.045
rc	0.990	0.988	0.931	0.999	0.895
hc	0.750	0.741	0.707	0.679	0.642
tz	1.333	1.252	1.072	1.030	1.012

Séquin Jeon Taori Jawale Amir .

Figure 3: *The optimized nonahedra displayed with identical viewing parameters.*

4. Constrained Geometry Optimization

If we let some components of the cost-function dominate the others, we may end up with a fully constrained system with no remaining degrees of freedom. One such example is shown in Figure 1(b) where all edges were forced to be of equal lengths. Instead, planarity of all faces could be chosen as another dominant constraint. This would force us to make the top and bottom edge triplets somewhat longer, leading to noticeably stretched pentagons.

Choosing to make the quad faces perfect squares already prevents the pentagons from being regular. It locks in the two pentagon angles near the equator to about 110.70° degrees. It also forces the pentagons to have a slope of 49.11° against the horizontal symmetry plane. This slope further forces the top angle of the pentagon to be 97.2° in order to form a water-tight corner. Thus, all the angles in the pentagon are determined by its planarity constraint. The only degree of freedom left is the distance, *es*, at which we place the perfect unit squares, and we use this distance as the only parameter for optimization. The value *es* then defines the length, *ee*, of the equatorial edge, as well as *ex*, its distance from the center. We can now choose either to minimize the variation in edge length or to force all vertices of the pentagon to lie on a circle, thus providing some form of "pseudo-regularity." It turns out that trying to get all vertices on a circle results in *ee* = 1.41 and pyramid edges of similar length. It seems preferable to either keep *ee* at 1.000, or make it even shorter. The variation in edge length is minimized for *ee* = 0.931. It results in the following parameter values: *ex* = 1.108, *rc* = 0.946, *hc* = 0.707, and *tz* = 1.253. This is rather close to the values in the first two columns in Table 1

5. Constructing Toroidal Rings

Next we will use nonahedra as building blocks for the construction of various toroidal configurations. We make small changes to the shape of the nonahedron, so that several identical copies can be assembled seamlessly into different symmetrical toroidal rings. The symmetry of the anticipated structure will define new constraints for the individual nonahedron module.

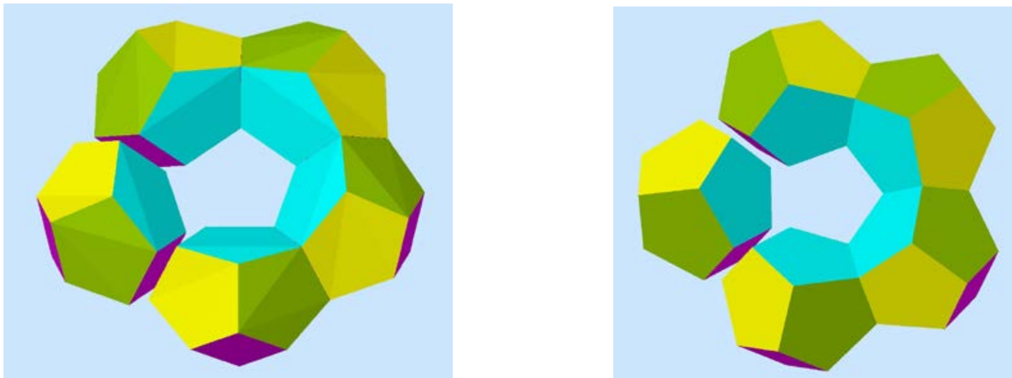
6-Ring

A ring composed of six nonahedra is the most natural assembly (Fig.1c). Because of the 3-fold rotational symmetry of the building block, pairs of quad faces occur with a dihedral angle of 60° between them. In order to glue these building blocks together with their quad faces, we make these faces planar. This can readily be achieved by calculating *rc* to lie directly above the middle of the diagonal of the quad. However, *hc* could still be retained as an independent parameter, if we allow the quad faces to assume a slightly rhombic shape. This extra degree of freedom (DOF) can be used to even out edge-length variation or to improve the planarity of the pentagons. <<Discuss some resulting parameter values>>

5-Ring and 7-Ring

Next, we try to arrange five of these nonahedra into a 5-fold symmetrical toroidal ring to emulate Bente Simonsen's design [1]. This is not possible without introducing some further distortions that break the 3-fold symmetry of the individual modules. The angle between adjacent quadrilateral contact faces has to be changed from 60° to 72°. In doing this, we can follow the implied recipe given by the original artist, (Section 1), and keep all the quad faces as perfect squares and then aim to make all edges of the same length. There are just enough DOFs to do this. We start by forming the tunnel of the toroid as a regular pentagon with unit-length edges. To this we attach five vertical squares radiating outwards with angles of 72° between them. Then we add another five vertical squares in a more tangential direction to the emerging torus and move them radially outwards until the outer edges between them, which lie in the equatorial plane, also acquire unit length. This now defines the overall geometry, except for the 10 top and bottom pyramid vertices. The position of these vertices is found by putting them at a unit distance from the three vertices to which they connect. The result is shown in Figure 4(a). The drawback is that the pentagons are noticeable non-planar. This becomes particularly noticeable if this sculpture is fabricated with a shiny, reflecting surface.

Just as in the case of the individual nonahedron, a tradeoff can be made between face planarity and edge-length uniformity. Since the slope of the inner pentagons is determined by the inner edges of the five quad contact faces, a planarity constraint defines the ratio between the height and the radial distance of the pyramid vertices as measured from the inner equatorial edges. Three are now three DOFs left: the length, *ie*, of the inner equatorial edges; the radial distance, *fx*, of the outer quad face, which implicitly defines the length, *oe*, of the outer equatorial edges; and the height/distance of the pyramid vertices. Rendering the outer pentagons planar, uses up two of these DOFs (the first three vertices of a pentagon define a plane that places one positional constraint each on the other two vertices). The last remaining DOF can be used to minimize the variations in the edge lengths. The result is shown in Figure 4(b).

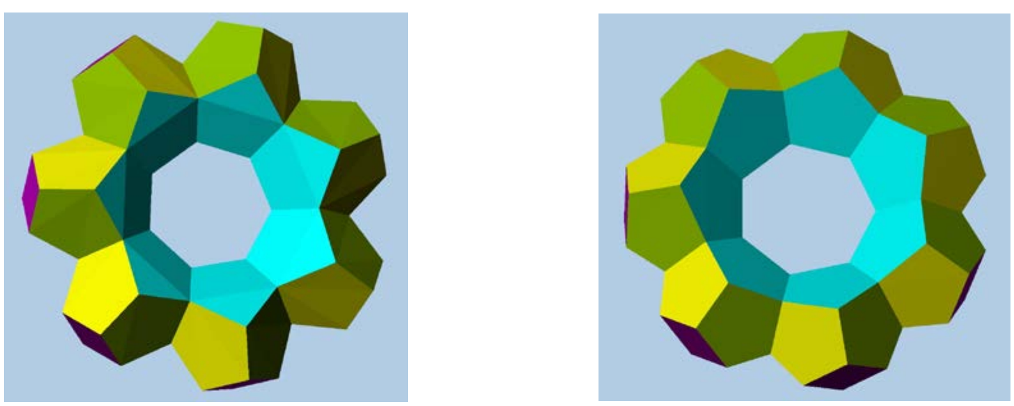


(a) (b)

Figure 4: *"Unnatural" toroidal assembly using five nonahedron building blocks; (a) using perfectly square quads and all equal edge lengths; (b) keeping all pentagons planar.*

Given the objective of finding the visually most pleasing toroidal structure, we can take into account that not all deviations from regularity are equally visible. While a deviation from a perfect square on the outer quad faces would quickly become noticeable, the exact shape of the contact quads is much less critical. In particular, any deviation of the corner angles of these quads is very hard to judge, and even deviations in edge length in these quads are less noticeable. Thus we can forego the original constraints placed on these quads, and use the additional DOFs gained to further increase the edge-length uniformity of the other, more visible edges. Programming this is geometrically more challenging, but results in a slightly more improved 5-unit toroid. The benefit is, that once the necessary dependencies have been put in place, they can also readily be used to design an optimal 7-module toroid. <<Specify resulting design parameters.>>

The approach for the 7-unit toroid is the same as for the 5-ring, except that we start with a regular heptagon of side length 1.0 in the *x*-*y*-plane, symmetrically placed around the origin. We then attach 7 vertical quadrilaterals angled outward with 360°/7 between them. The quad faces are again defined by their horizontal diagonal, *hd*, and their vertical half-diagonal, *hc*. The parameters *fx*, *tz*, and *tr* are defined in the same way as for the 5-ring, and the available DOFs can be used in the same way. Figure 5 shows some results. <<Specify design parameters used.>>



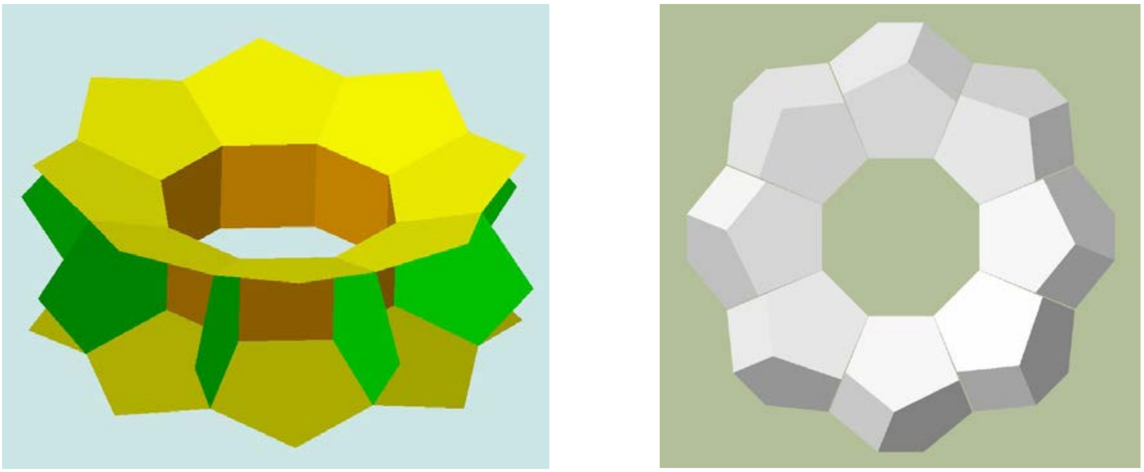
(a) (b)

Figure 5: : *Another "forced" toroidal assembly using seven nonahedral building blocks; (a) using perfectly square quads and all equal edge lengths; (b) with planarized pentagons.*

An 8-module Toroid

We can also try to glue nonahedra together with their pentagonal faces. Figure 6 shows a possible construction of an 8-module toroid. << Specify the relevant design parameters >>

<< Calculate a real solution, show parameter values; add more discussion>>



(a) (b)

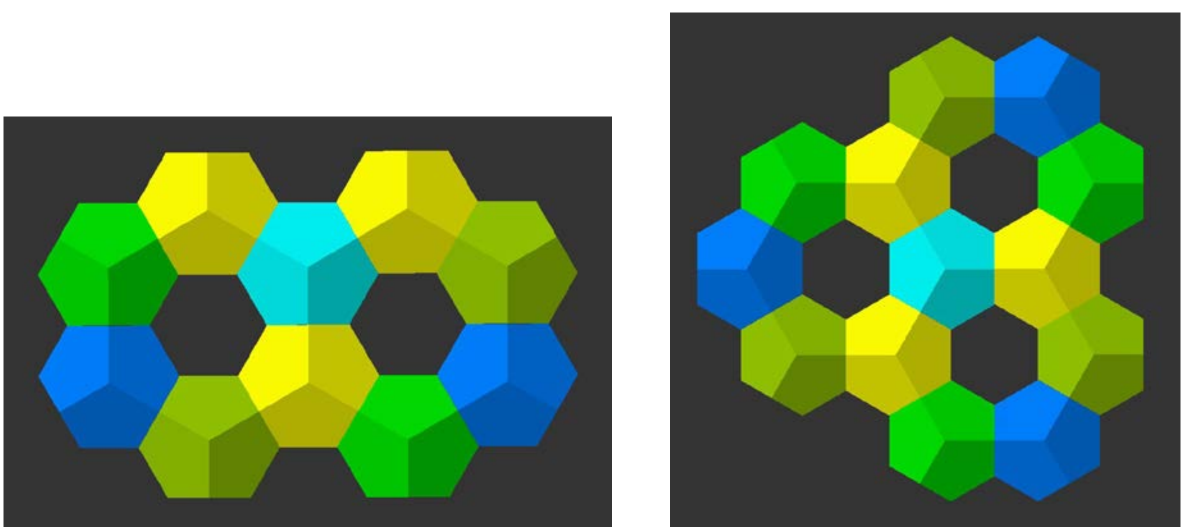
Figure 6: *A toroid constructed by gluing together pentagonal faces of the nonahedral building blocks; (a) the inner construction, (b) <<sketch of the expected result* → *show real solution>>*

6. Symmetrical Structures of Higher Genus

The nonahedral building blocks can also be used to make "handle-bodies" of higher genus, where the "handles" are formed from partial toroidal rings of the type discussed above.

Planar Networks

First, we can readily assemble multiple instances of the hexagonal toroid shown in Figure 1c, placing them into a single plane. Figure 7 shows planar structures of genus 2 and genus 3.



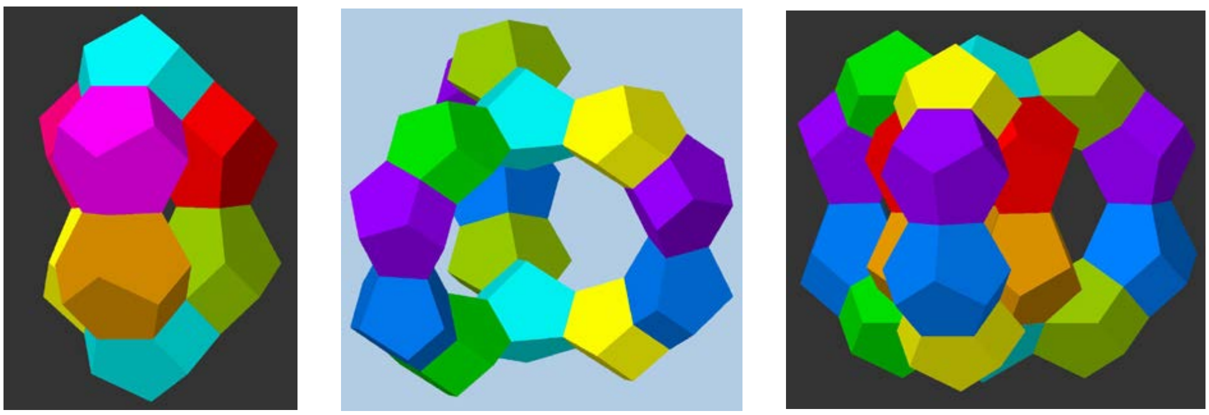
(a) (b)

Figure 7: *Higher-genus objects constructed from hexagonal toroids: (a) genus 2, (b) genus 3.*

3D Cage Structures

The modular assemblies become more interesting and more compact, if we branch out into the third dimension. Figure 8 shows structures of genus 2 that have 12-fold D_{3h} symmetry. Figure 8a is a very compact configuration. The vertical module pairs are connected to the (blue) pyramid tips by their pentagonal faces, angled at 45°, so that all the modules have the same shape. One arch from one of the pyramid tips to the other one constitutes half of another possible 8-module toroid, in which the contact faces alternate between squares and pentagons.

Attaching additional modules in the hexahedral pattern of Figure 7, does not change the slope of the pentagonal faces. Thus the same vertical module pairs can be used to make connections in different locations between two copies of such a planar network. We can use this fact to form a looser genus-2 cage with larger loops between the top and bottom pyramid modules (Fig.8b). We can also combine the inner, tight connectors and the looser, outer loop and thereby create a cage of genus 5 (Fig.8c).



(a) (b) (c)

Figure 8: *3-dimensional cage structures of higher genus: (a) a compact genus-2 cage, (b) a looser genus-2 cage, (c) a combination resulting in a genus-5 cage.*

We also can combine the techniques of Figure 7 and Figure 8 and connect two planes of mostly planar networks into cage structures of higher genus. Figure 9 shows another genus-5 structure resulting from this approach. The upper and lower networks are like Figure 7b without the three dark blue nonahedra.

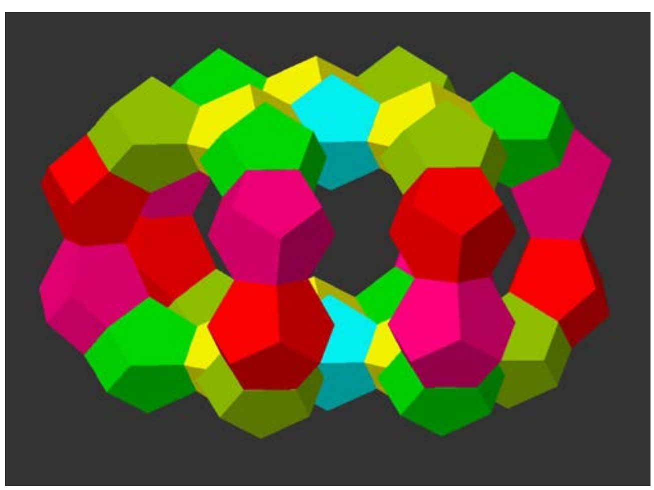
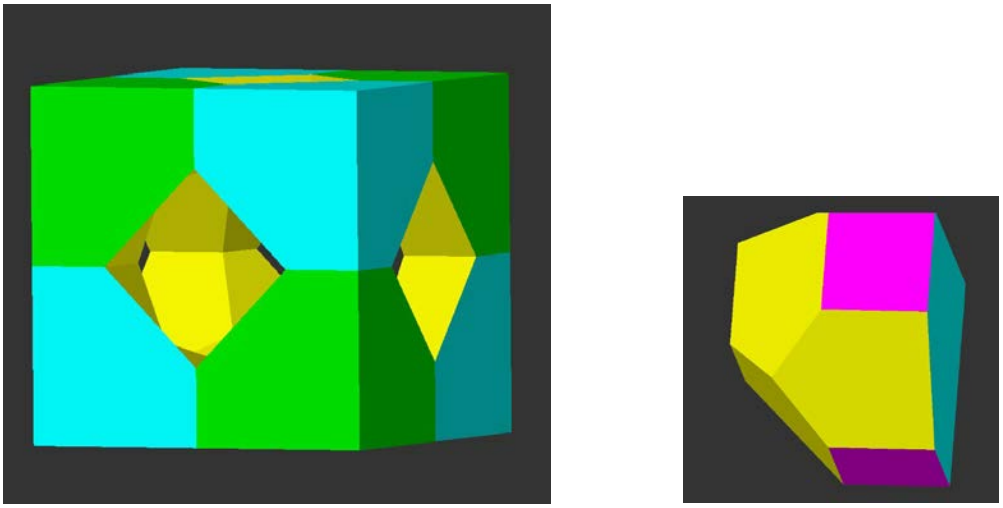


Figure 9: *Genus-5 structure.*

Cube-based Structures

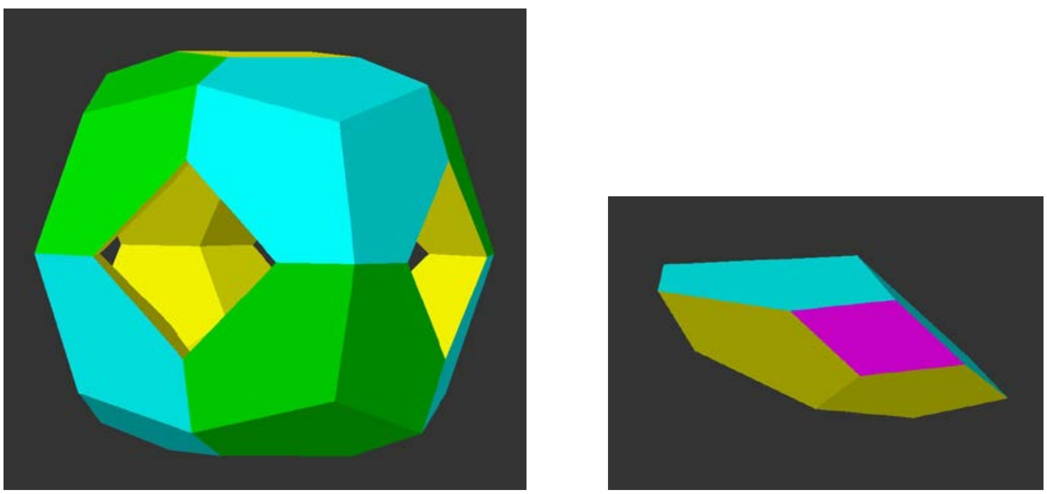
If we are willing to accept more strongly deformed nonahedral building blocks, we can construct even more compact polyhedral surfaces of higher genus. Some of these may be derived from Platonic solids with valence-3 vertices (because of the 3-fold symmetry of the nonahedron). For instance, we can place eight nonahedra at the corners of a cube and let them connect to form a genus-5 cage. Figure 10 shows a simple solution with all planar faces and perfectly square contact faces, where all the outer faces lie in the surface of a cube. The nonahedron geometry is easily defined by placing the red contact squares (Fig.10b) perpendicular to the cube edges at their midpoints. This leaves two DOFs: the size of the red squares, and the position of the yellow inner pyramid vertex; the latter can be chosen to render the yellow inner pentagons planar.



(a) (b)

Figure 1 : *(a) Genus-5 structure based on a cube, (b) one of its nonahedral units.*

Figure 11 shows a more rounded solution with rhombic contact faces, where all faces again have been adjusted to be planar. This assembly has two DOFs: the lengths of the two diagonals of the red rhombic contact faces. In Figure 11, they were adjusted to give the resulting polyhedron an overall pleasing look.

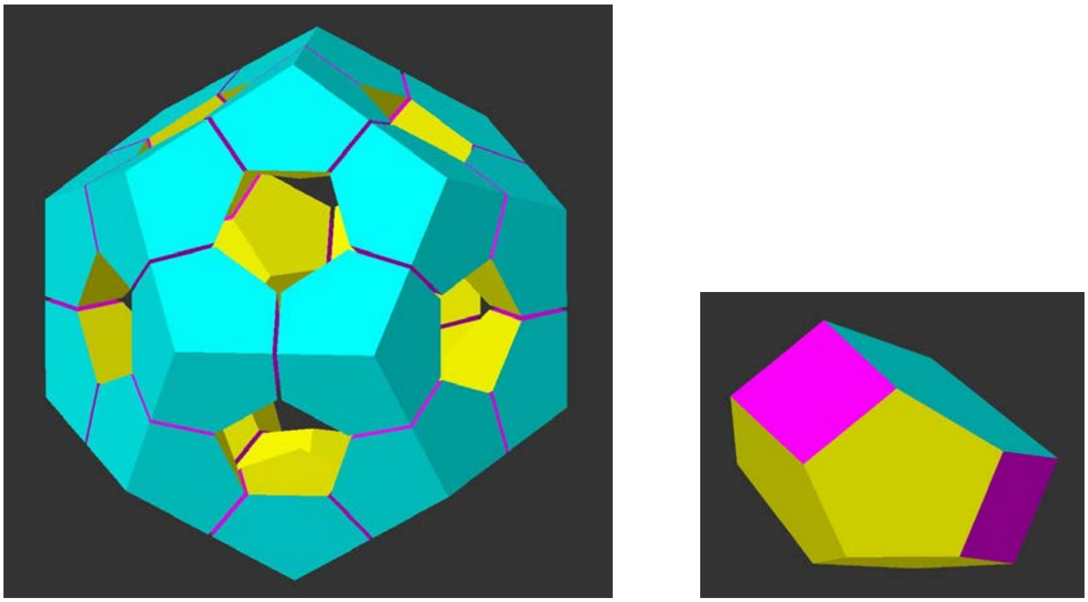


(a) (b)

Figure 11: *(a) Genus-5 body with planar faces based on a cuboid, (b) one of its nonahedral units.*

A Dodecahedral Structure

The same basic approach can be taken to make a highly symmetrical structure of even higher genus by placing 20 nonahedral modules at the corners of a dodecahedron. If the overall shape has a convex hull in the form of a perfect dodecahedron, then the angles in the red contact faces (Fig.12b) are defined by the dihedral angle of ° of the dodecahedron; their size can still be chosen to set the diameter of the pentagonal tunnels. The positions of the inner pyramid vertices are then determined by the planarity constraints of the yellow pentagons.

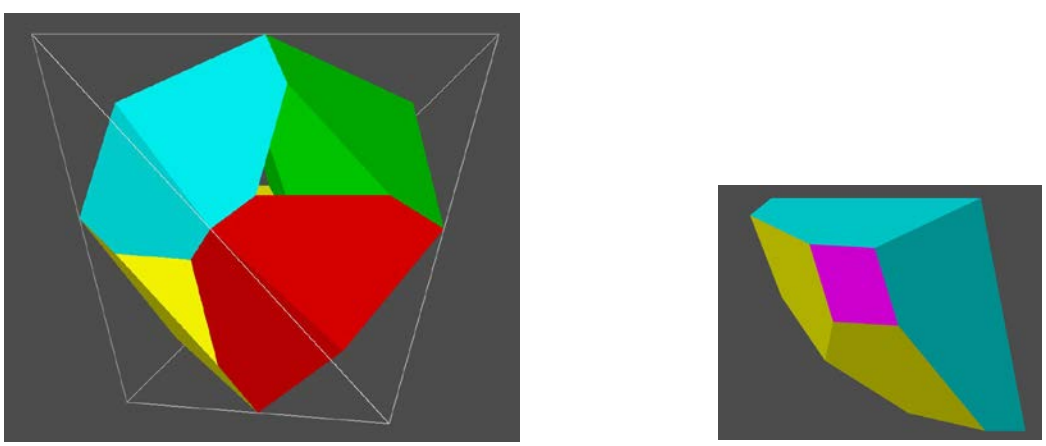


(a) (b)

Figure 12: *Genus-11 structure based on a dodecahedral assembly of twenty nonahedral units: (a) overall assembly, (b) one module showing the (red) contact faces.*

A Tetrahedral Structure

We can even create a tetrahedral cage of genus 3 if we allow a more severe distortion of the nonahedral building block. In Figure 13, the contact faces have been kept as perfect squares and the inner as well as the outer pentagons have been kept planar. This leaves only a single DOF: the relative size of the square compared to the size of the tetrahedral frame.

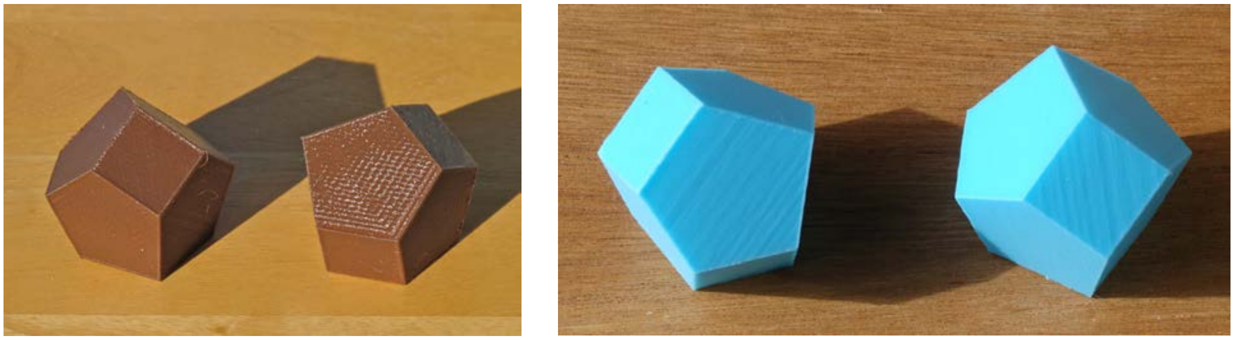


(a) (b)

Figure 13: *Genus-3 structure based on a tetrahedral assembly of four nonahedral units: (a) overall assembly, (b) one module showing the (red) contact faces.*

7. Physical Realizations

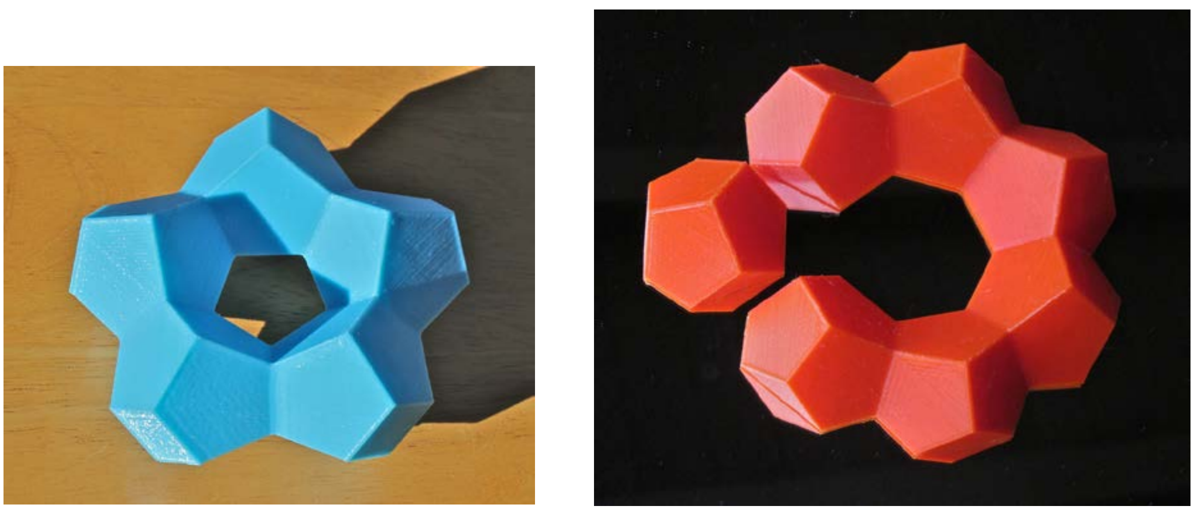
Some of the models discussed above have been realized on some inexpensive 3D printers [4]. Figure 14 shows individual nonahedra. Figure 14(a) is a visually optimized version corresponding to column 1 in Table 1; Figure 14(b) shows a version in which all faces have been made perfectly planar, as discussed in Section 4.



(a) (b)

Figure 14: Isolated nonahedral units: (a) visually optimized as in Table 1, (b) optimized while keeping all faces planar and minimizing edge-length variations.

Figure 15 shows toroidal rings with five and with seven nonahedral units. In both cases, those units have been optimized to have all planar faces. The exposed quad faces are perfect squares, while the quad contact faces were allowed to take on irregular shapes to free up additional DOFs that allow to minimize edge-length variations.



(a) (b)

Figure 15: *Toroidal rings: (a) with five modules; (b) with seven modules.*

Figure 16a shows the physical realization of the tetrahedral frame composed of four nonahedral units. Finally, a dodecahedral model has been built from two identical units composed of ten nonahedral modules each (Figure 16b); the design is based on Figure 12.

Figure 16*:* *(a) Tetrahedral frame of 4 nonahedral units; (b) dodecahedral assembly of 20 modules.*

8. Summary and Conclusions

We have shown the way how to make many more "miracle" sculptures that seem to be composed of almost regular nonahedra: First, one should establish a list of the fixed, non-negotiable demands such as the overall symmetry of the sculpture or the regularity of certain polygonal faces. Then one needs to figure out how these constraints can be incorporated most easily into the design, and how many degrees of freedom (DOFs) remain after these constraints have been satisfied. Next, one has to make some choice about the relative weights given to different kinds of imperfections, e.g., non-planar facets or variations in edge-length; the remaining DOFs are then used to optimize the design under these considerations. This can be done approximately with an interactive computer-aided modeling environment, or it can be done more rigorously with some procedural optimization algorithm. If the initial design is close enough to the final goal, simple greedy gradient descent is often a practical method.

Acknowledgments

Thanks to the staff of Jacobs Hall for their help with the construction of the various models on their 3D printers.

References

[1] Bente Simonsen, *Torus*. Bridges 2016 Art Exhibit. -- http://gallery.bridgesmathart.org/exhibitions/2016-bridges-conference/bente-simonsen

[2] Wikipedia, *Near-miss Johnson solid: Truncated triangular bipyramid.* -- https://en.wikipedia.org/wiki/Near-miss_Johnson_solid

[3] D. Gordon, *Computing the Plane Equation of a Polygon*: -- http://cs.haifa.ac.il/~gordon/plane.pdf

[4] Type A machines. -- https://www.typeamachines.com/