Chaos and Graphics

Gaudi’s organic geometry

Cameron Browne

SWiSHzone.com Pty. Ltd., The Basement, 33 Ewell Street, Balmain 2041, Australia

Abstract

Spanish architect Antoni Gaudi drew on natural precedent to create fantastic organically inspired designs. This paper examines some common motifs in his works, and demonstrates how simple geometric operations can be applied to polyhedral models to achieve similar results. The importance of the partial hyperbolic paraboloid surface (or hypar) is discussed in this context.

Keywords: Antoni Gaudi; Hypar; Hyparhedron; Geometry; Art

1. Introduction

Antoni Gaudi (1852–1926) worked mostly in Barcelona and was famous for his strikingly unique organic style, most notable in his unfinished masterpiece the Sagrada Familia cathedral. Gaudi drew inspiration from natural curves, forms and growth patterns, and incorporated these principles into his designs using a process known as organic construction in which one structural idea adds to another and transforms as it grows [1]. His creations are bold, eccentric, and quite often breathtaking; after 100 years they still look fresh and even ahead of their time.

Fig. 1 shows some typical examples of Gaudi’s work to be found in Barcelona: the roof and tower of a house in Park Güell (left), a chimney pot on the roof of Güell Palace finished in the colorful trencadís style of mosaic tiling (middle), and a sculpture study based upon the branching pattern of the evergreen shrub Abelia floribunda (right), one of the many prototype models in the Sagrada Familia museum that reveal some insight into the underlying principles behind some of Gaudi’s designs.

A common motif in Gaudi’s work is the use of strong, elegant polyhedral models deformed into more organic-looking shapes. We describe four simple geometric operations that can be used to soften the lines of polyhedral models to achieve this effect (Fig. 2):

(1) Relax: Remove an edge without collapsing its end points, which has the effect of combining two adjacent faces into a single face fitted with a minimal surface.

(2) Sag: Bend an edge somewhat towards this minimal surface, but maintain the edge as a sharp crease between the two faces.

(3) Smooth: Smooth local features such as sharp creases, corners and crevasses while retaining their global shape (the sculptural equivalent of a high frequency blur).

(4) Twist: Twist the model about its central axis to impart a spiral effect.

The relax and sag operations may improve a design’s structural qualities while the smooth and twist operations are strictly ornamental. The following sections discuss these operations in detail, with examples, after the related concept of the hypar is introduced. The final section briefly covers implementation issues related to the construction and rendering of the models.

2. Hypars

Fig. 3 (left) shows a pair of adjacent non-planar triangles; we describe such pairs as elbows. Relaxing the
shared edge and fitting a quadratic and doubly ruled surface to the resulting skew quadrilateral gives a shape called a hyperbolic paraboloid (right) or, more accurately, a partial hyperbolic paraboloid bounded by the quadrilateral’s sides [2]. The architect Engel introduced the contraction hypar to describe such shapes [3]. Hypars are also known in the graphics community as bilinear patches.

The hypar is the “simplest” surface to fit between four points [4] and is minimal in the sense that its curvature and surface area are minimized, making it generally stronger than more highly curved shapes and hence an important architectural motif. The hypar shape occurs commonly in nature, for instance in the webbing between the fingers and toes [5], demonstrating how the study of natural form can lead to optimal designs. The hypar is therefore more than just an elegant shape and can be found at both the microscopic (decorative) and macroscopic (structural) levels in Gaudi’s work.

3. Hyparhedra

Demaine et al. introduce the term hyparhedra to describe paper models composed of hypars based upon source polyhedra [2]. Their method of construction involves
creating for each polygonal face a “hat” consisting of a cycle of hypars fitted to each edge, then gluing adjacent hats together. For a polyhedron with \( e \) edges and \( f \) faces this method will create a hyparhedron with \( e + f \) sharp peaks where adjacent hypars meet.

Fig. 4 shows a simple method for constructing hyparhedra with only \( f \) peaks. The source polyhedron (a dodecahedron in this case) is stellated, and each resulting elbow relaxed to form a hypar. Note that the hypars do not join smoothly with their neighbors and that pinch points are clearly visible where three hypars meet. Since the shape of each hypar is defined strictly in terms of its own boundary, \( C^1 \) continuity between neighboring hypars will not occur except very rarely and by coincidence. Fig. 4 (right) shows the figure smoothed to reduce discontinuities.

Demaine et al.’s method of construction builds upon the source polyhedron and always adds to its volume, however there is no reason that a hyparhedron cannot penetrate its source shape. For instance, Fig. 5 (left) shows an octahedron eroded into an elegant hyparhedron by relaxing four convex elbows into hypars. There is a sculpture of this exact shape in the Sagrada Familia museum.

4. Relaxed models

Relaxing more than one edge per face can give interesting non-hypar shapes. For instance, Fig. 5 (right) shows a cube with parallel pairs of edges relaxed to give a figure with two eight-sided surface patches (or ideally a single, infinitely thin minimal surface with front and back sides).

Fig. 6 shows a simple fractal development based upon the stella octangula, which is polyhedral compound composed of the tetrahedron and its dual (left). Fig. 7 shows a modified version in which concave elbows are relaxed as they are created; the process is cumulative in that the deformation caused by each relaxation propagates through its children. Patches corresponding to terminal nodes will be hypars (the final shape is smoothed for effect).

Fig. 8 (top row) illustrates the underlying polyhedral form of the abelian study introduced earlier in Fig. 1 (right). This structure can be generated by making a unit of the stella octangula with top and bottom points removed (first iteration) then stacking these units with \( 180^\circ \) rotation per iteration.

The final shape can be achieved by sagging the edges of convex creases between levels and relaxing concave elbows (Fig. 8, bottom row) to impart a natural flow while retaining the sharp spiky features. A sufficient amount of sag is applied so that the longitudinal creases join smoothly between levels to form an arc between each spike. Fig. 9 shows an extrapolation of this design along a logarithmic spiral path, with an additional \( 10^\circ \) twist per level.

The surface patches fitted to the elbows in Figs. 8 and 9 are no longer strictly hypars, as each patch has two sagged
Fig. 6. Fractal development of the tetrahedron.

Fig. 7. Relaxed version of the tetrahedron fractal.

Fig. 8. Basic structure of the abelian study (top), sagged and relaxed to give its final form (bottom).
(curved) sides while a hypar is a bilinear surface with straight sides [4]. In such cases we use bicubic Bezier patches shaped to approximate hypars by constraining each of the patch’s four interior control points to the plane formed by the nearest corner vertex and its two adjacent boundary control points. This is a necessary condition for the patch to be minimal, and although does not guarantee minimalism is sufficient for our visualization purposes.
5. Twisted models

Twisting a polygonal model is a simple way to break its regular lines and add dynamicism. For example, Fig. 10 shows three regular primitives after various degrees of twist.

Fig. 11 demonstrates how two columns of square cross-section undergoing opposed twists give a ridged grid pattern when combined. When smoothed, this shape yields a set of periodic undulations closely approximating those of the Park Güell tower shown in Fig. 1 (left). Fig. 12 shows this same process applied to two intersecting torii, which are then finished in a checkerboard *trencadís* mosaic style similar to that of the actual tower.

Fig. 13 shows two seashell shapes formed by four tiled bands twisted around a logarithmic spiral surface. C++ code for calculating points on this surface, based on a short C program by mathematician and paleontologist Øyvind Hammer [6], is given in Listing 1. Although the path followed by each spiral has a square cross-section (colored differently on each side) there is sufficient twist to give it the appearance of a series of concentric rings. The left figure shows a variation on the sag operation in which the surface between the ridges sags inwards towards the shape's central axis (essentially the dual of the edge sag operation), while the right figure shows a negative surface sag giving outward inflation.

Fig. 14 shows a similar decorative approach applied to a topological structure called Alexander's horned sphere, which branches recursively to an uncountable number of wild points [7]. The main trunk also has a square cross-section tiled a different color each side, and each horn is given a twist before branching. In can be seen from the detail of the busy part of the figure (right) that the colored twists do little to aid the visual interpretation of what is already a complex figure—wild indeed!

6. Implementation details

The relax operation was applied to polyhedral models by removing the appropriate edge, and replacing the two incident faces with either a bicubic spline patch (if four sides) or a subdivision surface patch. The sag operation was applied to an edge by replacing the two incident faces with two bicubic spline patches (if four sides) or two subdivision surface patches separated by a sharp crease,
and reshaping the edge and incident control points as appropriate. The twist operation was applied by identifying the shape's central axis, then rotating the shape's vertices and control points around that axis as appropriate.

The textured models in this paper were rendered using the public domain ray tracing software POV-Ray 3.6. Initial models were constructed using polygons and bicubic Bezier patches, each corresponding directly to a POV-Ray polygon or bicubic_patch primitive.

Models to be smoothed were then converted to a subdivision surface format and standard subdivision techniques applied. This can be seen in Fig. 15, which shows a subdivision surface corresponding to an elbow converging to a hypar shape as the number of iterations...
increases. The process is kick-started by adding a vertex at the average of the elbow’s four corners (top right) to nudge the subdivision towards the desired outcome.

Subdivision was performed using Biermann and Zorin’s Subdivide 2.0 library, which implements both extended Loop and Catmull–Clark algorithms [8]. The Loop method was used as it provides strictly triangular output regardless of the model’s initial topology, convenient for the use of POV-Ray’s smooth_triangle primitive. Subdivide 2.0 proved ideal as input vertices may be tagged as corners, creases or darts [9], allowing sharp features such as the spikes in Figs. 5 and 8 to be retained regardless of the degree of subdivision.

Code Listings

```cpp
// u, v: [u,v] surface coordinates in radians (v in range 0 to 2*pi)
// w: width (0 gives path centre, 1 makes concentric rings touch)
// ht: degree to which spiral is “pulled out” of the plane
// ht_exp: non-linearity of height growth (1.333 gives good results)
//
CPoint3 SpiralPoint(double u, double v, double w, double ht, double ht_exp)
{
    double d = exp(u * 0.1) - 1.0; // distance from axis (uncoiling factor)
    double r = w * d; // radius relative to d
    CPoint3 pt
    (r * cos(v) * cos(u) + d * cos(u),
     r * cos(v) * sin(u) + d * sin(u),
     r * sin(v) - ht * pow(d, ht_exp);
    return pt;
}
```

Listing 1. C++ code for calculating points on a seashell (logarithmic spiral) surface.

Fig. 14. Alexander’s horned sphere, twisted and tiled.
The mosaic tilings of Figs. 12–14 are textures defined in POV-Ray’s texture grammar. Fig. 16 shows the steps in constructing the blue mosaic texture by way of example. Firstly, a dappled color distribution based upon the target hue is used to create slight variations in tile color (left). A flat grouting color is then added in a random “crackle” pattern (middle) and the same crackle pattern used to modify the surface normals to give the impression of slightly raised tiles embedded in a substrate (bump mapping). Listing 2 gives the complete definition for this texture.

This technique provides a simple way to achieve an acceptable faux-tiling that conveys the desired effect, however it has some flaws that do not stand up to close scrutiny:

(1) The grouting component uses the same “finish” definition as the ceramic component due to limitations in the texture mapping syntax, and therefore has bright phong-shaded highlights where it should be matt. This is fortuitously hidden by the highlights on nearby ceramic cells.
(2) Dappled color changes do not exactly match cell shapes; ideally each tile would be a uniform hue.
(3) Cells may straddle hue boundaries to give tiles with sections of more than one target color.
(4) Cell shapes created by the random crackle pattern can be degenerate.

Textures may be independently translated to address some of these shortcomings, at the risk of clipping tile shapes at hue boundaries.

7. Conclusion

This paper explores some common themes in the remarkable designs of Antoni Gaudi, and suggests some
simple geometric operations for deforming polyhedral models into similar organic-looking shapes. These operations are used to reconstruct some of Gaudí’s designs and extrapolated to create new designs. It should be pointed out that these techniques were observed in a casual study of Gaudí’s works over a weekend spent in Barcelona, and only hint at the full richness of his works.

References


Listing 2. Faux-tiling texture defined in the POV-Ray Language.


