Artistic Rendering of Curves via Lattice Paths

Anton Bakker
Outsite Networks
2551 Eltham Ave
Norfolk, VA 23513, USA
ABakker@OutsiteNetworks.com

Tom Verhoeff *
Dept. of Math. and C.S., Eindhoven Univ. of Techn.
P.O. Box 513
5600 MB Eindhoven, Netherlands
T.Verhoeff@tue.nl

Abstract

We present a general technique to transform arbitrary 3D curves into more artistic solid curves. The transformation first aligns the curve to the lattice; next, it transforms the curve into a lattice path with straight line segments; finally, it rounds this path into a smooth curve again. We also show an elegant way of thickening the curve. There are various parameters to play with, such as lattice alignment, the type of lattice, and cell size of the lattice. We implemented this technique in *Rhinoceros* with *Grasshopper*, and show some examples based on closed curves.

Introduction

A considerable class of mathematical artworks start from an idealized (mathematical) curve that is subsequently transformed somehow to yield a thickened version that looks nice. The starting curve can be as simple as a circle, or as complex as a highly entangled polylink. In most of the work by Koos Verhoeff (see [4] and subsequent Bridges articles), this transformation involves beams cut at an angle and connected by miter joints. Smooth curves play a role in [1, 2, 5].

We present a transformation that takes an arbitrary 3D curve as input and that outputs a version of that curve which has been reshaped under the influence of a lattice. Reshaping in this way lends a coherent overall structure to the curve, derived from the lattice. Fig. 1 illustrates this with a circle as input (left, yellow), which is subsequently transformed through the body-centered cubic lattice (left, light blue lines) into a straight-line lattice path (left, dark blue), and finally is rendered as a smoothed and thickened curve (right). In the next sections, we describe the curve transformation, the curve thickening, and we show some examples (the supplement presents more examples).

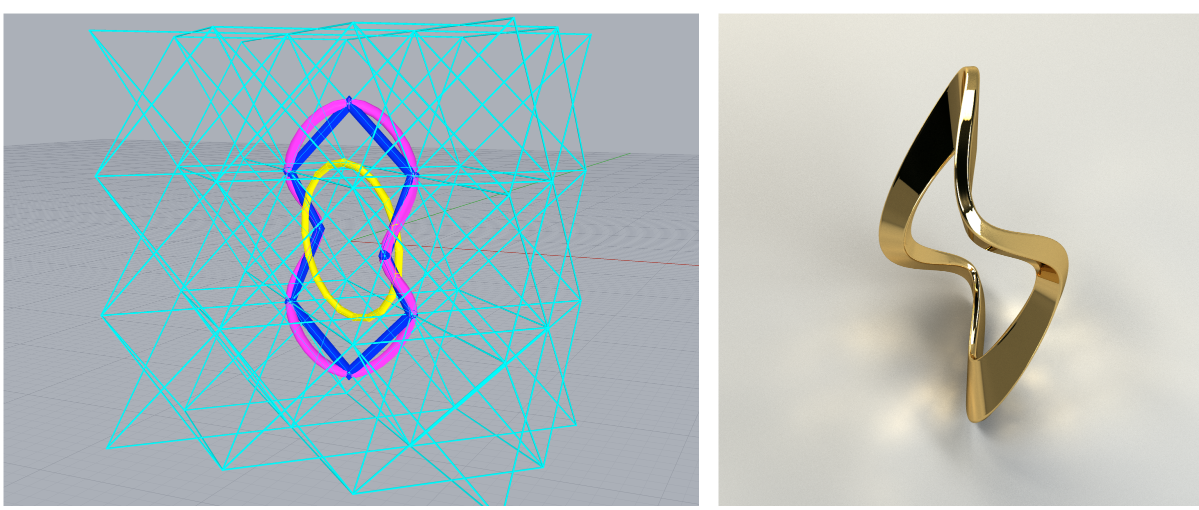

Figure 1 : *Yellow input, cyan lattice, blue lattice path, purple output (left); thickened rendering (right)*

In a way, our approach is directly inspired by, but also a reaction to, the work of Koos Verhoeff. Many of his works involve spatial paths constructed from straight beams with a constant cross section connected by angular miter joints. Although his constructions exhibit nice composition and symmetry, we felt that it could be appealing to generate more organic shapes, that curve casually and whose cross section varies subtly. Beams with a constant cross section and miter joints are classical manufacturing techniques for wood. New manufacturing techniques, such as 3D printing and laser cutting, offer more design freedom.

In our approach, the artist starts with an arbitrary curve, which only needs to be a rough approximation. That curve is then reshaped by a lattice. Here, a lattice is a regular spatial collection of points connected by edges; in particular, we use the three cubic Bravais lattices[1]. The constraints of the lattice dictate an overall structure. Each lattice has its own characteristic point distribution and angles between edges, and offers different symmetries. By playing with the transformation parameters, or with the input curve, the artist can choose an appropriate result quickly. The final result is ready for manufacturing, e.g., through 3D printing.

Curve Transformation Using a Lattice

The curve transformation has been implemented in *Rhinoceros*[2] with *Grasshopper*[3]. The transformation can be broken down into three steps (also see the three *Grasshopper* blocks in Fig. 2).

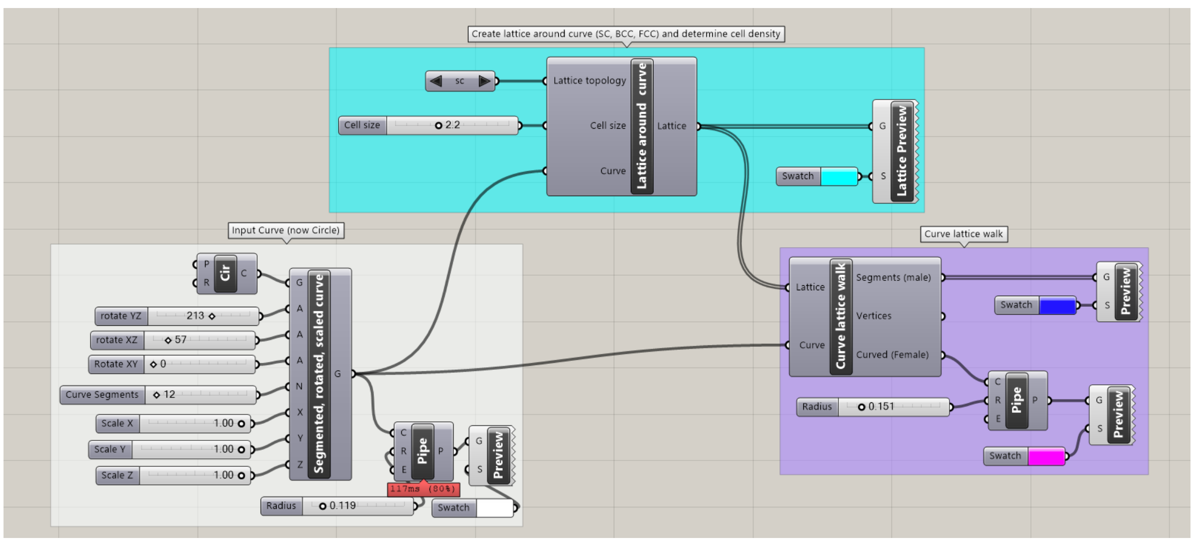

Figure 2 : *The three main Grasshopper blocks for the curve transformation*

1. **Preprocess the input curve**

   The inputs to the first step are:

   - The original curve, being any open or closed 3D *Rhino* curve, including polyline or free-form
   - An optional 3D rotation, to position the input curve in the lattice
   - An optional scaling in X, Y, Z directions, to better match the lattice cells
   - The number of equal-length segments in which to cut the input curve

   The output is a repositioned, rescaled, and segmented curve. A preview of this output curve is shown as a yellow pipe in Fig. 1.

[1] http://en.wikipedia.org/wiki/Bravais_lattice
[2] http://www.rhino3d.com/
[3] http://www.grasshopper3d.com/

2. **Create lattice elements in the bounding box of the curve**

   The inputs to the second step are:

   - The choice of lattice: simple cubic (SC), face-centered cubic (FCC), body-centered cubic (BCC)[4]
   - The cell size of the lattice, which determines the granularity of the approximation in the lattice
   - The preprocessed curve (its segmentation is ignored)

   The output consists of all lattice elements (points and edges) inside the bounding box of the curve. A preview of the created lattice elements is shown in light blue in Fig. 1.

3. **Create (male) lattice path and (female) rounded curve**

   The inputs to the third step are taken from the first and second step:

   - The preprocessed curve, in particular its segments
   - The lattice elements in the bounding box of the curve

   The outputs of the third step are:

   - A curve that approximates the original curve and that steps along (a subset of) the selected lattice elements (more details are explained below); it consists of straight line segments and we refer to it as the *male* curve; a preview of the male curve is rendered as a dark blue pipe in Fig. 1
   - A rounded version of the male curve, which we refer to as the *female* curve; a preview of the female curve is rendered as a purple pipe in Fig. 1
   - The list of selected lattice points (useful for further processing)

   Each curve segment is transformed as follows. Take its starting and ending points, and find the two nearest lattice points. These lattice points are then connected by a shortest path in the lattice, using the *Shortest Walk*[5] plug-in for *Grasshopper*. The male output curve is created by transforming each segment of the preprocessed input curve separately, and then combining the resulting lattice walks into a single curve, while removing duplicate edges. The female curve is obtained by interpolating a free-form *Rhino* curve through the points of the male curve.

This transformation should be viewed as an artist's tool. It is important to play with the various parameters to get good results. Also, the transformation is unaware of symmetry concerns or self-intersection. Therefore, manual tweaking of the input or output curve is sometimes still desirable. To that end, we use a lattice-based curve editor (also implemented in *Grasshopper*).

Curve Thickening and Examples

There are various ways in which a curve can be thickened. One way is to sweep a fixed profile (such as a rectangle or triangle) perpendicular to the curve across its full length, either using intrinsic (Frenet) or rotation-minimizing frames (RMF). Intrinsic mode brings the risk of 'pinching'. With RMF the profile rotates as it sweeps along the curve, and when the curve is closed, the initial and final profile need not align (see e.g. [4]). We have used another way, which guarantees alignment at closure and gives rise to developable surfaces. It involves a central point, e.g., the curve's center of gravity. The profile at each curve point is a trapezoid, such that its two slanted sides extend to the central point, and the parallel sides are perpendicular to the ray connecting the curve point to the central point. That way, the two slanted walls are generalized cones and, hence, *developable surfaces*[6]. This means they can be (laser) cut from a flat sheet of material. The central point must be chosen such that the curve tangents never extend through it. The other

[4]See supplementary material for placement of points and edges in these lattices.
[5]`http://food4rhino.com`
[6]`http://www.rhino3.de/design/modeling/developable/`

two walls we keep narrow enough they need not be developable. Furthermore, we modulate the width and breadth parameters of the profile with the distance to the central point. Typically, one parameter (say width) will increase with distance, whereas the other (say breadth) decreases. This ensures that the area of the cross section does not vary too much, so that the thickened curve is everywhere strong enough when manufactured. The result is an elegantly undulating cross section.

Fig. 3 shows some further examples. On the left, there is a trefoil knot using the simple cubic (SC) lattice. In this case, the input curve was not knotted. Also, the output curve was manually modified to increase symmetry. On the middle, the input curve was a square, which was rendered using the face-centered cubic (FCC) lattice. More examples are available in the digital supplement.

Figure 3 : *Trefoil knot in SC lattice (left); square in FCC lattice (middle); trefoil knot in BCC lattice (right)*

Conclusion

The transformation we presented is a helpful tool to obtain elegant organic curves starting from a rough mathematical idea for a curve. Currently, the tool needs manual tweaking, for instance, to ensure symmetries. A future challenge is to integrate symmetry into the tool as well. It could also be useful to generate multiple options, and let the artist choose.

Acknowledgments We would like to thank Koos Verhoeff [3] for many decades of inspiration and guidance on this topic, and Yoann Mescam for his prolonged help with the use of *Grasshopper*.

References

[1] Michelle Galemmo, Carlo H. Séquin. "Tria-Tubes', *Proceedings of Bridges 2014: Mathematics, Music, Art, Architecture, Culture*, pp. 373–376, 2014.

[2] Carlo H. Séquin. "Tubular Sculptures", *Proceedings of Bridges 2009: Mathematics, Music, Art, Architecture, Culture*, pp. 87–96, 2009.

[3] *Foundation MathArt Koos Verhoeff* (Stichting Wiskunst Koos Verhoeff). URL: `wiskunst.dse.nl`

[4] Tom Verhoeff, Koos Verhoeff. "The Mathematics of Mitering and Its Artful Application", *Bridges Leeuwarden: Mathematics, Music, Art, Architecture, Culture*, pp. 225–234, 2008.

[5] Tom Verhoeff, Koos Verhoeff. "Lobke, and Other Constructions from Conical Segments", *Proceedings of Bridges 2014: Mathematics, Music, Art, Architecture, Culture*, pp. 309–316, 2014.