

Algorithmic Form Generation for Crochet Technique

A study for decoding crocheted surface behaviour to explore variations

J. Gozde Kucukoglu¹, Birgul Colakoglu²

Yildiz Technical University, Turkey

¹<http://www.automaticlogic.net>, ²<http://www.bot.yildiz.edu.tr/>

¹jgkucukoglu@hotmail.com, ²colak@yildiz.edu.tr

Abstract. *In architecture use of generative computation suggests a possibility of rethinking the form finding process. In order to generate form, one method could be predefining first the production technique and constraining the form by the rules of it. In this study crochet-knitting technique is chosen as a production technique. To explore various forms developed through this technique; a computational model, which the behavior of crochet-knitted surface is embedded into, is developed. This paper explains the process of decoding the behavior of a crochet-knitted surface for a computational model in order to extract the crochet-knitting patterns of complex geometrical surfaces.*

Keywords. *Form generation; crochet technique; hyperbolic geometry; decoding rule*

INTRODUCTION

This research aims to understand the behavior of a crochet-knitted surface and decode its rule for a computational model so that it can be utilized in architectural design process. Use of generative computation suggests a possibility of rethinking the architectural design process. Such rethinking could lead to slightly different comprehension of the designers' decision making. In this study, the production technique is predefined (crochet-knitting) and the form generation is constrained by the rules of production technique. Through algorithmic process of crochet-knitting technique, various surfaces -which imply spatial and structural features- can be generated and automatized for physical production through a computational model. This computational model, which contains the knowledge of knitting surface behavior, facilitates exploration of various forms developed through crochet-knitting

technique. Once a form is developed through this computational model, the rule that is extracted from this computational model is generic and also used for physically knitting of it (Figure 1). This paper illustrates the first stage, which is concerned with the development of crochet-knitting computational model. Further research of this study will follow automatizing the physical production.

ALGORITHMIC THINKING IN KNITTING

Algorithm is a precise specification of a sequence of instructions to be carried out in order to solve a given problem (Rajaraman, 2003). Each instruction tells what task is to be performed. Algorithm serves as a codification of the problem through a series of finite, consistent, and rational steps. Although the sequence of an algorithm is simple, the outcome could still be very complex and unpredictable. Fol-

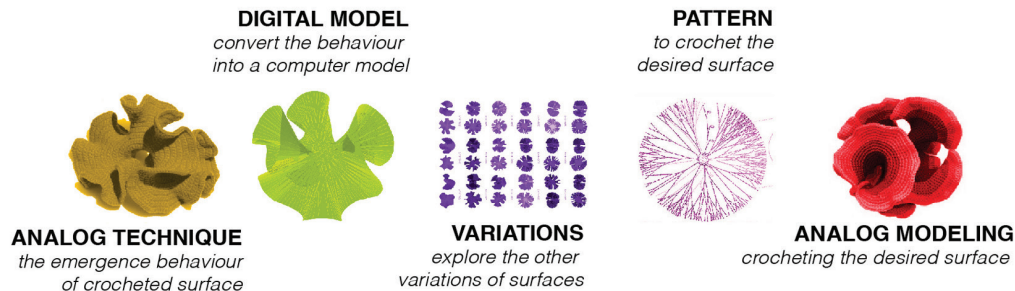


Figure 1
Process of the research aim

Following some specific rules -or a sequence of instructions to do a job- is used in many fields. Knitting is one of the main algorithmic procedures utilized by humans till their early existence. The example below highlights the relation between knitting instructions and algorithmic thinking.

Example: Instruction to knit a sweater

Step1: Cast on 133 stiches

Step2: Repeat steps 3 and 4, 11 times

Step3: Knit 2, *Purl 1, Knit 1, Repeat from * to last stitch, Knit 1

Step4: Knit 1, *Purl 1, Knit 1, Repeat from * to End...Similar steps (Rajaraman, 2003)

By proper permutation and combination of this elementary set of actions (knitting, purling, casting stitches on or of needles), an infinite number of knitting patterns can be created. The algorithm, which is the rules for generating knitting patterns, can be also seen as a translator between human mind and computer. The power of computation, which involves vast quantities of calculations and recursions, can detect abilities that may have not ever occurred to the human mind. The computational model, which is based on knitting algorithm, expands the limits of human imagination.

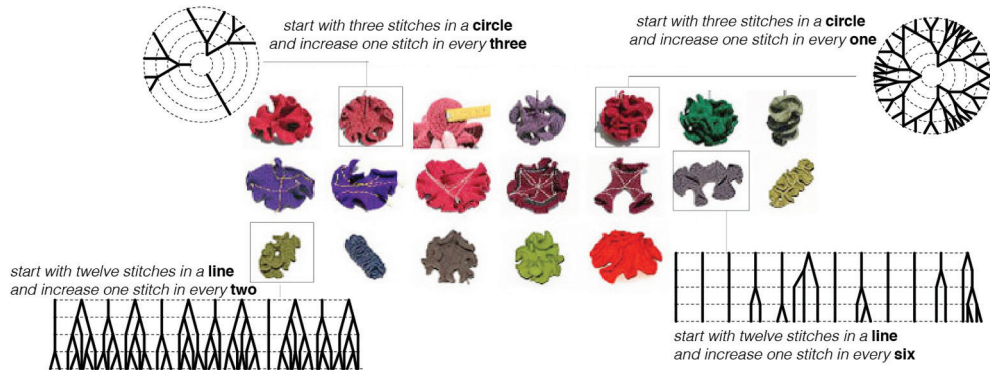
The algorithm instruction given above example illustrates that knitting technique can be automated due to its algorithmic process. Knitting machine, which is invented by William Lee in 1589, uses almost the same principle with hand knitting [1]. It knits patterns with algorithmically defined needle movement. Today, the needle movements can be

controlled by computers, but the mechanism of the knitting machines is not developed. If the mechanics of these knitting machines can be modified for certain behaviours, it would be possible to knit as a whole, those complex forms that explored through the computational model.

KNITTING TECHNIQUE

Knitting is a technique where one continuous line/thread composes not only Euclidian but also non-Euclidian surfaces with a very simple operation. As stitches are added, they push and pull on each other and create an emergent surface. The size of those stitches, and the number of their neighbours in the rows above and below, determines the shape of the work [2]. For knitting a desired surface, the mathematical concept of the surface needs to be converted into a pattern. This conversion requires a greater understanding of the behaviour of the knitted surface because one has to figure out where exactly to increase and decrease stitches so that the resulting surface as a whole is as close as possible to the desired surface. Each work can be quantified as its own pattern. Once knitting pattern is derived than it is generic and the same surface is created aside from the tension of the working yarn. Crochet as one of the knitting techniques is chosen for handmade experiments, since it is easier to compose complex surface. However the surface that is composed through knitting technique is looser than the crocheted surface, both techniques essentially create the same geometry.

Figure 2
Hyperbolic crochet variations
of Daina Taimina and its
crochet patterns.



CROCHET GENERATES HYPERBOLIC GEOMETRIES

Hyperbolic geometry is a surface that has infinite number of lines that go through a point that is specified on the surface but never meet the line that is described on the surface previously [3]. When hyperbolic geometry was first discovered, mathematicians did not understand how it looks like. In 1997, Mathematician Daina Taimina did explain how hyperbolic geometries look like by using her crocheted and knitted surfaces. She found out that just by repeating a very simple operation, it is possible to knit variations of these complex mathematical geometries (Figure 2).

Daina Taimina has crocheted variations of hyperbolic geometries by starting from a row of fixed stitch number and then adding rows. The principle of generating hyperbolic surface is adding one extra stitch in every n stitches (David and Taimina 2001). The number of stitches increases per row and this arises negative local curvature. If n is smaller, more crochet stitches are added so that the concluded surface has a larger negative curvature. This curvature is constant if the process is repeated the same everywhere (Osinga and Krauskopf 2004). This fact demonstrates that the crochet technique promises to create complex surfaces even with a very simple, repeated operation.

DESCRIBING THE CROCHET SURFACE BEHAVIOUR

Every single decision of increasing or decreasing stitches affects the shape of a whole crocheted surface. While making decisions for each stitch, it is almost impossible to predict how the entire surface would be affected from this decision. It is because crochet a desired surface is mostly an intuitional act and it is not so easy to describe what makes a linear thread to create a 3D surface. This makes the behaviour of the crocheted surface emergent. An emergent behaviour or emergent property can appear when a number of simple entities operate in a system to form more complex behaviours together. The reason of emergent behaviour is usually the relation across different scales and there is often a top-down feedback in entire system. Such emergent behaviour cannot be modelled through a standard modelling software, since it should be embedded into the modelling tool. This requires the necessity to understand the behaviour of the stitches on the knitting surface. In order to find out the geometric logic of this emergent behaviour, another technique of constructing hyperbolic geometry is studied. This technique is called polyhedral model. Polyhedral model consists of equilateral triangles and creates a hyperbolic surface if each vertex on the surface belongs to seven equilateral triangles (David, Taimina

2001). Since these two techniques can compose the same hyperbolic surface, is there any connection between them?

As a result of exploring these two techniques, in both of them there is the same physical attraction that forces the whole surface into a hyperbolic form. In other words, the polyhedral model behaves almost the same as the crocheted surface. Each stitch in the crocheted surface behaves like one equilateral triangle in the polyhedral model. While in crocheted surface, each stitch pushes and pulls on each other and whole system creates an emergent 3D surface, in polyhedral model each edge of the equilateral triangles tries to stabilize itself in the same distance. This knowledge is the key argument that enables to perceive and geometrically describe the behaviour of the crocheted surface.

In order to establish the relationship between crocheted surface and polyhedral model, the crochet patterns that are extracted from polyhedral models are tested. In this research: instead of equilateral triangles; pentagons, hexagons and heptagons are chosen for computational and physical model generation. Because using these polygons generates similar smoothness that the crocheted surfaces have (Figure 3).

As shown in Figure 4, hexagonal pattern generates a flat surface. If pentagons and heptagons are also inserted, the system deforms itself into 3D surface.

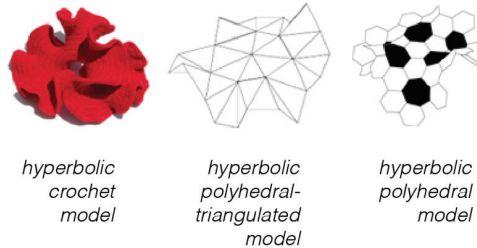


Figure 3
Crocheted, polyhedral models consist of triangles and polygons.

COMPUTER MODEL OF GENERATING SURFACE

The computational model (Figure 5) that simulates crochet technique is created through the polyhedral logic given above and is coded in *processing programming* language. Each pentagon, hexagon and heptagon is added one by one and attached to each other with at least two vertices of the previous polygon. The code defines vertices as well as the centre point of each polygon as a node and forces each node in order to be in the same distance with its neighbours. The form of the surface is governed by the position of these nodes, which provide an easy process for calculating the overall geometry. In *processing* code, each node as a particle is connected with its neighbouring nodes through springs. Therefore, the position of each node needs to be calculated until an equilibrium state was reached for the entire model while any node is added. During modelling, each node determines the emergent be-

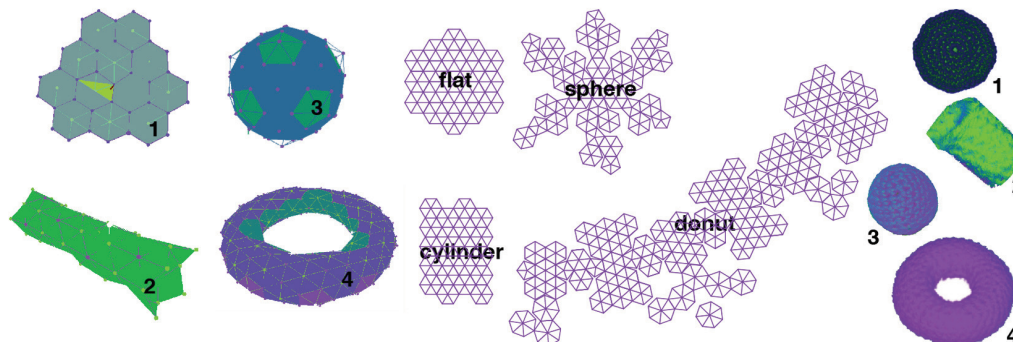
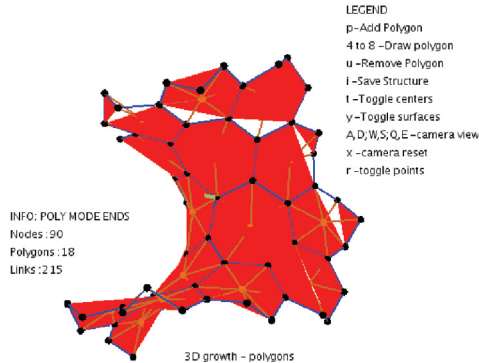


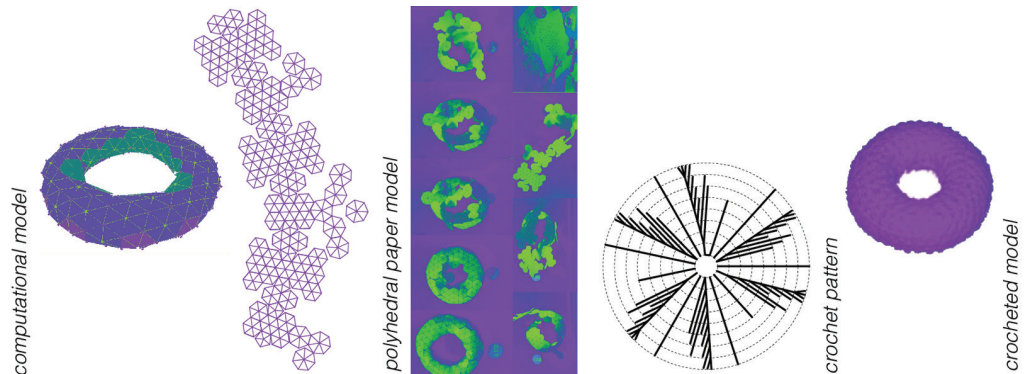
Figure 4
Extracting crochet rules from polyhedral model.

Figure 5
Print screen of the code interface written in processing programming language.



behaviour by affecting on the overall shape. The process is iterative and it has a different approach than a standard computational modelling. It does not start with a pre-defined geometric surface, besides the generated geometric surface is unpredictable. The process of form generation is nonlinear and it enables negotiation between several nodes simultaneously. This negotiation between nodes generates the global form from the local conditions and decisions. The computational model does not have any material properties such as elasticity, etc. because the geometry of the concluded knitted surface is not affected by the property of the material. On the other hand the material affects the rigidity of the concluded form.

Figure 6
The whole process from digital model to analog crochet.



EXTRACTING THE KNITTING PATTERN

To generate the prototype, the code, which is written in the *processing programming* language, is used to create computational polyhedral model of the desired surface. The creation of the surface starts with the first polygon definition then the user controls the surface generation by deciding the position and the number of the edges of the next polygon. Once computational model is generated then it is printed as a flattened surface to build its physical paper model. The pattern of crochet prototype (the number and the order of the stitches for each row) is extracted by counting the number of triangles in each vertex on the paper model. This pattern that is the output of the computational model is used for crocheting the replica of desired (Figure 6).

REALIZATION OF A FULL SCALE PROTOTYPE

To scale the production of crocheted surface, the polyhedral model is also used. In polyhedral model if the number of equilateral triangles is increased, the scale of the whole crocheted surface becomes bigger. In order to increase the number of triangles, loop subdivision method (Figure 7), which is developed by Charles Loop, is applied. This method multiplies the triangles by adding new vertices in the middle of each edge [4]. This polyhedral model with more triangles –created through loop subdivi-

sion method- is used to extract the crochet pattern, which will make the crocheted geometry bigger.

CONCLUSION

This research demonstrates that extracting the crochet rules for each surface is possible through computational polyhedral model. The crochet technique is more promising than Taimina's crocheted models that are shown in Figure 2 in order to generate different variations of hyperbolic geometries. Using hyperbolic geometries in architecture have an important potential since they present an opportunity to achieve self-contained structures. But the conventional way of building them is expensive and it results in material wastage because of complex custom-made casting. In this context, crochet technique could provide building hyperbolic structures by eliminating the need for complex casting. Moreover, the crochet rules that are extracted from computational polyhedral model can also be used as *generator code* during the further research on digital fabrication of these complex crocheted surfaces (Figure 8).

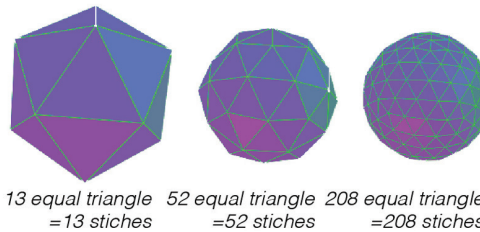


Figure 7
Loop subdivision methods.

REFERENCES

- David, WH. Taimina, D. 2001, 'Crocheting Hyperbolic Plane', *The Mathematical Intelligencer*, 23(2), pp. 17–28.
- Osinga, HM. Krauskopf, B. 2004, 'Crocheting the Lorenz Manifold', *The Mathematical Intelligencer*, 26(4), pp. 25–37.
- Rajaraman, V. (ed) 2003, 'Computer Algorithm', *Computer Programming in Pascal*, Prentice Hall of India, New Delhi, pp. 1-3.
- [1] www.supernaturale.com/articles.html?id=277
- [2] www.en.wikipedia.org/wiki/Knitting_machine
- [3] www.ted.com/talks/margaret_wertheim_crochets_the_coral_reef.html
- [4] www.en.wikipedia.org/wiki/Loop_subdivision_surface

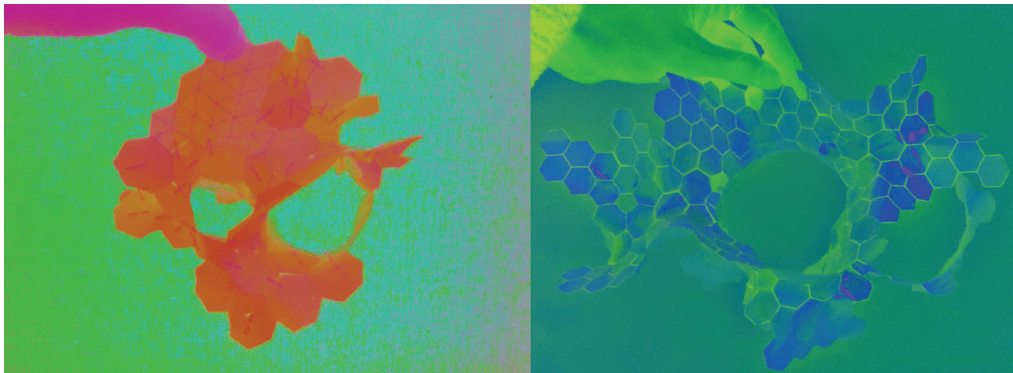


Figure 8
The polyhedral models that are chosen for prototyping.