ARCHITECTURAL FORM GENERATION BASED ON THE DNA ALGORITHM

CHUANWEN LUO¹, MINGYU JIN² and LU YIN³
¹,²School of Architecture and Art, North China University of Technology, China
¹witchwen@163.com ²jinmingyu@126.com
³School of Architecture, Tianjin University, China
³yinlu1983@126.com

Abstract. With the development of computer technology, complex patterns in architectural design modeling have become a reality. The core of architectural form generation in the parametric design is the algorithm. The article observes the morphology and structure of DNA and implements the algorithm by the secondary development of Rhinoceros. By comparing the different parameter combinations that generate forms, the affection of each parameter is more intuitively understood. In the end, a prototype design of a landscape bridge has been carried out applied with the Double Helix algorithm.

Keywords. DNA algorithm; Double helix structure; Architectural Form Generation; Secondary development; parametric design.

After Modernism, architects devoted themselves to the design of diverse building designs. One important way is to learn from nature and biology. It is generally believed that the utilized of biomimicry for form-generation is for efficiency, because biological forms have evolved for higher efficiency and survival over generations.

This paper proposes a novel method for the architecture design, based on the digital diagram of the biological form of DNA double helix structure, analyzing the prototype and form characteristics, using the computer technologies, the algorithm is designed and developed and applied to architectural design.

1. Research background

1.1. THE PRINCIPLE OF FORM GENERATION

As a complex of function, technology and art, architecture has its own characteristics. It is both an opportunity and a challenge to learn from or transform natural biological forms. The current parametric architectural design combines computer technology with graphics, mathematics, biology, and geometry to form a more intelligent and logical architectural design method. New architectural forms are generated based on algorithms, which can help inspire designers. The form that algorithmic logic produces leads to deeper thoughts.
The integration of parametric into architectural design leads to parametric design. In the parametric design, the core is algorithm, and algorithms can translate geometric rules into different forms. The algorithm contains the logic and functional relation of the form to be generated, and determines the influences of each parameter in the algorithm, and, also, determines the result of the form generation after the calculation. A series of related researches of parametric design carried out by applying algorithm has been studied, such as DNA algorithm, cytoskeleton algorithm, and birds foraging, etc. (Xu Weiguo 2018) According to Xu Weiguo, a professor at the school of architecture, Tsinghua University, parametric design is to quantify parameters changes in design. Each parameter is an analysis of one or more important factors in the design process. Changing the values of the parameters will lead to different design results. (Xu Weiguo 2016)

1.2. APPROACH

As Peter Eisenman said, as a generative device in process of architecture design, the diagram is a form representation. (Eisenman 1999) The pioneer architects combined graphics theory with digital technologies to form a digital graphic design method that integrates the graphics with algorithms in specific operations, which provides a new method and content for architectural design. (Xu Weiguo 2018) At the same time, the development of computer technology also provides a much more powerful technical support for digital graphics. The gradual perfection of the modeling, rendering and human-computer interaction functions in graphics, and the development of computer language all play a crucial role in the architectural prototype design, model analysis and scheme optimization of the parametric design. (Leach 2012)

To borrow biological form for architectural form generation, firstly, the characteristics of biological morphology have to be distinguished and embody these features in the algorithm. Then, the algorithm is written into the program. A series of graphics that share the same biological morphology characteristics can be generated as the embryonic form of architectural design after the computation. The program is not only the simulation of the biological form, but also the generation of plenty of architectural prototypes according to different parameters.

The logic of this research is as below:

1. Morphological feature analysis of biological archetypes.
2. Research and design of the algorithm with a flow chart.
3. Second development based on Rhinoceros, which is programming.
4. Generating the design prototype by inputting parameters.
5. Revise and perfect the logic of the algorithm.

1.3. INNOVATIONS

The innovations of this paper are as follows:

1. The logic of the DNA algorithm has been sorted out and completed.
2. The realization of algorithm directly using C# language on both Visual Studio and Xamarin platforms, that makes the algorithm able to run on Windows
2. Prototype analysis

Currently, the recognized secondary structure of DNA is the double helix structure, and the establishment of this model is based on two aspects. The first is the quantitative analysis of various DNA bases by E. Chargaff et al in the early 1950s (Edström 1951). Secondly, Wilkins’s team used X-ray diffraction to study the crystals of DNA and measured the DNA molecules as spirals (Zubay, Wilkins 1962).

James D. Watson and Francis Harry Compton Crick are based on the study of Maurice Hugh Frederick Wilkins and Rosalind Elsie Franklin in Nature (Watson, Crick 1953), which describes the double helix structure of DNA for the first time, determines the basic morphology of this biological macromolecule and gives the schematic of the double helix structure in Figure 1. The DNA double helix structure they discovered was the B-DNA confirmation, and that is the confirmation of DNA under physiological conditions or under low saline conditions. This confirmation makes the DNA in the lowest energy state, the most stable structure, and it is an ideal state. In addition, there are multiple conformations of DNA such as A-DNA, C-DNA, D-DNA, E-DNA, T-DNA, Z-DNA, etc., but these conformations are based on a double helix structure.

The morphology of DNA can be divided into two related forms: secondary structure and tertiary structure. The secondary structure of DNA molecules is formed by two long chains of deoxygenated nucleotides coiled in antiparallel, in the form of a double helix. Based on which, the double helix chain is constantly changing, then forming different types of helix space segments. DNA tertiary structure is a super-helix structure formed by continuing winding and rotating on the basis of secondary structure.

The first thing to study DNA morphological algorithm is to study the algorithm of the helix. On the basis of helix algorithm, the chain formed by helix is deformed to form a super-helix structure. A spiral can be formed based on an existing curve. If the curve, which the initial helix formed along, is straight, the resulting form is similar to the secondary structure of DNA. Or the resulting form is similar to the tertiary structure if the origin curve is a spatial curve.

Based on the geometric principle of DNA morphology, this study uses the control point method to construct the algorithm, which is setting control points generates the curve. The control points can be set in various ways and can be selected randomly or one by one. For example, the production curve of control points can be selected randomly from the generated points in the rectangular plane. According to the same geometrical principle, the morphological generation is carried out through different data structures and parameter relations.
3. Implementation of the DNA Algorithm

3.1. PLATFORM

In this paper, according to modeling requirements, Rhinoceros is used as a modeling platform, Xamarin and Visual Studio are program development platforms, and based on Rhinoceros Application Programming Interface (API), the algorithm can be developed in C# language on Xamarin and Visual Studio. Rhinoceros is professional 3D modeling software developed by McNeel in 1988 based on NURBS surface, which runs smoothly on personal computers. It can be widely used in architectural design, 3D animation, industrial manufacturing, scientific research, and mechanical design. Its model is suitable for almost all 3D software.

![Figure 1. RhinoCommon Plugin Support Library has been loaded in Xamarin Platform.](image)

Under the OS system, the Xamarin software platform can directly call Rhinoceros in time and execute C# program scripts after the RhinoCommon Plugin Support Library, including wizards and the Rhino debugger, (Thompson 1917) has been loaded. Compared to Visual Studio on Windows, Xamarin was acquired in 2016 which lead it contained the same functionalities in Visual Studio in September 2017. To develop an algorithm, it needs to be generated a .dll file and manually invoke the script in Rhino. The steps are complicated and many temporary files are generated. Therefore, Xamarin on the OS system was used for development in this study.
Figure 2. Xamarin Platform and Programming Interface, Xamarin can directly Calling Rhinoceros for modeling using the command newly developed.

The algorithm development in Xamarin creates Rhinoceros’ Application object, establishes the root object, realizes Rhino application control and interface rewriting through the access to RhinoCommon, Rhino.UI and other function libraries, and invokes the properties and methods of the root object. The referenced library eventually implements handle the exchange.

3.2. STEPS OF ALGORITHM FUNCTION IMPLEMENTATION

Based on the biological morphology, characteristics, and graphical learning and analysis of secondary and tertiary DNA structures, it can be seen that the DNA morphology algorithm needs to study the helix algorithm at first, and on this basis, the helix is deformed into a super-helix structure. The main implementation features are as below:

1) Select the curve A to generate a form on (the type curve including the type of line). According to the Rhino API defines, the RhinoCommon curve type contains the method DivideByCount (Int32, Boolean, Point3d []), using by which, A can be divided into equal segments by inputting the parameter n, the method will also output the equal points collection points3d_part on the curve segment.
(2) A vertical plane can be generated at a specified point on the curve object by calling the method `PerpendicularFrameAt(double t, out Plane p)` contained in the curve object shown in the Rhino API.

(3) The algorithm sets two different generation modes $T$. Mode $T=1$ means Random mode, in this mode, the user enters the parameter as the circle radius $R$, each generated circle are with radius $R_i$ which takes a random value between $(0-R)$ at each point in `points3d_part`. Mode $T=2$ is the normal mode, in this mode, the radius of the section circle is what the parameter user inputted, that is $R_i = R$.

(4) Same as (1), equally divide each generated circles circumstance $C_i$ into $m$ equal segments by inputting the integer $m$, which will generate $m$ points `points3d_circle` on each $C_i$. Then input parameter $d$ as the distance between two helix’s curve, the method `DivideByCount(Int32, Boolean, Point3d[])` will choose out a set of points $P_i$ from points $3d\_circle[i]$ as set `point_choice`, and $P_i+d$ will compose set `point_choice2`.

Besides, the mode $T$ will work on the index $i$ of `points3d_circle[i]`. In mode 1, $P_i$ is randomly selected; in mode 2, $P_i$ is selected by gradually with next index $i$, that is rotating $P_i$ an angle of on the next circumstance.
(5) Generating the two curves derived from helix, *curve1* and *curve2*, by using the method of `CreateInterpolatedCurve (IEnumerable (Point3d), Int32)` based on the sets of `point_choice` and `point_choice2`.

Call the method `SweepTwoRail()` in the Geometry library displayed by the Rhinoceros API, that is, call Rhinoceros’ own commands `Sweep2`. Using *curve1* and *curve2* as the sweep paths, connecting the endpoints of the two curves as the cross-section curve, and sweep by `Sweep2` command, and then plough into the surface.

Table 2. Divide each generated circles.

```csharp
for (int j = 1; j < Num; j++)
{
    ArcCurve circle1 = new ArcCurve(new Circle[] {circle[i].points3d[j].radius2[0]});
    circle1.DivideByCount(per, true, out points3d_circle);
    points3d_circle_multi.Add(points3d_circle);
    if (typeum == 1)
    {
        index = num.Next(0, per);
    }
    else
    {
        index = j % per;
    }
    Point3d point_choice = points3d_circle[index];
    int index2 = (index + d) % per;
    Point3d point_choice2 = points3d_circle[index2];
    points_end.Add(point_choice);
    points_end2.Add(point_choice2);
}
```

Table 3. Generating the two curves.

```csharp
Line line0 = new Line(points_end[0], points_end2[0]);
Curve line_between = line0.ToNurbsCurve();
var sweep = new SweepTwoRail();
sweep.AngleToleranceRadians = doc.ModelAngleToleranceRadians;
sweep.ClosedSweep = false;
sweep.SweepTolerance = doc.ModelAbsoluteTolerance;
sweep.MaintainHeight = false;
var breps = sweep.PerformSweep(nc, nc2, line_between);
for (int i = 0; i < breps.Length; i++)
{
    doc.Objects.AddBrep(breps[i]);
    doc.Views.Redraw();
}
```
4. Architectural form generation

4.1. PROTOTYPE SIMULATION

The parameters affecting the shape of the DNA algorithm program are as below: the initial input curve $A$, the number of copies of the curve $n$, generate mode $T$, the section circle equals the fraction $m$ (instead of the angle of rotation), two curve intervals $d$, the radius $R$ of the circle of the section (distance of the origin curve from the generation line). In the prototype generation simulation, the line and circle are selected as the typical curve types in the Rhino API. Lines, curves, circles represent two-dimensional curves; spirals and spatial curves represent more complex three-dimensional curves. It is a multi-parameter combinatorial experiment of algorithms and scripts.

By contrast, the input of different parameters generates different shapes, but it also contains a certain rule. The parameter $n$ controls the spiral wrap density of the resulting shape, the parameter $T$ selects the spiral generation mode with free or always equal radius, the parameter $R$ controls the size of the spiral, the parameter $m$ controls the amplitude of the spiral rotation, and the parameter $d$ controls the interval between two spirals, which controls the width of the final swept surface.

4.2. APPLICATIONS OF ARCHITECTURAL FORM GENERATION

This paper takes the design of landscape bridges in a project in Zhengzhou, China, as an example to illustrate the application of DNA algorithms in architectural design. The basis for the suspension of the bridge is generated based on the DNA helix algorithm. The prototype is shown in Figure 4. Based on this prototype, the battery program of the Grasshopper plug-in is used to assist with the detailed design of the landscape bridge. The design of this project is a body generated by algorithms and digital design tools, and it is gradually processed and refined.
5. Conclusion

The formation of biological forms often requires a number of relatively simple rules, while the patterns generated based on these rules can be very different. Combined with the computer’s powerful computing capabilities, a large number
of architectural forms can be generated. The research method and technology of DNA algorithm has superiorities:

1. The bionic form, point elements and three-dimensional curves, can be generated based on the DNA algorithm.

2. The DNA algorithm has the potential to generate and optimize the architectural form, which can produce complex and varied forms, and also provide diversified shapes for two-dimensional plane and three-dimensional surface.

3. With the help of Xamarin and Visual Studio software, Rhino can provide the technical support for the research algorithm after the secondary development, which is much more efficient and accurate.

This algorithm is applicable to the space form composed of rods, or the generation of the spatial and planar forms constructed by the ribbon and tubular shape elements, which can be adapted to the two-dimensional and three-dimensional shapes design. It can generate architectural forms not only of large scale, but also be applied in design of small scale.

Acknowledgement

The authors were supported financially by the National Natural Science Foundation of China Youth Foud (approval number: 51508003).

References

Ball, P.h.i.l.i.p.: 1999, The Self-Made Tapestry-Pattern formation in nature, Oxford University Press, New York.


Leach, N.e.i.l. and Yuan, P.h.i.l.i.p. F.: 2012, Scripting the Future, Tongji University Press, Shanghai.


Thompson, D.’A.r.c.y. W.e.n.t.w.o.r.t.h.: 1917, On Growth and Dorm, Cambridge University Press, Cambridge.


Xu, W.e.i.g.u.o.: 2016, Parametric Non-linear Architecture Design, China Construction Industry Press, Beijing.

Xu, W.e.i.g.u.o. and Li, N.i.n.g.: 2018, Digital Diagrams from BIO-Form for Architectural Design, China Construction Industry Press, Beijing.