Contextualizing Generative Design

by Saeed Arida

Bachelor of Architecture (2000)
Damascus University

Submitted to the Department of Architecture
in Partial Fulfillment of the Requirement for the Degree of
Master of Science in Architecture Studies
at the Massachusetts Institute of Technology
June 2004

© 2004 Saeed Arida. All Rights Reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.

Signature of Author ....
Department of Architecture
May 19, 2004

Certified by ....
Terry Knight
Associate Professor of Design and Computation
Thesis Advisor

Accepted by ....
Julian Beinart
Professor of Architecture
Chair, Committee on Graduate Students

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
JUL 09 2004
LIBRARIES

ROTCH

Thesis Readers

Mark Goulthorpe
Associate Professor of Architecture

Martin Demaine
Visiting Scientist, Computer Science & Artificial Intelligence Lab

Contextualizing Generative Design

by Saeed Arida

Submitted to the Department of Architecture on May 19, 2004
in Partial Fulfillment of the Requirement for the Degree of
Master of Science in Architecture Studies
at the Massachusetts Institute of Technology
June 2004

Abstract

Generative systems have been widely used to produce two- and three-dimensional constructs, in an attempt to escape from our preconceptions and pre-existing spatial language. The challenge is to use this mechanism in real-world architectural contexts in which complexity and constraints imposed by the design problem make it difficult to negotiate between the emergent output, the context, and the controllability desired by the human designer. This thesis investigates how generative systems address contextual parameters, including the designer, client, user, meaning, aesthetics, environment, and function. This is demonstrated through my case studies, in which my aim was to avoid computerized unprocessed formalism that does not implicitly allow for any contextual and cultural content. I sought to extend simple algorithmic form-generation processes to allow for the subtleties of a given context to be effectively addressed. Some challenges and questions arose from these case studies. By interrogating different generative machines, common threads and challenges, similar to mine encountered in the case studies, were found. All of the processes that strove towards the creation of a generative system struggled with similar issues: How can we use rule-based systems without sacrificing meaning or function or the humanistic touch? How can we address contextual parameters without a loss?

Thesis supervisor: Terry Knight
Title: Associate Professor of Design and Computation

Acknowledgements

My list of acknowledgements is very long to an extent to which I can make a thesis out of it; were I to write it, this manuscript will turn into an autobiography. Although what is, after all, a thesis but an autobiography? It is a crystallization of how I see the world through my own lens. Through the help of many people, these crystallizations made me see the world differently.

I would like to thank the people who made my trip to MIT possible, including Sinan Hassan, Wael Samhouri, and especially Nasser Rabbat, whose ambiguous fatherly care made me fight my way through this program.

I would like to thank Terry Knight for her unlimited support and great insight.

I would like to thank Mark Goulthorpe who made my architectural lenses sharper.

I would like to thank Takehiko Nagakura for his silent creativity and vision.

I would like to thank Martin and Erik Demaine for showing me the beautiful folds of mathematics.

To all my fellow SMArchS students, thank you for the beautiful days.

For Chika, another thesis could be written. She exposed my Saeed-ness. She made me believe in myself. She made me write bravely knowing that she will edit not only my text but my lenses too.

For my family: I love you.

CONTENTS

Preface

I am tired and bored of my limitations and limited imagination, proscribed by my own and others' aesthetic judgments and images. I feel that I am circumscribed within a small space of creativity. I want to fly out of this box. I want to fly out of my exogenous and endogenous images. I want to fly into unknown terrains, indeterminate and not yet prescribed by my whims. I want to break out of the molds – my molds. I want to use stuff that I would never use when I am conscious. This yearning for the new is what keeps me rolling. I want to be creative to myself and to human history too.

How do I achieve this? How do I displace myself and let the unknown vibrate? I don't want anyone else to interfere, because we, humans, share the same limitations.

A simple solution would be to rely on human cultural production, to read as much as I can and thereby expand my conceptions and ideas. The problem, however, still persists: I am still here. My agency still limits my range of creativity, and yet I cannot control the birthing of the creative spark.

Or should I follow Descartes and his 'analysis-synthesis' theory – to deconstruct our preconceptions and start building up from unquestionable simple notions to reconstruct human knowledge? He started with postulates and axioms to create secure theorems. The problem, though, still persists in translating this data into formal constructs. This is why design methodologists had ended up with abstract diagrams that are difficult to formally translate without depending on pre-existing knowledge of how to translate functional ideas into spatial forms. The solutions cannot be automatically generated from analysis.

Thus, the only solution would be to use a machine, either natural or artificial, that is external to my agency in order to instigate the generative process. But if I resorted to using a machine, I will lose the immediacy of my creative gesture.

So suppose I decide to go on and build a tool that will liberate me from myself. But how can I build a machine that is not designed by myself? A machine will design a machine to design a machine to design a machine – and, maybe, after infinite steps, a machine will design an artifact. When should I halt this loop and do something? .

It seems that the argument is flawed from the beginning. I will have to make a compromise in order to make my own machine. I will design it so that I do not interfere with what the machine generates, and how.

So suppose I magically build a machine to build my artifact. What kind of machine is this? Is it man-made or God-made? (Maybe I have to exclude metaphysical matters at this juncture.) Is it mechanical or electronic? Mechanical machines are known to work in a deterministic linear fashion, upholding a direct correspondence between the input and the output. Thus, I have to use only electronic machines. How can I add complexity to this machine? Since I built the machine and it is electronic, I can disrupt and alter the internal code of the machine. Otherwise, the machine will be useless if I can anticipate the output from my input.

Now, what kind of raw materials should I feed into this machine? Does it process material or immaterial input (information)? Since it is electronic, it is going to digest only immaterial things.

Will the machine simulate my cognitive creative process? What is the point then? I want a machine that transcends my capabilities or at least do things differently. I can make the machine simulate natural growth processes. Maybe, then, will I be able to compress and speed up morphological evolution and generate forms that are alien to me. This might not work either, however, because forms in nature evolve within very complex interrelated environments – climatic, cultural, social, and aesthetic – and over a very long span of time. If I could somehow quantify these factors, then I can input them into the machine. But how is this possible?

Suppose that my machine has produced unanticipated novel forms. What I am going to do with the complexity of these formations? I intended, at the beginning, not to constrain my machine by functional or cultural parameters. So I ended up with forms that cannot be handled. Maybe I should constrain my machine at the very beginning. The resultant form would then be a response to these parameters rather than a response to my own yearning for the new. To become a design logician was not my aim; I am seeking after a machine that would surpass my creative impulse. Besides, this will diminish the unpredictability that I am looking for at the expense of having forms that are controllable.

What kind of machine, then, can be developed to mediate between a complex context, a complex designer, and the proposed forms to be built?

After this hectic process, is it even worth going down this path? This challenge is what fueled my thesis.

This thesis is a quest into form-generation processes that are not metaphoric.

Chapter 1: Definitions

1.1 Context

Context can be defined following Christopher Alexander as "Anything in the world that makes demands of the form"[1] — including designer, client, user, meaning, aesthetics, environment, and function.

1.2 Generative Design

Mitchell, in his book "Computer-Aided Design,"[2] traced back the origin of generative systems to philosophy, literary composition, and musical composition. In architectural design, he traced generative systems back to Leondardo da Vinci, whose idea was later formalized by the textbooks of the Ecole Polytechnique and the Ecole des Beaux-Art during the 19th century. Mitchell implicitly defined generative systems as having various architectural elements which belong to a certain vocabulary, that are assembled in different combinations to generate architectural form.

Generative architecture can be more broadly defined as employing a generative system – such as a set of natural language rules, a computer program, a set of geometrical transformations, a diagram, or other procedural inventions – in the design process through which the final design emerges. The generative system has different degrees of autonomous action, ranging from a fully automated process to a step-by-step user-controlled one. This process involves designing the algorithm (rule), adjusting the starting parameters and shapes, steering the derivation process, and finally selecting the best variant.

(Endnotes)

[1] Alexander, Christopher. 1964. *Notes on the Synthesis of Form*. Cambridge, Harvard University Press, p.19.

[2] Mitchell, William J. 1977. *Computer-Aided Architectural Design*. New York: Petrocelli/Charter.

Chapter 2: My Machines

2.1. Introduction

This thesis unfolds as a narrative without a conclusion but a beginning. First, I will mention that an architecture competition, achieved in collaboration with a team of architects and mathematicians, was what instigated me to take up the subject of this thesis. I followed up on this with a later project in which I capitalized on some of the advantages and disadvantages pertaining to the previous project. These initial case studies were mostly drawings punctured by words. This thesis, then, proceeds to answer some of the questions raised by these case studies by looking at different generative formalisms.

Throughout my case studies I was very keen on designing formative machines through which the final design would emerge. For each project, a distinct generative machine was created. These machines were mathematical at their root but constrained within an environment that allows them to be systematically assessed within a rich and nuanced design process. I also tried to extrapolate some principles or a strategy to work with generative systems in real-world contexts.

Through this methodology, my aim was to avoid computerized unprocessed formalism[1] that does not implicitly allow for any contextual and cultural content. I sought to extend simple algorithmic form-generation processes to allow the subtleties of a given context to be effectively addressed.

The challenge that persisted throughout these projects resided in handling the complexity and constraints imposed by the design problem that made it difficult and sometimes impossible to negotiate between the emergent computerized output, the contextual parameters, and the controllability desired by me, the designer.

Two different programming languages were employed to create the generative machines: Python and MEL (Maya). The aim of using these scripts was to create a process description rather than a state description. Creating a process description would bring unexpected results. Since the code is a reduced description, more control can be exerted on the form generated by that description; more variations can be produced by slightly modifying that description. In the words of John Frazer:

> We are inclined to think that this final transformation should be process-driven, and that one should code not the form but rather precise instructions for the formative process.[2]

(Endnotes)

[1] This term was used by Birger Sevaldson in his article, "Dynamic Generative Diagrams." Paper eCAADe 2000, Weimar

[2] Frazer, John. 1995. *An evolutionary architecture*. London: Architectural Association. p.69

2.2 First Machine

Nam June Paik Museum

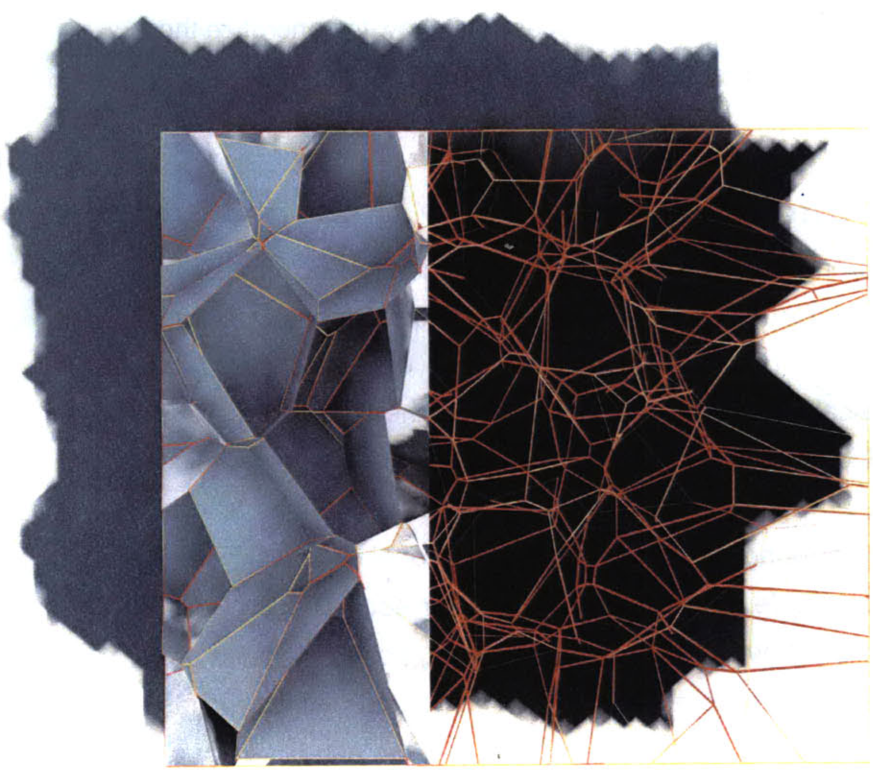


Figure 2.1: An abstract 3D object produced by a software developed specifically for this project.

Background

I, together with a team of mathematicians and architects including Erik Demaine, Martin Demaine, Eddie Chan, and Talia Dorsey, entered a submission for the Nam June Paik Museum Competition held in summer of 2003. The aim of the competition was to design a museum that will enshrine the work of Nam June Paik.

Concept

We approached the design in an experimental manner that would embody the spirit inherent to Nam June Paik's work, in which notions of improvisation, indeterminism and emergence played a significant role.

A form-generation process that is based on natural form was developed to guide and assist us in this design. We started this process unknowing where it could unfold. The starting point was customized software implementing algorithms for computing the Voronoi diagrams. The challenge that persisted throughout the process was how to concretize the abstraction of mathematically generated forms.

Voronoi Diagrams

A Voronoi diagram of a set of points is the decomposition of space into cells whose edges are equidistant from these points. This mathematical process can be thought of physically as lighting a fire at each of the points in a grass field, or growing bacteria seeded at each of the points. The lines at which the fire burns itself out, or where the bacteria stops growing, are the edges of the Voronoi diagram. These edges divide space into cells, one for each defining point from which we started growing. Thus, points in space are assigned to cells according to which of the defining points is nearest.

Voronoi diagrams arise naturally in many contexts, such as crystal growth, animal and plant ecology, mammal coat patterns (e.g., giraffes and jaguars), and bee honeycombs. Recently, astronomers have shown that the distribution of galaxies in the universe is concentrated on the facets of a Voronoi diagram (Icke and van de Weygaert 1987), as described in *Nature* (Webster 1998).

Figure 2.2: A picture of the site.

References

A wide range of applications about Voronoi diagrams can be found at <<www.voronoi.com>>.

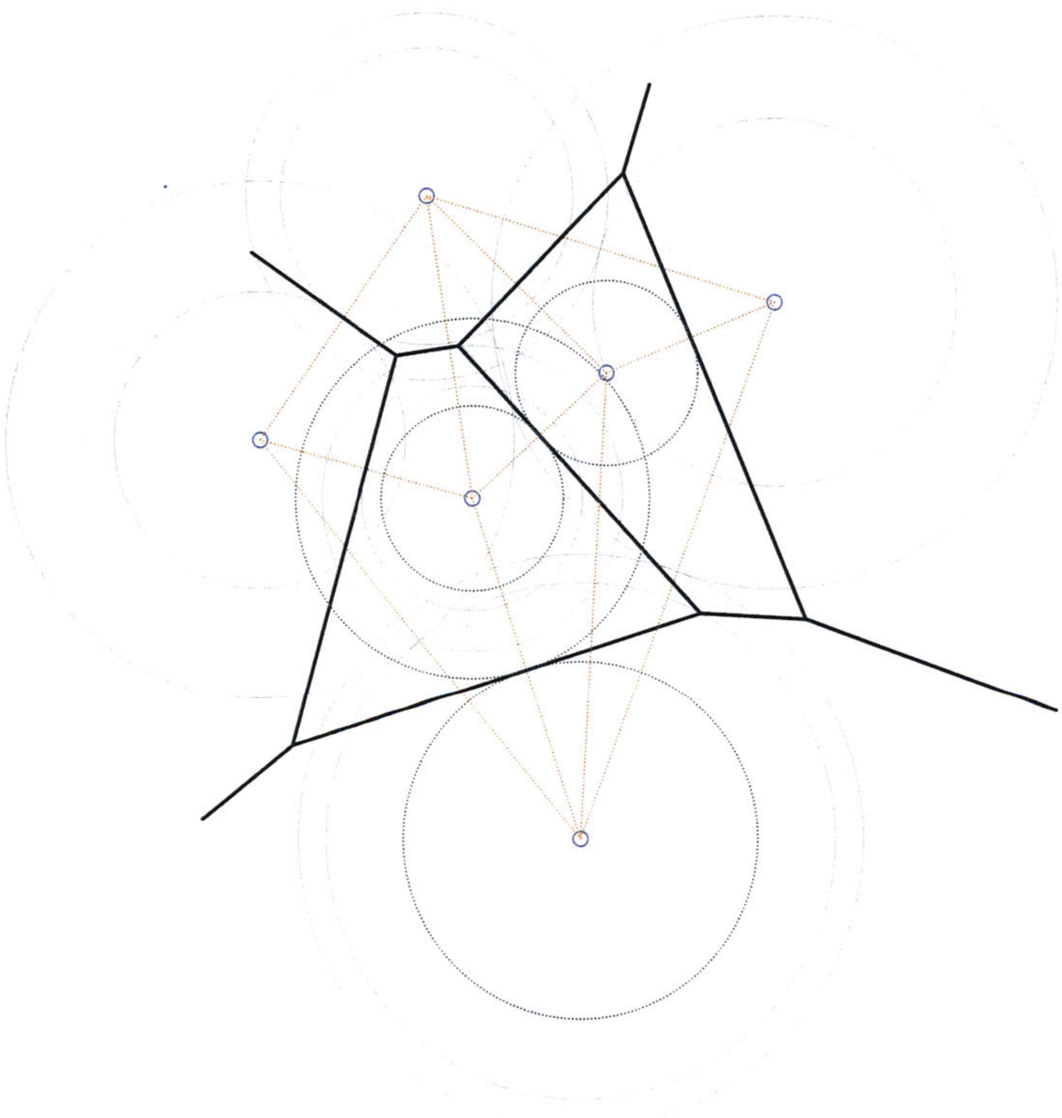


Figure 2.3: Voronoi diagrams are constructed starting from a set of points, as shown here.

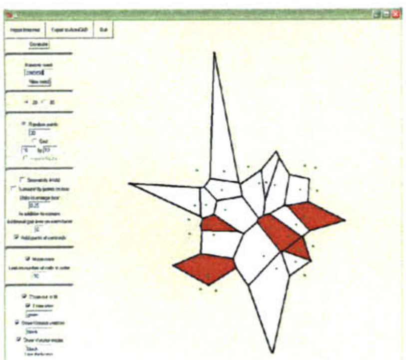
The process of developing the software

1. 2D Voronoi

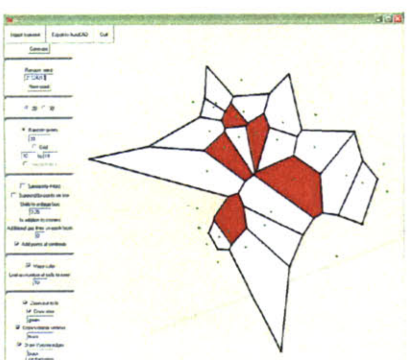
First, we started computing the Voronoi algorithm manually, but after a short period we encountered the impossibility of such an action. We then started developing a software that would implement this algorithm to permit a generative design process that is precise and quick. The development of the software was achieved in stages because it was difficult to predict from the beginning all the parameters that are required to control the algorithm and achieve certain functions in our specific context. The first version was limited to two-dimensional compositions. While the number of points is defined by the user, the placement of these points was at random. Randomess was introduced to get different results by changing the seed. By so doing, the results were unpredictable.

Randomness was introduced to imbue the algorithm with unpredictibility in an otherwise automated process. Every different seed would yield a different result.

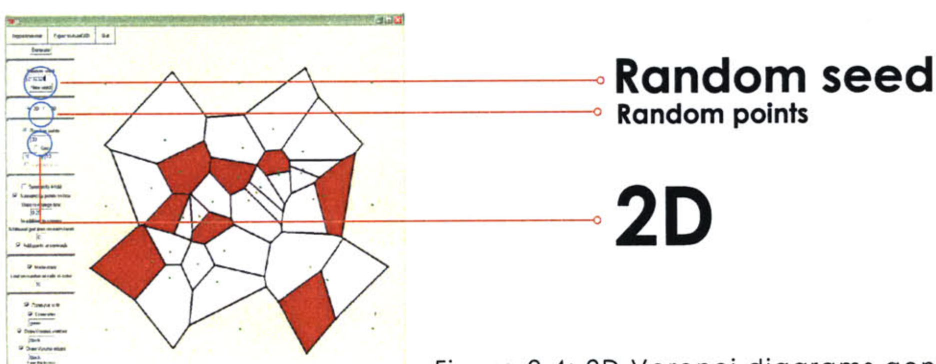
A function was added to delineate the boundary of the resulting shapes. It functioned by adding a series of invisible points placed at the perimeter of a rectangle that surrounds the original points.



seed: 2965656



seed: 21324213



seed: 213234

Figure 2.4: 2D Voronoi diagrams generated by the software based on different random seeds.

2. 3D Voronoi

The software was further modified to include three dimensional potentiality. The placement of the points remained at random. Despite the seemingly aberrant complexity of the algorithmically generated forms, it was compelling to observe the endless unpredictable variations produced by the process. The notion of emergence was conspicuous in this sampling process, where small changes in the initial variables, number of points, and the seed number yielded precise yet distinctive results that were to a certain degree, although mathematically prescribed, unpredictable and indeterminate. Unpredictability not only originated from the randomess of the points, but also from the difficulty of locating the bisecting planes and then delineating where they intersect. This process was achieved quickly and precisely by the software.

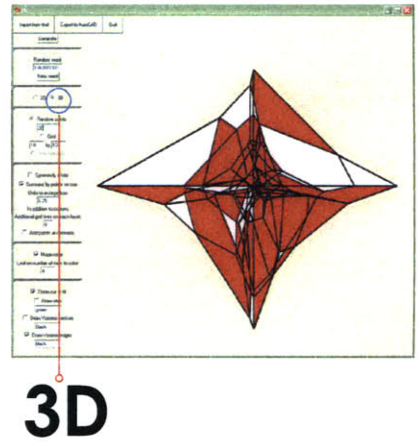
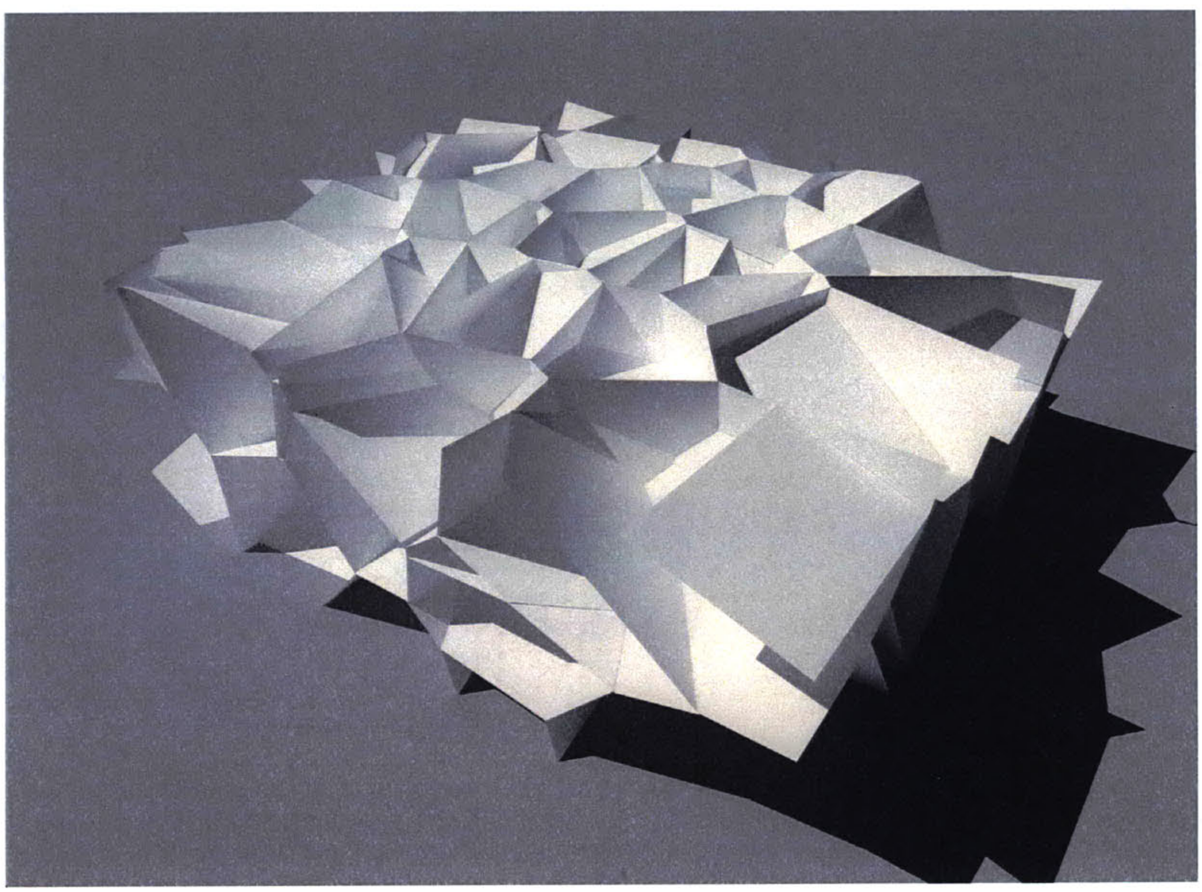


Figure 2.5: 3D Voronoi diagram.

Figure 2.6: A rendered image of the upper diagram.



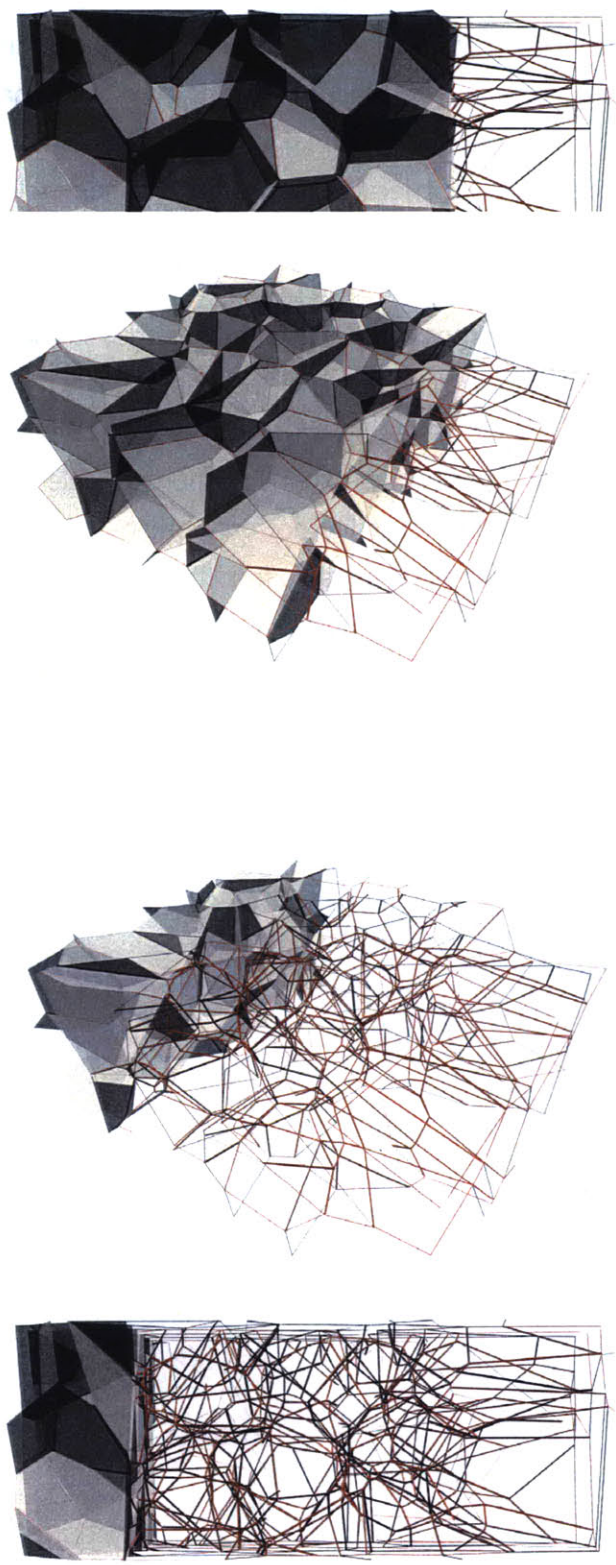


Figure 2.7: Diagrams showing the complexity of the resulting shapes.

3. Mapping The Points

After an initial phase of open generative exploration, the problem became how to modulate the emergent forms given by our rudimentary software into a real-world context (the 'site'). More modifications were developed in an attempt to parameterize the algorithm in an ever-more contextually-constrained manner. Our focus became how to allow for a 'precisely indeterminate' emergence of form, but tempered to meet the physical and social limitations of the site. A new function, "import from text," was added to enable us to enter our points rather than using random ones. These points were mapped out from the site and were based on different parameters - mainly the topography and the programmatic configuration of the museum itself. This helped to produce forms that are closer to the arrangement of points that we have initially entered.

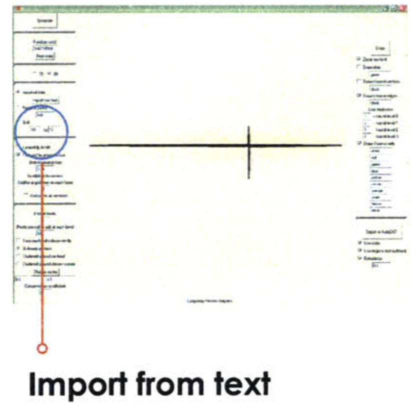
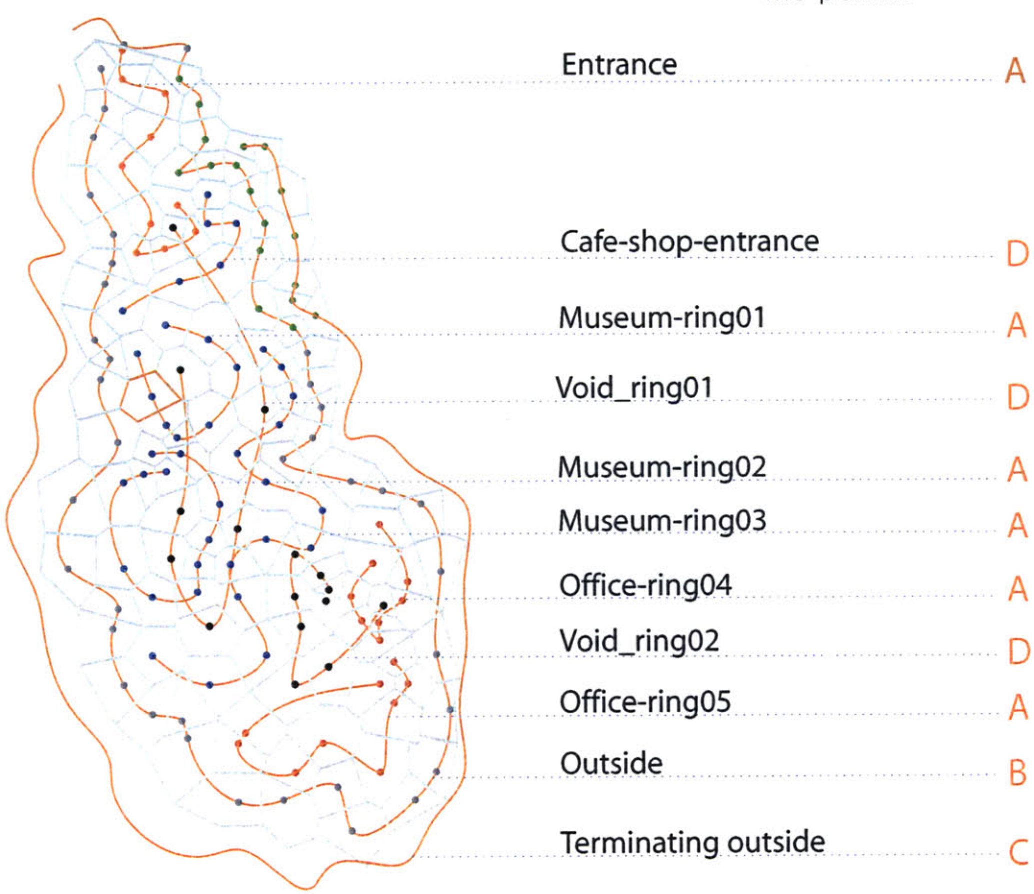


Figure 2.8: A new interface with a new function added.

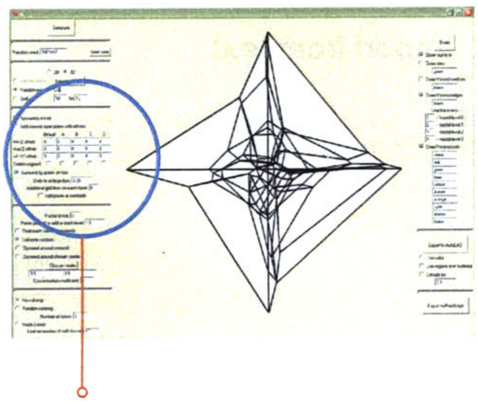
Figure 2.9: Adiagram showing the placement of the points.



4. Parameterizing The Software

However, inputting a pre-determined set of points was not enough to constrain the algorithm in a manner that could be handled. A compromise had to be made at this point, which was to use this generative apparatus to produce only the outer surface of the project and not all of its spatial configurations.

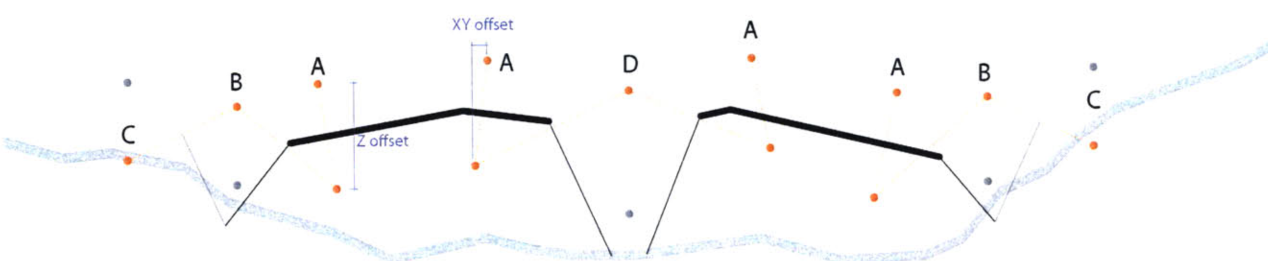
The diagram below (Figure 10) shows how points in Voronoi diagrams should be arranged to produce a continuous surface that folds in a certain way. Following this diagram, another layer of points were added to obtain the desired result. The points located at the voids were offsetted upwards on the z axis (with deleting the original points) to allow for an oblique line to be formed and subsequently to formulate a void. These points were labeled as D. The points on the periphery had the same behavior and were named B. The rest of the points were labeled A, and were offsetted on both the z and the xy axes (with keeping hte original points) within a distance controlled by the added parameters.



	A	B	C	D
Min Z offset				
MaxZ offset				
+/- XY offset				
Delete orginal				

Figure 2.10: A diagram showing the table of parameters that were added to the software.

Figure 2.11: A diagram showing the labeling of each set of points according to the desired functionality.



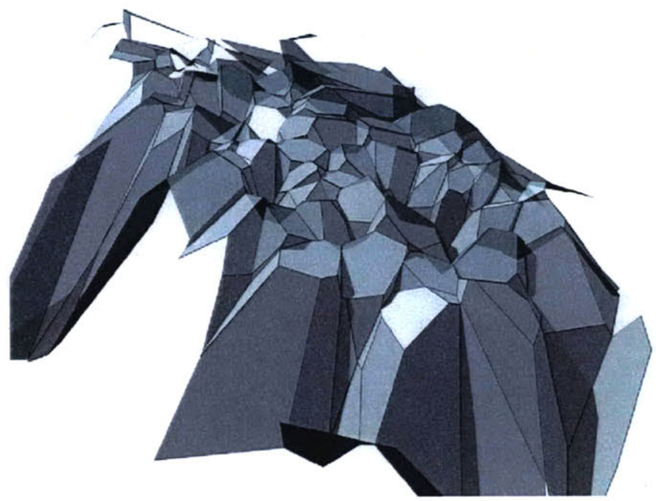
5. Variations

At this point, the algorithm became fully under control and very constrained. The parameters that were added in the previous stage prescribe the behavior of the surface rather than its actual form.

The values assigned to each parameter were selected carefully within a calculated margin to allow different surfaces to be generated. Here are four variations of these endless ones.

1

	A	B	C	D
Min Z offset	3	6	13	3
MaxZ offset	4	7	15	5
+/- XY offset	3	0	2	1
Delete orginal		/	/	/



2

	A	B	C	D
Min Z offset	9	3	6	6
MaxZ offset	10	4	7	9
+/- XY offset	2	0	0	0
Delete orginal		/	/	/

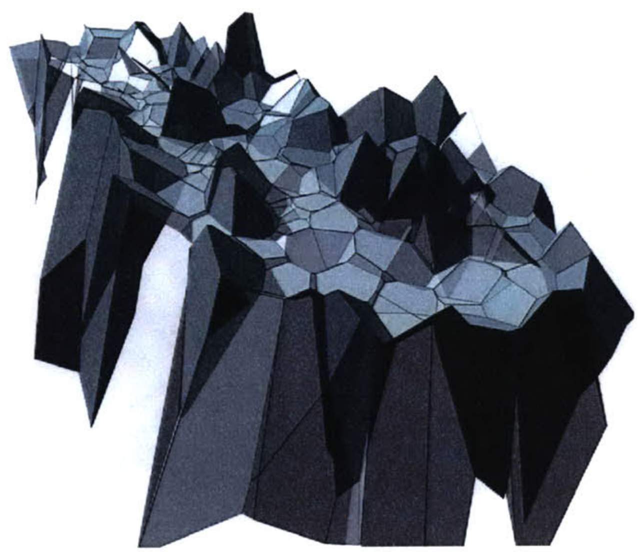


Figure 2.12: The variatiovns using the parameterized software.

3

	A	B	C	D
Min Z offset	4	8	0	5
MaxZ offset	4.5	9	.3	6
+/- XY offset	.7	0	0	0
Delete orginal				

4

	A	B	C	D
Min Z offset	4	8	0	5
MaxZ offset	4.5	9	.3	6
+/- XY offset	2	0	0	0
Delete orginal				

Figure 2.13: The variatiovns using the parameterized software.

6. Selection

In addition to the aesthetic point of view, different criteria played a role in selecting this specific variation. Most important was the mutual correspondence between the building itself and the topography. As we can see from the sections, the specificity of the site topography mandated a special configuration of the section of the building. The parameters controlled three main elements: the disruption of the surface, the light voids, and how the surface connects to the ground. The values shown in the table below offered the right configuration of these elements.

	A	B	C	D
Min Z offset	4	4	0	6
MaxZ offset	4.5	4.5	.3	7
+/- XY offset	.7	0	0	0
Delete orginal				

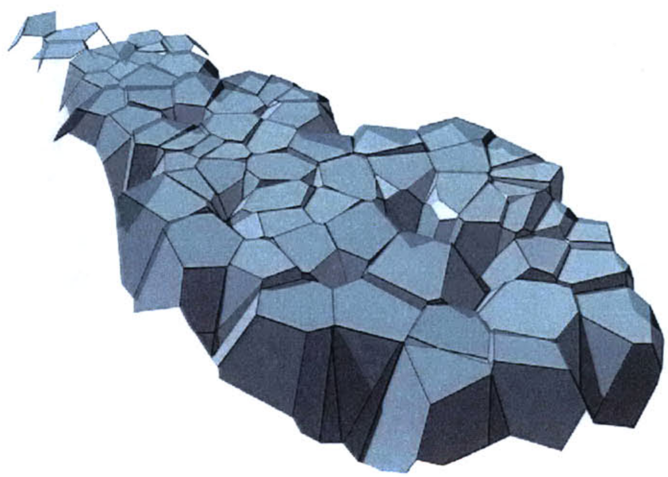


Figure 2.14: The selected shape.

6. Refining

This process went hand by hand with the previous stage. In each variation, we had to dissect the building at different points to get a closer look at how the surface behaved. This process was important to understand exactly how each set of variables would prescribe a definite shape. This permitted a process of fine tuning to these values.

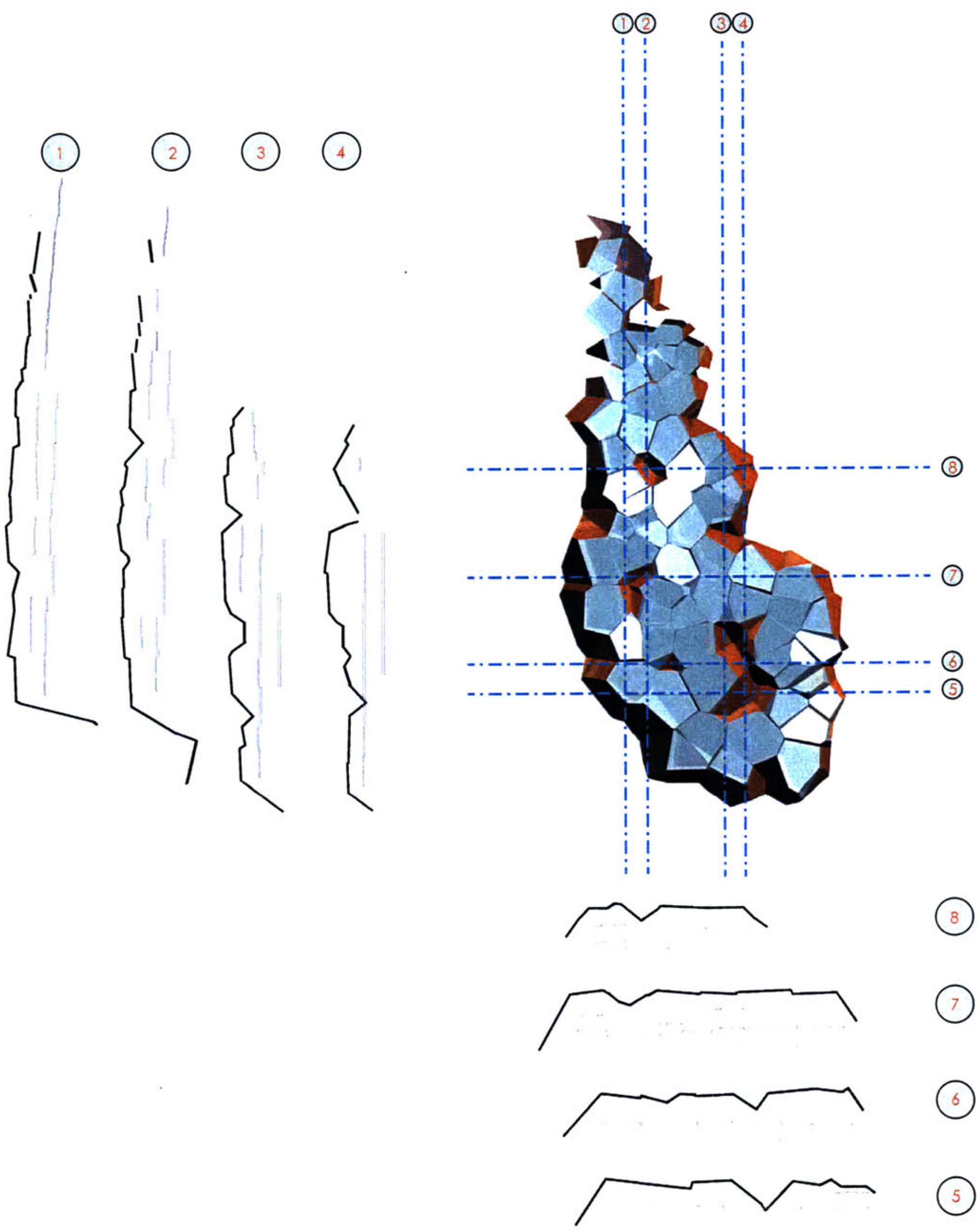


Figure 2.15: The dissection process

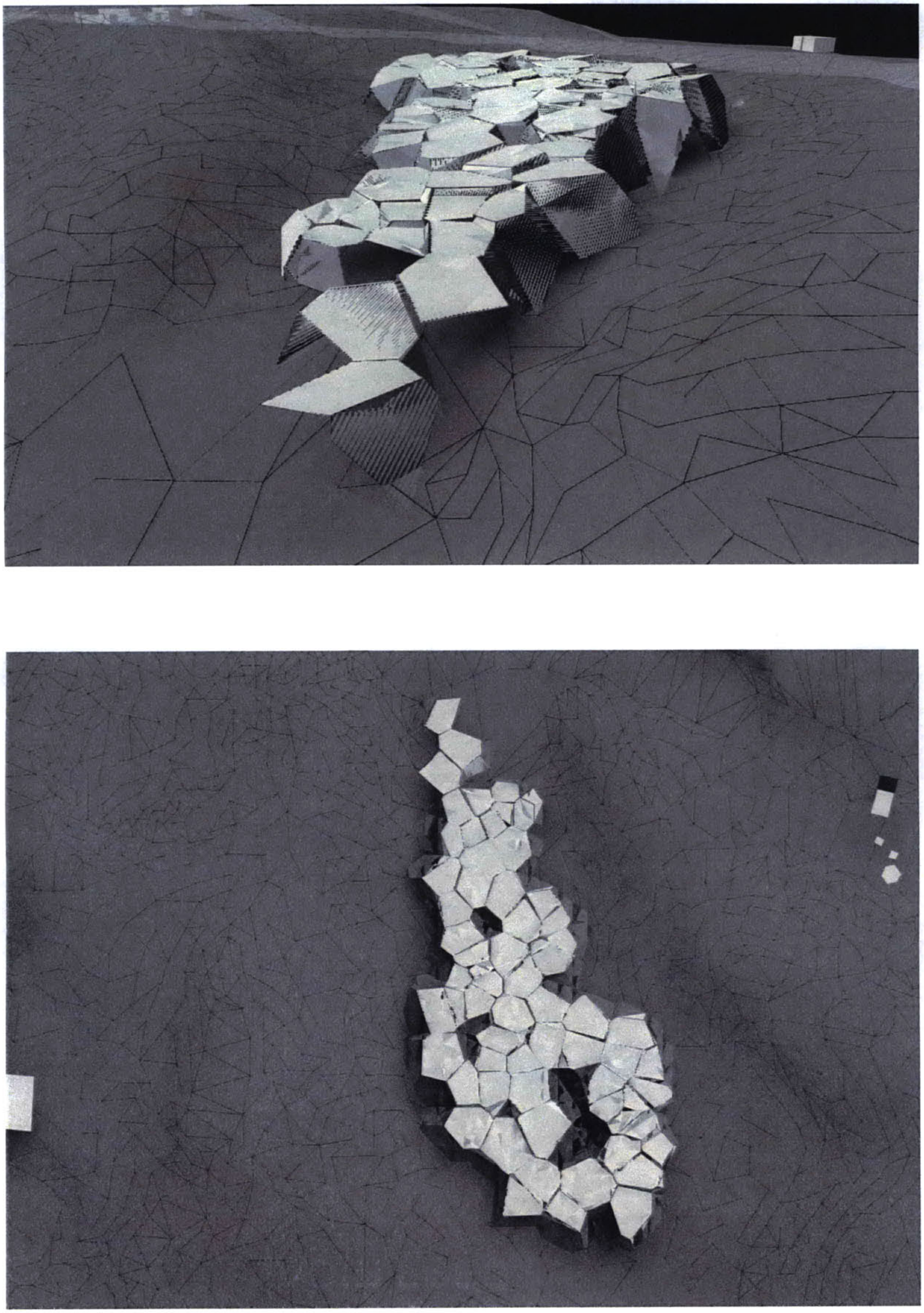


Figure 2.16: Rendered images of the final form.

2.3 Second Machine:
Spiritual Space

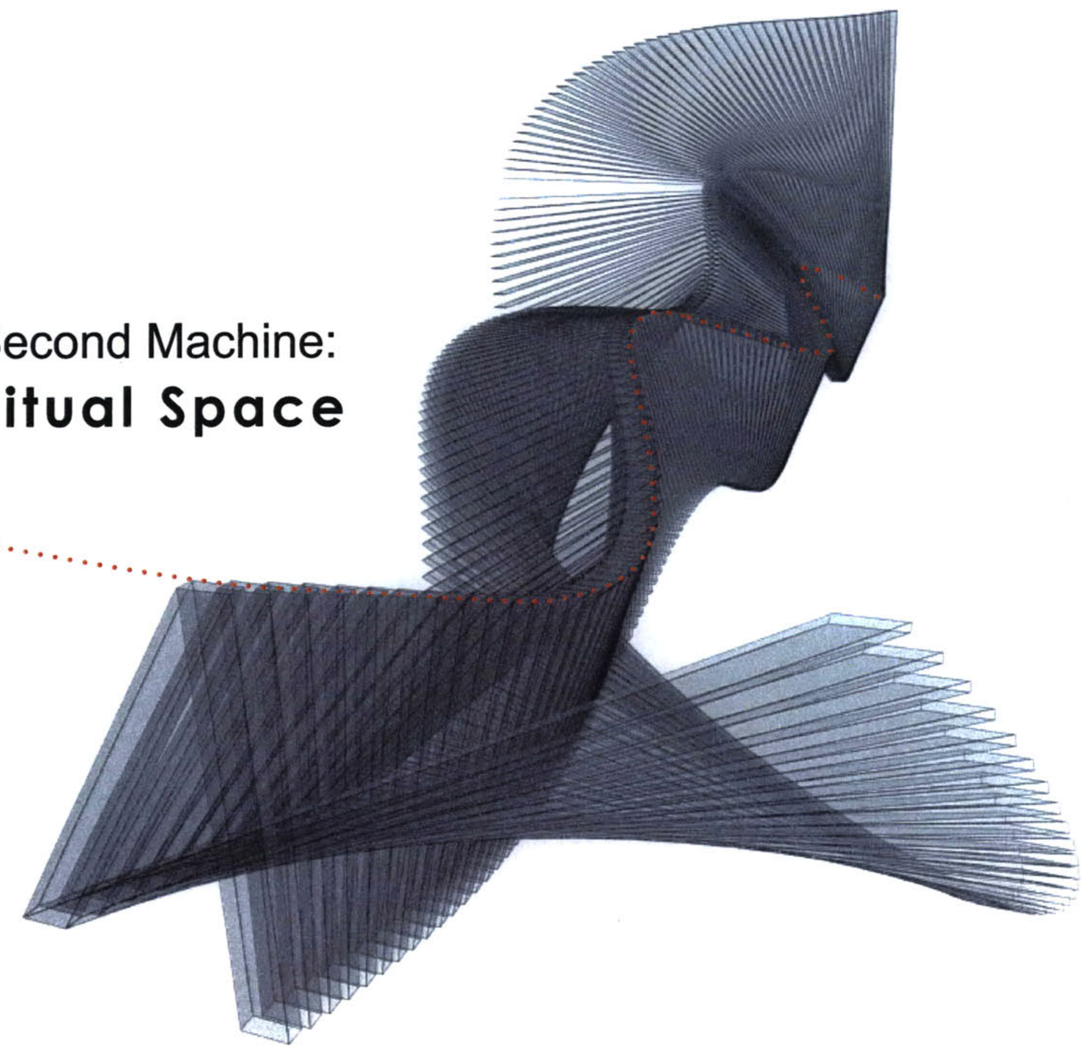


Figure 2.17: An object produced by the algorithm developed specifically for this project

Concept

The aim of this project was to design a spiritual space - a mosque, through an algorithmic means. In this project, more attention was paid to designing the algorithm itself. I wanted the algorithm to be based on concepts stemming from the project. The algorithm was designed to encode physical movements of a genuflecting worshipper during a prayer. In doing so, the algorithm embodied this spiritual materiality on a human scale.

Mapping the Prayer and designing the algorithm

The different postures of a worshipper during prayer were traced as shown below. The algorithm was then designed to capture this movement in a way that allows the changes in any of the segments to propagate to the other two parts. Each one of these segments starts from the end point of the previous segment, and each one has a different rotational angle that can be controlled separately. This imbued the algorithm with fluidity and dynamism especially when it is recursively applied.

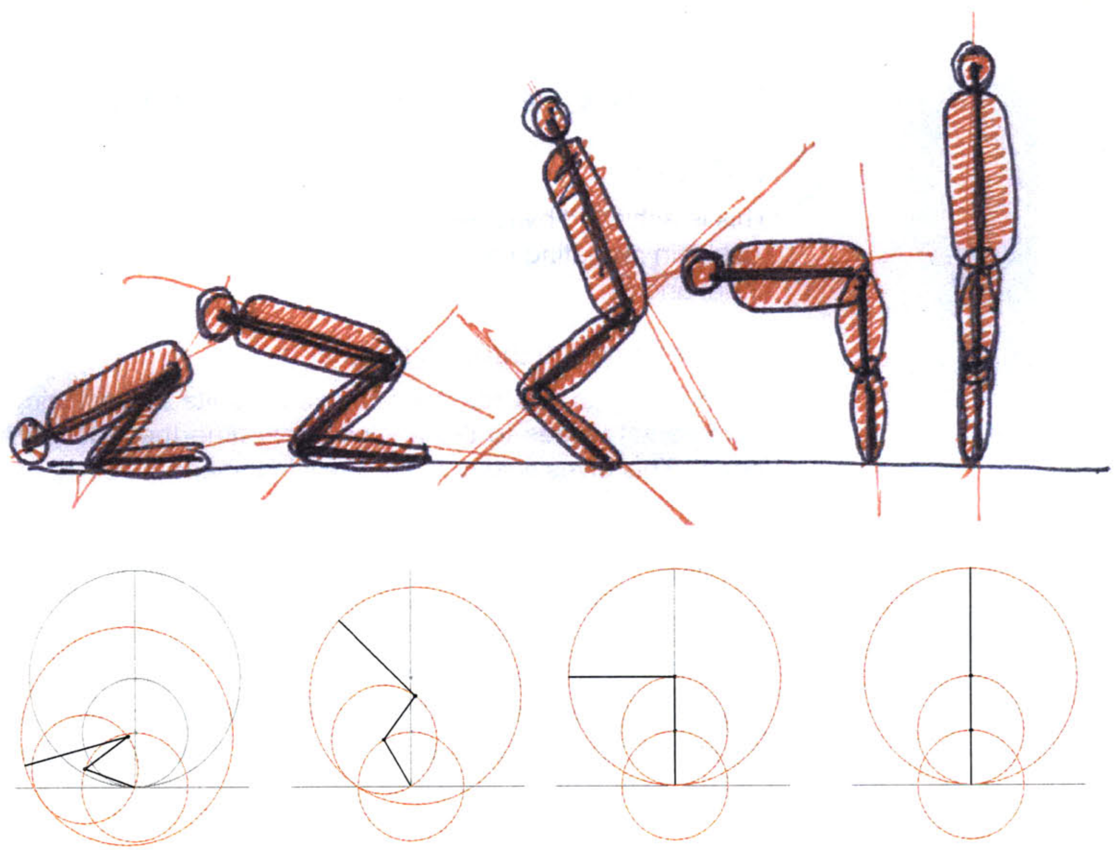


Figure 2.18: A diagram showing different postures during prayer.

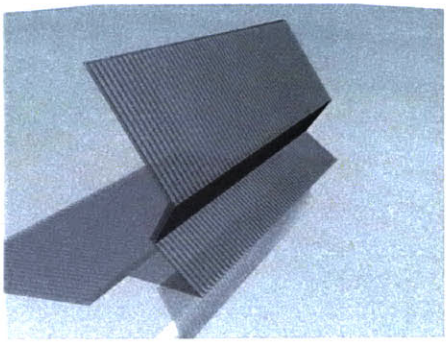


Figure 2.19: Repetition.

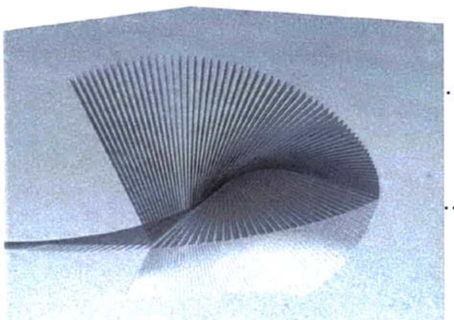


Figure 2.20: Repetition with difference.

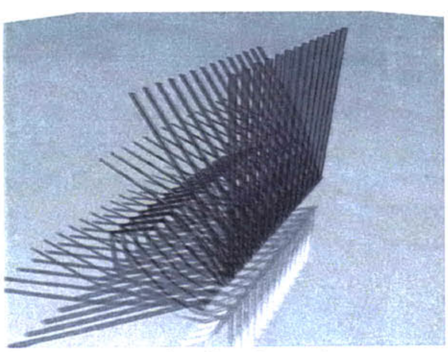


Figure 2.21: Using the Remainder function.

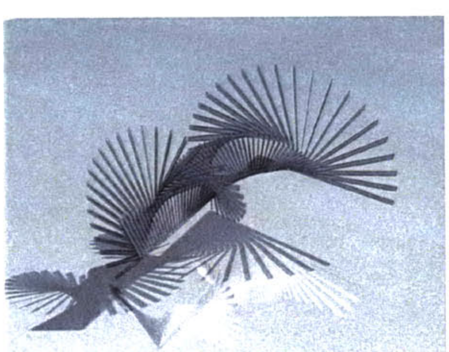


Figure 2.22: Differentiating the segments.

The Algorithm

The algorithm was writtin in MEL - the scripting language of Maya. As shown below, each segment starts from the end point of the previous one and each segment has its own rotational angle.

```
Curve -d 1 -p 0 0 0 -p 0 4 0 -n "h";
 string $angle=30+"deg";
 Rotate -p 0 0 0 0 0 $angle;
 eval("makeIdentity -apply true");
 float $s[]= eval ("getAttr h"+".cv[1]");                  Segment 1
..............................................................................
 curve -d 1 -p ($s[0]) ($s[1]) 0 -p ($s[0]) ($s[1]+4) 0 -n "r";
 string $angle1=-10+"deg";
 rotate -p ($s[0]) ($s[1]) 0 0 0 $angle1;
 eval("makeIdentity -apply true");
 float $sa[]= eval ("getAttr r"+".cv[1]");                 Segment 2
..............................................................................
 curve -d 1 -p ($sa[0]) ($sa[1]) 0 -p ($sa[0]) ($sa[1]+8) 0 -n "g";
 string $angle2=60+"deg";
 rotate -p ($sa[0]) ($sa[1]) 0 0 0 $angle2;                Segment 3
..............................................................................
 attachCurve h r;
 attachCurve h g;

 curve -d 1 -p 0 0 0 -p .5 0 0 -p .5 0 .2 -p 0 0 .2 -p 0 0 0 -n "v";   Profile
..............................................................................
 eval("extrude -upn true -et 1 v h");                      Extrusion
..............................................................................
```

The Process of Developing the Algorithm

1- Repetition

This is achieved by recursively applying the algorithm. This results in repeating the same segments with the same rotational angles.

2- Differentiation through repetition

Adding time to the rotational angles results in assigning incremental values to the angles every time the algorithm loops. This gives fluidity to the resulting shape.

```
$angle=$i*2+"deg";
```

Adding the Remainder function allows different repetition modes to be formed.

```
$angle=20*($i%4)+"deg"
```

3- Differentiating The Segments

Assigning different values to the rotational angles results in segmenting each line so that each series of segments formulate a continuous movement.

```
$angle=2*$i+"deg";
$angle1=5*$i+"deg";
$angle2=10*$i+"deg";
```

4- Waving

Waving can be achieved by employing the two functions sine and cosine. The formula was set up in a way that allows me to control three parameters of the wave: the phase, the amplitude, and the wave length.

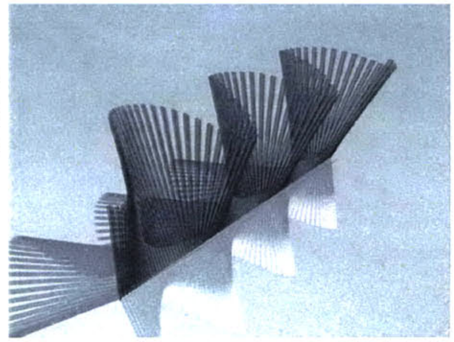
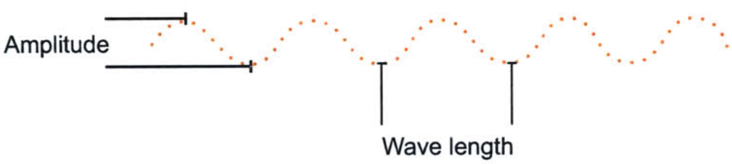


Figure 2.23: Waving.

20*(cos(10.0-$i/5.0))

Amplitude Phase Controls Wave length

Counter



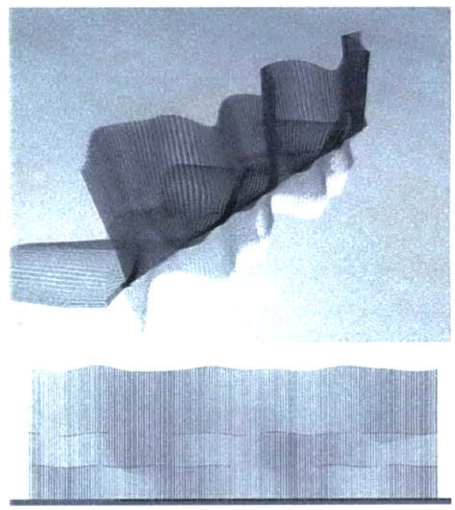


Figure 2.24: Waving with segmentation.

5- Waving With Segmentation

In addition to waving, segmentation was added so that every series of segments would formulate a different wave with different parameters. Because the algorithm was designed in a way that allows the changes in any segment to propagate to the other segments, the three connected waves started differentially undulating.

6- Adding The Absolute Function

Adding the Absolute function to the Cosine function results in limiting the wave to wave only in the positive direction.

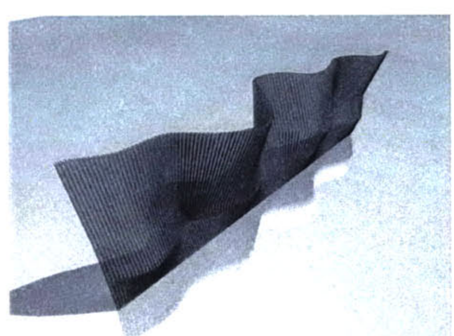
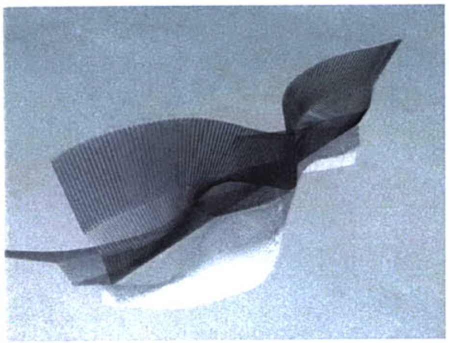
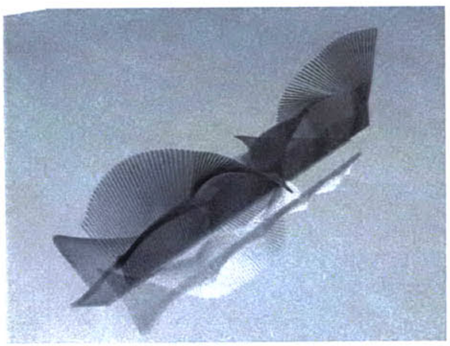


Figure 2.25: Adding the Absolute value.

7- Waving Only Two Segments



Keeping the segmentation but removing the waving from one segment forces the remaining segments to rotate in a full circle while the other two segments swing between negative and positive values according to the cosine parameters.



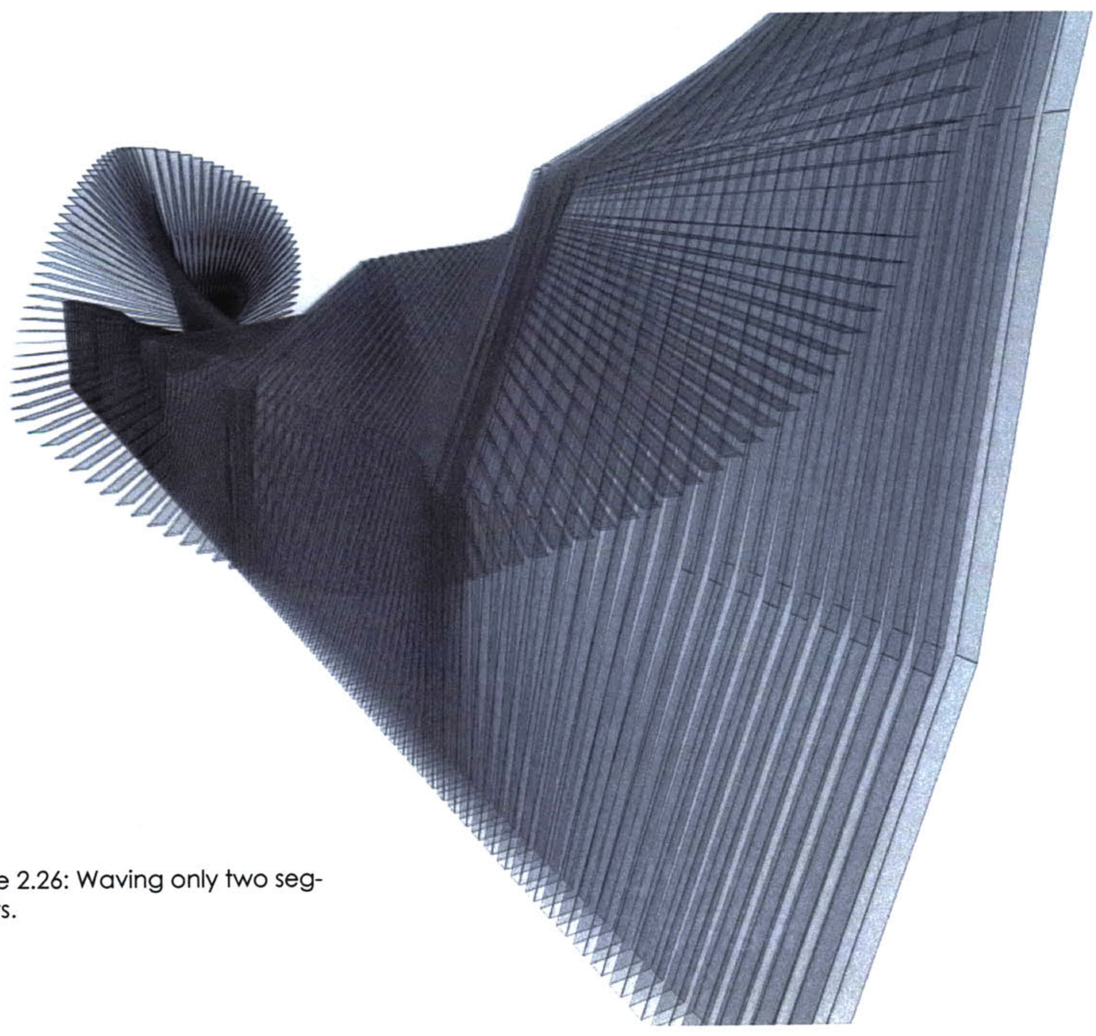
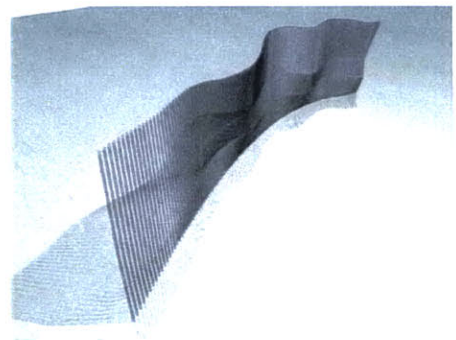


Figure 2.26: Waving only two segments.

8- Curving The Wall

To explore different possibilities, the lines themselves were given a rotational angle. This results in a wall that has a curvy undulating shape.



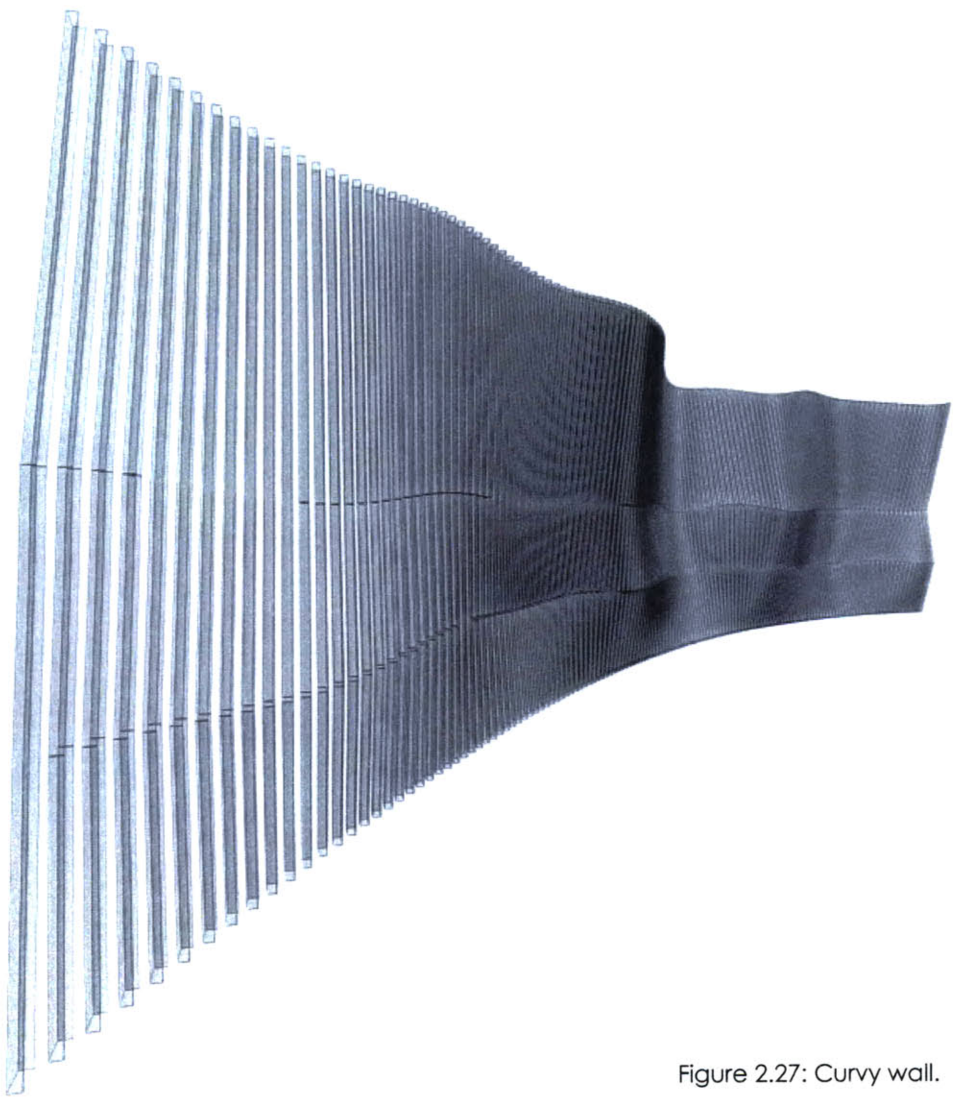


Figure 2.27: Curvy wall.

9- Exploring Different Possibilities

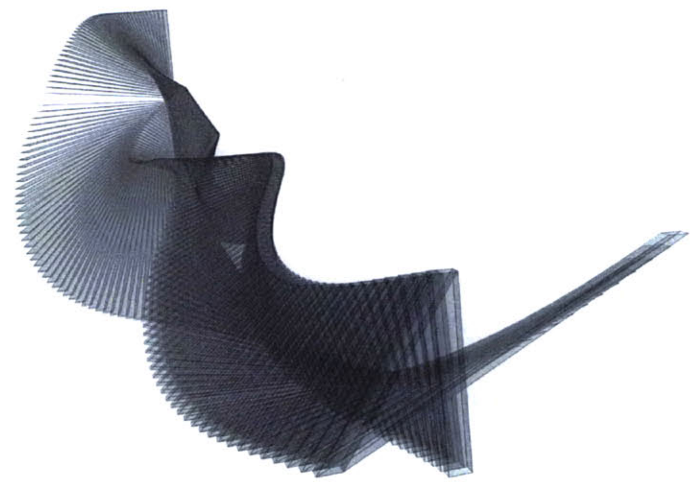
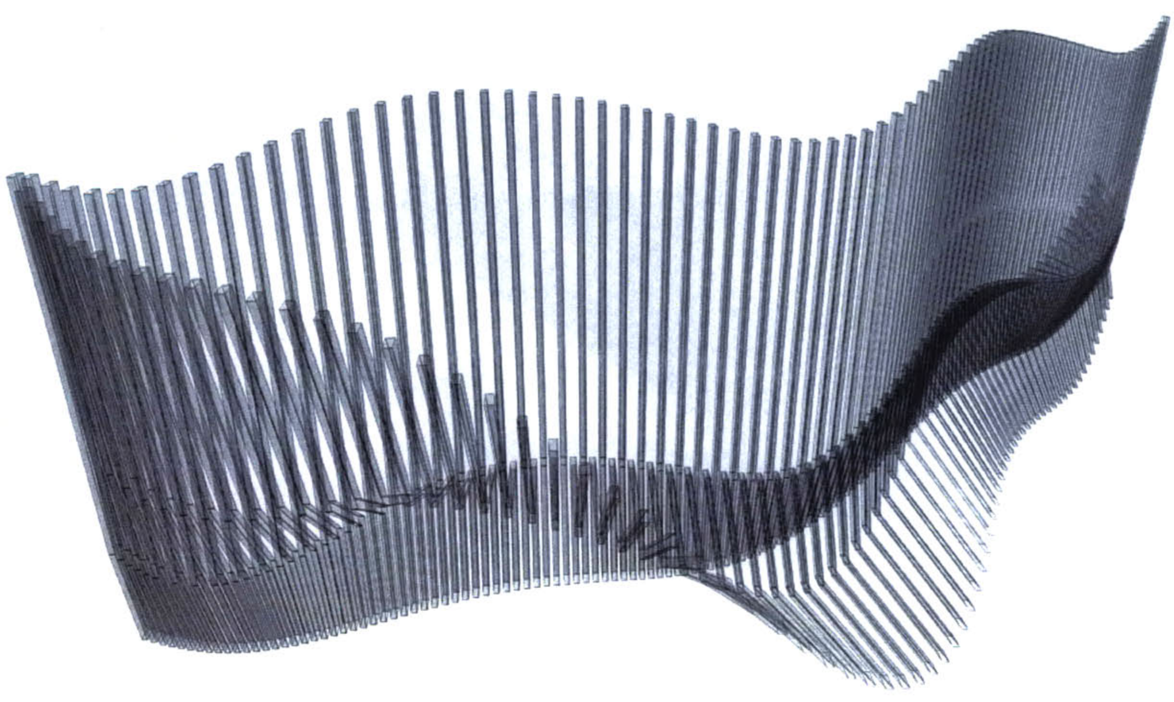


Figure 2.28: More Variations .



The Concept Of The Project

Because I did not have any holistic preconception about the design of the space when I started developing the algorithm, the challenge became how to apply the variations attained by the algorithm into a design project.
I started by looking at different examples of how the structure of a mosque is constructed. The most wide-spread typology is a big hall buttressed by columns. I adopted this typology but add more intricacy to the arrangement of the supporting columns.
It is spiritually more rewarding to pray in the first row. This experience was intensified in the design by granting the first row the maximum disruption that fades away to reach a calm wall that announces the entrance of the project. This incremental increase of disruption resulted in a series of walls that morph into each other. Embodying different intensities of the prayers' postures, these walls became part of the worshippers that prayed with them.
The disrupted wavy wall permits concave spaces to formulate, creating niches referred to as *Mihrab* in Islamic architecture. What is traditionally a discrete concave node was incorporated in the sinuous design of the wall.

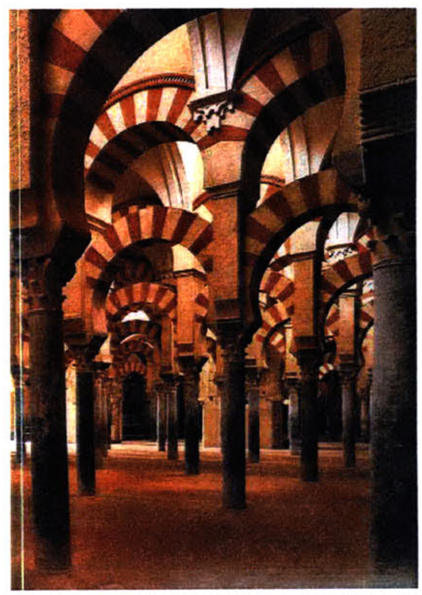


Figure 2.29: The intricate arrangements of the columns within a prayer hall in Cordoba mosque.

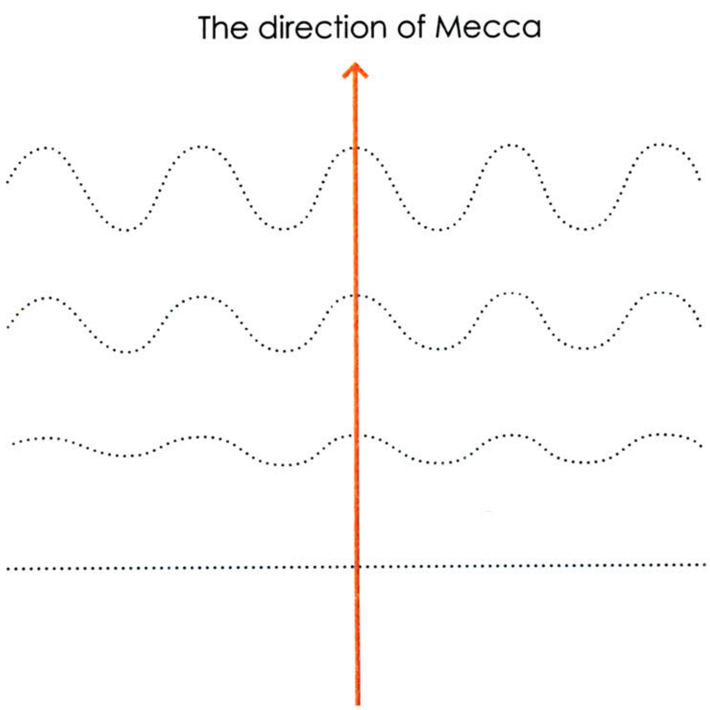


Figure 2.28: A scheme showing the arrangement of the walls with respect to the direction of Mecca.

Figure 2.30: Rows formed by worshippers during prayer.

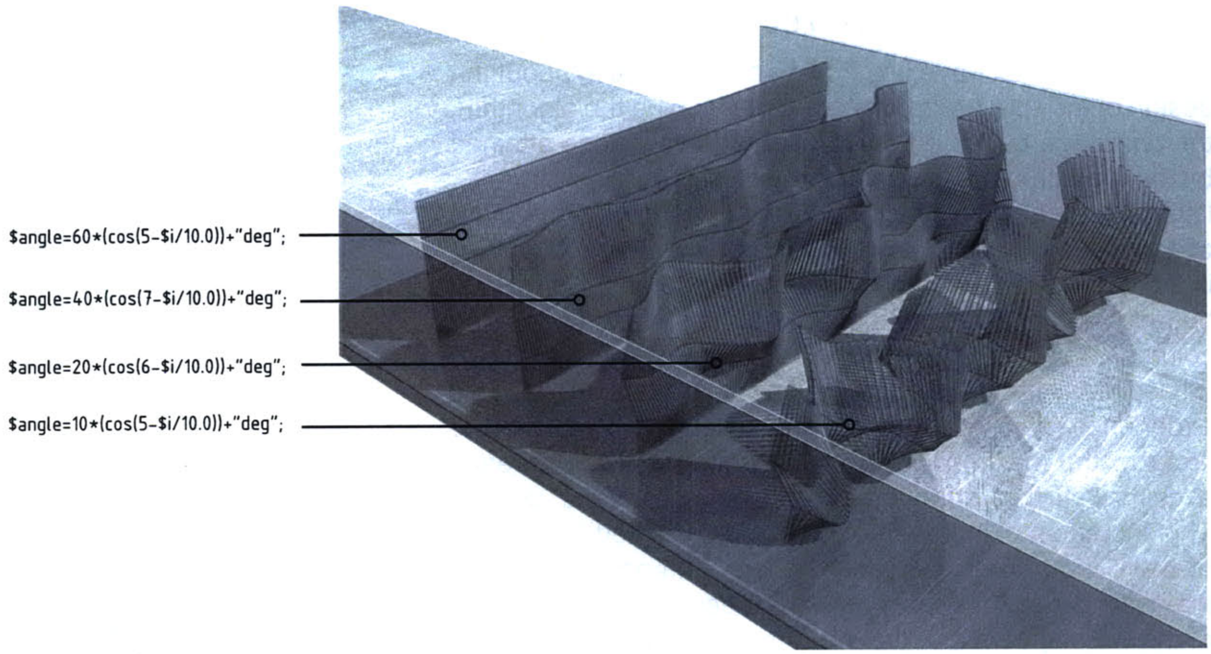
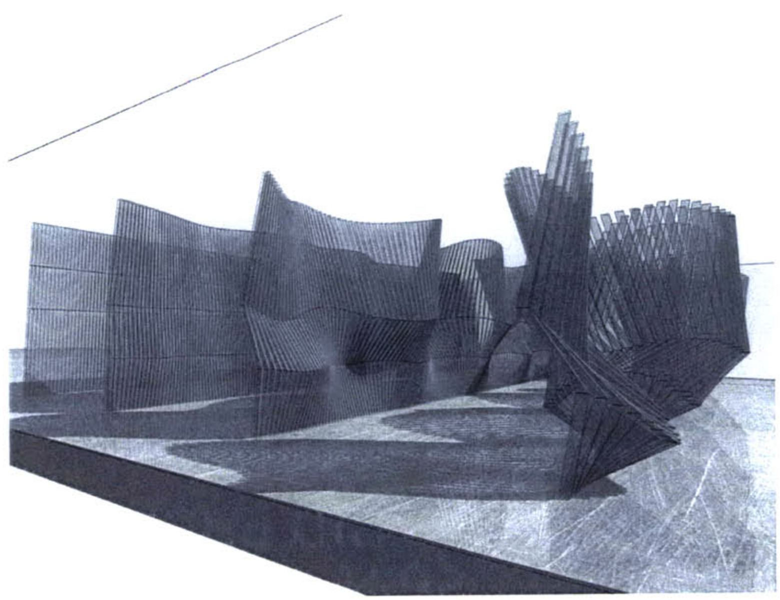


Figure 2.31: The image above shows the different values assigned to the rotational angles of each wall to give a sense of morphing between the walls.

Figure 2,32: An image showing the morphing between the walls.



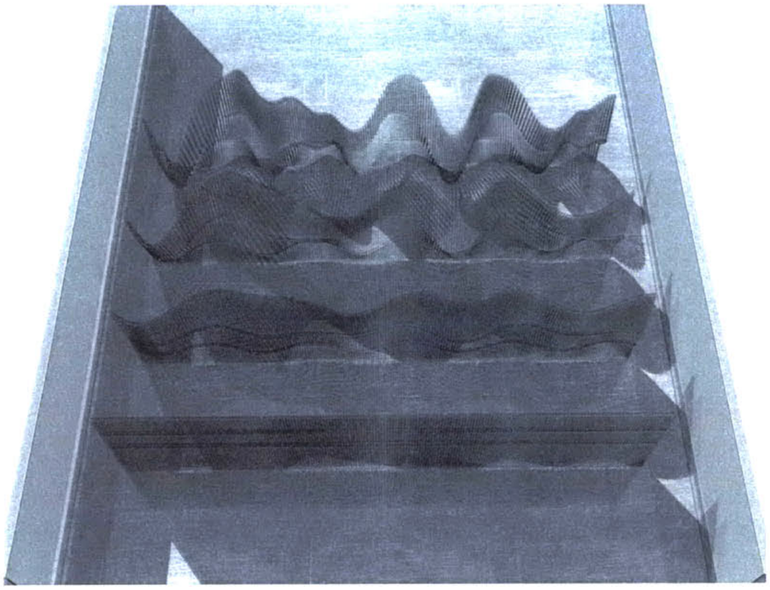
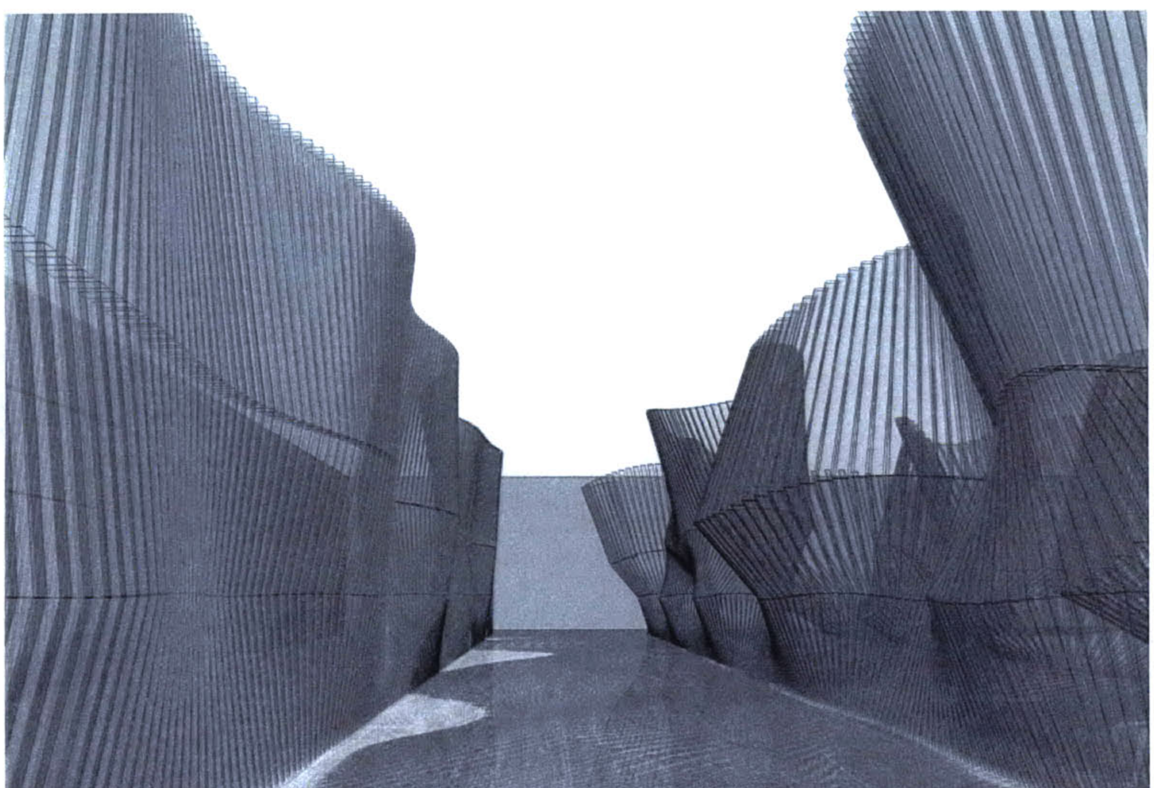


Figure 2.33: An image showing the morphing between the walls.

Figure 2.43: An image showing the dynamic space formulated between the walls



2.4. Critique

The First Machine

The two main problems in the First Machine can be attributed to Voronoi formalism and the abstraction of the algorithm: We could not go beyond the formalism inherent in Voronoi diagrams. We did not add enough sophistication to the algorithm, and hence the project ended up as an abstract Voronoi diagram. Because the algorithm was very abstract and was not related to the project itself, many compromises had to be made along the way to constrain the algorithm which consequently reduced many qualities of the design, mainly the spatial quality. The attempt to materialize forms generated by this generative machine brought with it substantial sacrifices.

The conceptual space of the algorithm was inadequately small. While we were initially thrilled by our results, it became increasingly boring and predictable towards the end. I felt that we were trapped rather than being offered more possibilities. Our experience was similar to what happened to Douglas Hofstadter, who wrote a computer program to generate English sentences. While he was excited in the beginning, he later became frustrated over the fact that all his solutions fell within a 'conceptual space' that it could not go beyond:

> At first it seemed very funny and had a certain charm, but soon it became rather stale. After reading a few pages of output one could sense the limits of the space in which the program was operating; and after that, seeing random points inside that space – even though each one was 'new' – was nothing new. This is, it seems to me, a general principle: you get bored with something not when you have exhausted its repertoire of behavior, but when you have mapped out the limits of the

space that contains its behavior.[1]

The Second Machine

Unlike the previous project in which the software implemented an already existing algorithm, the algorithm for this project was specifically designed based on contextual research done for the project. Through a systematic development of the algorithm, I was able to explore many variations quickly and precisely. Because the algorithm was designed without a pre-existing knowledge of how the spatial configurations of the project would be, the final design was limited to a repetition of walls. Although I was able to get many exciting variations, I could not utilize them in the project because I could not handle them spatially.

Although I was able to develop the algorithms into a versatile controlled condition, I still felt that I was over-constrained and confined within a very limited space of solutions despite the variability permitted by the algorithm. However, the advantage of producing many precise variations quickly through adjusting the parameters can not be overemphasized.

The questions that have to be asked after completing these projects are: Did these machines allow me to produce something that I could have not produced before? Did these machines help me to exceed or extend myself? Does the problem lie in the approach itself or in designing bad machines – as per Arnheim's hyperbole, "an incorrectly built engine can blow up a factory"[2]? Is it because I had used only one formal system in each project? Is it because all my attention was afforded to adjusting the parameters?

I have to acknowledge the fact that I would not have designed anything similar to what I have designed using generative machines were I to have followed intuitive ways of design. But different does not necessarily mean better.

With only a few snapshots of excitement, the whole process was hectic and did not satisfy my expectations. A series of compromises had to be made throughout the process of constraining the algorithm.

Next step

Because of this, I wanted to interrogate previous generative machines to find answers for many questions that arose in the previous two examples.

- How can we use rule-based systems without sacrificing meaning or function or the humanistic touch?
- How can we address contextual parameters without a loss?

(Endnotes)

[1] D.R. Hofstadter, Godel, Escher, Bach: an Eternal Golden Braid, 1979, p.621

[2] Arnheim, Rudolf. 1977. *The Dynamics of Architectural Form: Based on the 1975 Mary Duke Biddle Lectures at the Cooper Union*. Berkeley: University of California Press, p. 163

Chapter 3: Generative Machine versus Context

3.1 Introduction

1- Given a system, how can we assess the forces which act upon it and arise within it?

2- Given a set of forces, how can we generate a form which will be stable with respect to them?[1]

These two questions asked by Christopher Alexander in his not very-known article, "From a Set of Forces to a Form," encapsulate the topic of the second half of this thesis. The aim is to interrogate some generative processes based on these two questions. The first question concerns the representation of contextual parameters, both exogenous and endogenous. Because these processes rely on abstraction to compute designs, it is important to see how these mechanisms address contextual parameters. The second question concerns the generative engine itself, whose task is to generate forms that address the representation of contextual forces.

The generative systems that I am going to examine in this thesis include: Jean-Nicolas-Louis Durand's compositions, Peter Eisenman's transformational rules, George Stiny's shape grammars, Christopher Alexander's relational methods, and Greg Lynn's Maya expressions. These processes were selected because they employ rule-based systems throughout their respective design processes. Hence, the generative formal engine is real rather than metaphoric.

All these processes relied heavily on an interdisciplinary repertoire of ideas. The inventors of these processes had great ambitions rather than a mere aesthetic quest. Jealous of science's seemingly objective methods, the inventors of these processes wanted to establish a science of design; they wanted to establish logical procedures that can assist designers. They wanted to make life easier for architects by creating this science.

First and foremost, I must acknowledge the difficulty of juxtaposing these generative systems according to certain criteria. Which is more important, the way in which the contextual parameters are represented, or the interpretive engine that is used to translate these forces into form? In these processes, two conditions of the context can be noted: explicit and implicit.

In all of these processes, there is a strong connection between how the context is being represented and how the engine is created for that purpose, thereby necessitating a classification according to context. The representation of the context and the engine are deeply interrelated so that it is difficult to talk about one without the other.

Although representation of the context largely defines the generative engine, the emphasis is usually focused only on the engine itself. For instance, we can use an algorithm to design the form by representing the context as a mathematical formula; we can use an analogical method to design form by physically simulating the forces of the site. Shifting the focus towards the context will yield more information about the applicability of each generative process in real-world contexts.

This classification allows for us to talk about two kinds of generative systems: linguistic and biological. In the linguistic model, the emphasis is directed towards the syntactical rules that govern semantics; the knowledge about the context is encoded in the syntactical rules; and the difficulty lies in acquiring the skill to encode this knowledge at the beginning of the process. In biological generative systems, on the other hand, the emphasis is shifted towards achieving a metabolic balance between the generated form and the environment, and requires simulating the environment within which the form is being embedded. This requires reducing the contextual parameters into simulatable objects. The linguistic model is geared more towards analysis while the biological model is geared towards synthesis.[2]

This classification also allows for us to talk about spatial versus numerical computation. The linguistic processes lie in the spatial realm where the computation is done directly on shapes. On the other hand, most of the biological processes rely on numerical computation to execute the process.

Privileging the context over the engine allows us to talk about a lineage of pre-computer and computer processes together. It should not be viewed as an attempt to devalue the role that the computer has played in enhancing generative systems. Most contemporary architects tend to talk about the computer revolution in design while negating the previous endeavors to systemize and logically respond to a particular context.

A crucial point to investigate in these processes is whether there is a direct correspondence between the forces and the form. If there is, we face the problem of determinacy. If there is no correspondence, we will describe the engine as being flawed, and thus mapping the forces would be useless. This opens the discussion to linear and non-linear processes, whether the form is directly prescribed by the forces or not.

How can we benefit from these two approaches: the linguistic and the biological? The former concentrates on issues of syntactical formations and communicating meaning, while the latter concentrates on achieving the metabolic balance between the generated form and its environment. Both approaches have advantages and disadvantages that could be extracted and discussed.

Hopefully by drawing a larger picture of these processes, I will be able to disentangle common issues pertaining to generative processes. Inspecting these processes, to my belief, is very crucial in understanding how rule-based systems could be applied in the design process.

3.2. Context and Representation

Representations of any kind, most clearly geometrical and numerical, entails reduction, a reduction that seeks to produce the same reality but with less information. It is called reduction because it reduces the information of a given reality at the expense of obliterating either redundant information or many subtle differences. Representation is a process of 'recoding' as described by Herbert Simon:

> By appropriate 'recoding,' the redundancy that is present but unobvious in the structure of a complex system can often be made patent. The commonest recoding of descriptions of dynamic systems consists in replacing a description of the time path with a description of a differential law that generates that path.[3]

In design, abstracting the context is mandatory because it is impossible to design within the same physical environment of the design problem. To elucidate the impossibility, Alexander cites a simplistic yet interesting example of what he calls the analogical method to deal physically with the forces found in a certain context. To distribute rightly the furniture in a living room, he suggests placing movable furniture. After a week or so, the positions of all the pieces would stabilize and take permanent positions. The issue becomes searching for the most comprehensive and objective representation.

It is less expensive to operate on highly abstract descriptions of objects. Through the design process, different levels of abstraction can be used: line drawings, sketches, and diagrams. Abstraction, however, bring with it a very crucial problem: How to transfer the solutions accomplished in the abstract space to physical reality. This is why most projects that have been produced by a generative system remained in the abstract level.

Can there be an absolute representation of context? Can this objective representation be codified? Why do we need an objective representation?

(Endnotes)

[1] Alexander Christopher. “From A Set of Forces to a Form.” In Kepes, Gyorgy. 1966. The man-made object. London, Studio Vista, p.98

[2] Terry Knight referred to this distinction in her article: “either/or → and” in Environment and Planning B: Planning and Design 2003, volume 30, pages 327 - 338

[3] Simon, Herbert. 1969. *The Sciences of the Artificial*. Cambridge, Mass.: MIT Press, p.209.

Chapter 4: Linguistic model

Structuralism and linguistics played a significant role in inventing and enhancing some of the generative design processes. Structuralism has been advanced as an alternative to atomism, which had dominated the world in previous times. Unlike atomism, structuralism stressed the **relationships** between the parts rather than the constituent elements themselves. Structuralism had its origins in Ferdinand de Saussure's work on the structure of the language system. Later on, the applications of structuralism were deployed to other disciplines.

Structuralist thinkers, following Kant, claimed that the mind does not passively receive the sensory experience; instead, the mind imposes its own structure on this flux of the sensory experience. These innate structures are given at birth. Noam Chomsky argued that the human mind has an innate universal set of linguistic rules. Structure is an active matrix through which experience is filtered.[1] Structure denotes the inner forces that drive the external form.[2]

The generative or transformational grammars invented by **Chomsky** in his "Syntactic Structures"[3] played a paramount role in developing the architectural generative processes that took linguistics as a main impulse behind their formulations. Chomsky argued that the formal structure of a language could be reduced to 'a kernel sentence.' By recursively applying a limited number of transformations on the kernel sentence, we are able to generate all possible sentences in a language. He aimed at devising a mechanism that could explain how the speaker of a language generates sentences.[4] How Chomsky's model of language made its way into architecture is demonstrated by Eisenman and Stiny.

Three processes were chosen because they denote the clear tendency towards having a design process that is based on explicit syntactical rules. Durand was one of the first people to work within a systemized formal system but without setting up a clear mechanism for doing this. Eisenman, in his early houses, also worked within a formal system, but also without setting up a system that can be transmitted or taught. It is only until shape grammars were invented that a design process based on explicit rules could be accomplished and many variations can be produced. Shape grammars marked the first spatial computational paradigm.

Many authors have already talked about the genealogy of these processes. Juxtaposing these processes with the linguistic model would help to establish a dialogue with the other processes mentioned in the biological model.

An important thing to note is that each of these processes worked within a certain style that permeated their explorations, although one of their main aims was to displace styles in architecture by working within an abstract environment. Only shape grammars stand out as an exception because not much architectural aesthetics are attached to this formal system. My aim is not to look at these processes as styles, but rather as a generative process that can be applied onto any project.

Another important note is that all of these processes applied rules-based systems to break out from the previous architectural heritage and to explore different formal potential.

4.1. Jean-Nicolas-Louis Durand's Compositions

In 1795, Jean-Nicolas-Louis Durand was appointed professor of architecture at the newly established École Polytechnique. His pedagogy still influences architectural discourse. Through his pedagogy, Durand emphasized on reason and the exclusion of metaphysical concerns. His book the Précis of the Lectures on Architecture Given at the

École Polytechnique soon became a classic of architectural education after it was published in 1802-5. [5]

His point of departure, like his teachers from the generation of 'revolutionary architecture,' was the exhaustion of the classical tradition based on the teachings of Vitruvius.[6] His generation wanted to break away from the Vitruvian tradition that is very rigid and does not espouse diversity shown in exotic architectures.

Durand's aim was to maintain the autonomy of the architectural discipline, trying to find a method specific for design. Troubled by how to teach architecture -- especially in Ecole Polytechnique which reigned at the forefront of mathematics, physics, engineering, chemistry, and other scientific disciplines -- he sought out a methodology of teaching architecture. He wanted to systemize architectural knowledge based on a scientific basis through graphical means. According to him, creating a science of architecture meant devising general principles and negating styles at the same time.

Collecting, classifying, and analyzing the buildings from the past was the first step to deduce general principles. Buildings were classified into the functional, the historical, and the formal. Because he was obsessed with finding general principles, he sometimes presented an approximated, regular, and geometric version of historical building's plans. Some differences were eliminated for the sake of grouping and devising these general principles – a process of regulation and simplification.

His elements of architecture consist of the simplest elements such as wall and columns that combine to define the parts of the buildings, or parties. The parties combine to define a building. He used abstract and physical notations to refer to these parties.

Generative Machine

By codifying architectural knowledge in the form of a method it becomes objective: it can be transmitted to and be applied by other architects; in other words, it becomes scientific.[7]

Durand developed his method after the analytical methods developed by Locke, Condilla, and Condorcet. Durand elucidated his method of composition through step-by-step procedures. His method applies to neo-classical buildings. The process consists of six steps. The first step starts by setting up the main axes of the composition, followed by adding a grid of secondary axes that complement the primary ones. Placing walls along these axes and columns between the walls comes next. The fifth stage involves adding the stairs, porticoes, and other architectural elements in the plan. In the final stage, sections and elevations were drawn out of the plan.

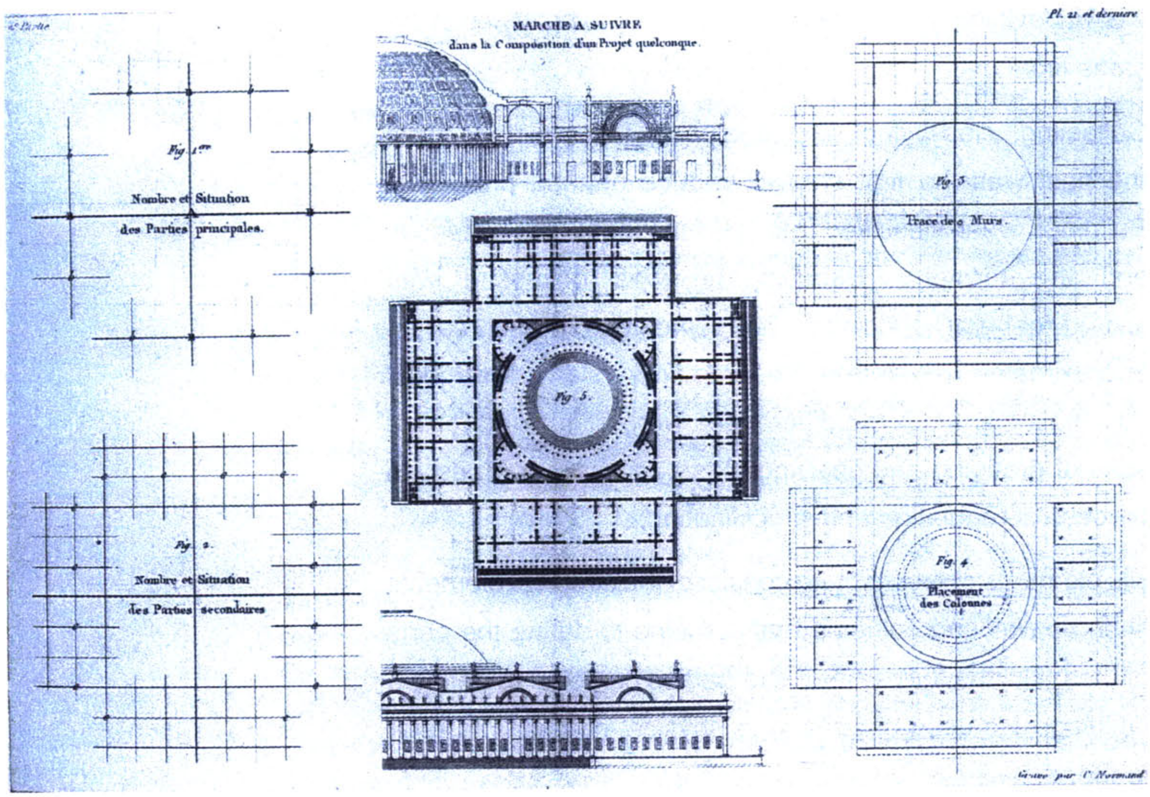


Figure 4.1. Jean-Nicolas-Louis durand, architect
(Procedure to be followed in the composition of any project)
(From Durand, Jean-Nicolas-Louis. *Précis of the Lectures on Architecture,* p.43)

Notes

Despite the claim that his process was combinatorial, Durand's process was predicated on transforming geometrical configurations into architecture. He started with geometrical schemes to avoid any stylistic references, which underwent a process of 'architecturizing' – adding walls to the axes. The geometric scheme became the generator of the architectural form rather than the combinatorial system.

Other sciences depended on numbers and mathematical operations. In architecture, however, there was no abstraction. This is why Durand resorted to geometry to aspire towards a science of architecture. Geometry provided him with the right tool to compose architecture abstractly. However, it was at a very low level of abstraction and that made it easy for him to take his finished geometrical configurations back into architecture.

In some of his explorations, he reached two different plans from the same starting point. Although the process is procedural, there were no clear rules of transformation. The computational process Durand employed was vague and could not be generalized outside the neoclassical style.

4.2. Peter Eisenman's Transformational Rules

> The idea of a transformational practice operating on a singular, well-defined language was abandoned for research into the nature of language genesis itself. The rules represented in classical architecture by orders, the notion of beauty, and rational Cartesian principles are abandoned and replaced with an ideology that stresses the importance of relationships rather than shapes, leaving architecture with a highly diffuse lexicon and an entirely new syntax.[8]

Eisenman relied heavily on the generative grammar of

Noam Chomsky to formulate his own argument that every form embodies the rules of a particular formal language. In his Ph.D. dissertation, "The Formal Basis of Modern Architecture," and in his later house projects, Eisenman presented the idea of a formal language that is based on explicit syntactic rules for creating new buildings. His work was a research in the syntactical dimension of architecture, excluding completely the semantic one, and emphasizing the relationships between the elements rather than the elements themselves. According to Chomsky, the syntactic structure of a language is the generator of that language. This grammar, as structure, is not inert but generative and transformational. Eisenman's process is rigorous both theoretically through his texts that accompanied his projects, and practically through showing step-by-step meticulous procedures.

Contextualizing

Eisenman developed his early houses to displace the 'metaphysic' of architecture, to work as freely as possible from functional and historical constraints. He wanted to infer new and unpredictable experiences in the house and to look for new ways of conceiving the architectural 'occupiable form'. He wanted to displace Modernism which was inscribed within the classical metaphysic of architecture because it glorified function as the foundation of architecture. In other words, Eisenman's architecture is acontextual insofar as his buildings do not refer to anything (user, structure, meaning, etc.) outside of themselves; it is an architecture that is devoid of meaning. In his anti-functionalism attitude, Eisenman tries to enclose meaning within the form.[9] The form generates itself from the inside. It is a syntactic system to repress any external influence.[10] Even his books with their critic's texts formulate an enclosed system that looks inward. Dismissing meaning, according to him, opens up more spatial possibilities that were masked by it.

To displace the traditional and existing metaphysic of architecture in his early houses, he wanted to design autonomous self-referential objects that negated their contexts, an architecture that refers inwardly to its intrinsic rather than extrinsic condition. The object itself is the record of its history, its morphogenesis. To make objects new again, a process of stripping objects from their 'acculturated meanings' was required. It was a process of recoding these signs by connecting them to other signs. Only by challenging and negating that 'comforting metaphysic' and symbolism of shelter attached to the house, can new possibilities of dwelling emerge that have previously been oppressed by that same metaphysic.[11]

Semantics derive their meaning not from themselves but rather from their interplay and relationships to each other. But the work has to function at the end. Meaning cannot be negated completely. "Unlike the formation of traditional signs which are coupled with actual or virtual functions and thus read as doors or rooms, **they are not generated from any functional logic**, and so in order to become architectural, that is, to avoid being merely sculptural, they must postulate an alternative syntactic system which still serves as a support for such functional meaning."[12]

The form-making process does not stem from traditional constraints – function, program, meaning, technology and client – but from a transformational process that negates these constraints. "The rationality of process and the logic inherent in form becomes almost the last 'security' or legitimation available."[13]

Preconceived image

Eisenman wanted to displace the preconceived image (both form and type) paradigm by providing a process-driven one. The preconceived image paradigm implies that the architect starts the design process with an initial form-image that undergoes a process of refinement according to the particularities of a given project. "This initial image describes and limits the actual choice from the range of alternatives."[14] On the other hand, the transformational process would expose and reveal more possibilities that are masked by the preconceptions of the designer. The perceived image narrows down the scope of choice while in the transformational method; each step opens up more possibilities as the process unfolds into different directions.[15] It is like searching for something that did not exist prior to the design.

Generative Machine

> Therefore, self-generating, transformational design processes that would perforce distance both the architect and the architectural history by ignoring cultural conventions were employed in the early houses.[16]

His process is predicated on transforming a generic form into a specific form through a series of transformations. While his generic form meant a platonic form imbued with its own inherent laws – that is, geometric forms potentially embody the transformational rules that will transform them into specific form – his specific form came to mean the form that is shaped and actualized in response to a specific intent and function. The generic form can be either linear or centroidal, for example a cube or a sphere. A double cube or a cylinder is linear.[17] The cube itself has two different starting conditions: the solid and the void. Each condition yields different paths of form evolution. While both conditions produce a figure, the solid one implies subtraction, the void one implies addition.

The transformational process is a step-by-step procedural model, a "logical formula," that unfolds following the internal properties of the starting shape, the neutral cube. The object itself weaves itself from inside without any external or functional constraints. This weaving process was more significant for Eisenman than the product itself. This process was intended to give a rational explanation of the process itself. The generated object does not have any history but its own morphogenesis. He also wanted to destabilize the notion of scale that was heavily institutionalized based on human dimension, as he wanted to dislocate human relationship with architecture. The resultant object is independent and does not comply with the 'vectors' of mankind.

The termination of the process is an internal consequence of the system of transformation when the process exhausts itself and no further steps are possible, rather than a

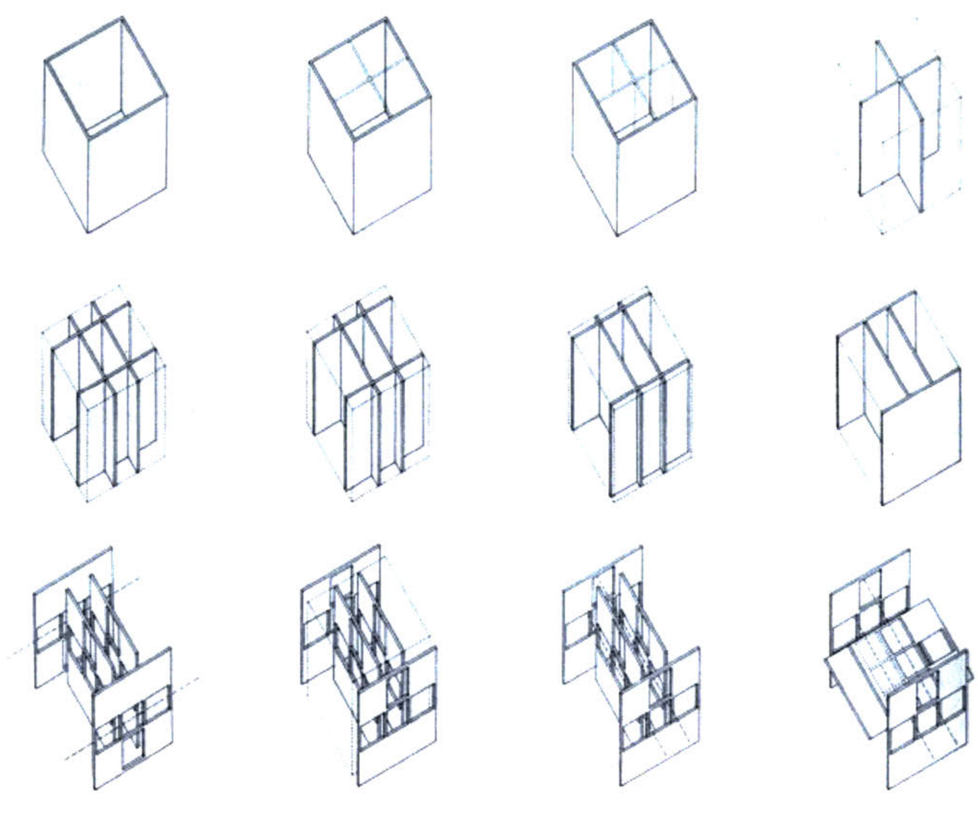


Figure 4.2. Peter Eisenman, House IV.
(From Eisenman, Peter; House of Cards, p.68)

decision adopted by the designer.

He intended the process to be a linear transparent process that can be reconstructed from the starting point based on its visual spatial linearity. This linearity allows for us to see that any element is derived from another element in a previous process. His aim was to reveal the process to the user, allowing the process to communicate its morphogenesis.

Subject

"... [I]n Eisenman's work the subject has been there from the beginning; its presence is not excluded as in the lexicon of Durand, or the classificatory theory of Saussure, but, with its architectural knowledge it acts more like that quasi-theoretical subject of Chomskian linguistics."[18] For Eisenman, the subject creates the syntactic process and steers the derivation of the process by means of intuition. The subject is not reduced to zero.

Eisenman hoped that creating an autonomous transformational process would yield an autonomous object wherein its meaning is enfolded within itself, which would displace all history but its own generative history. As the process propels forward, the object distances itself from its author who is charged with cultural and aesthetics prejudices. The object is being steered by the transformational process itself and not by the author. After the architect sets the object in motion, the object looks away from the author towards a new condition embedded in the nature of objecthood.[19] This can be understood by the fact that Eisenman would have not designed a house like this without following such process. The process does not start from a preconceived image in the head of the designer. This ambiguous relationship between the subject and the object is interestingly defined by Eisenman as a parallelism of existence: "... [S]ince the object does not necessarily mirror, confirm, or deny the architect's existence, it assumes a condition of parity with him. The new distance, then, is a parallelism or equivalence of existence: object and architect, two non-intersecting but interrelated entities."[20]

Although Eisenman claimed the process to be unauthored and autonomous, his interventions are very conspicuous at every step of the process. Tafuri comments on this issue: "in the folds of Eisenman's 'absences' hide easily recognizable 'presences'."[21] However, we cannot deny the fact that Eisenman worked against the explicit subjectivity of the Modern Movement.

Notes

In Eisenman's words:

> First, can any architecture be totally devoid of a general set of preferences, which amounts to asking whether it is possible to totally deny the influence (conscious or unconscious) of a historical or personal predisposition – i.e., style? And second, can any architecture, if all architecture deals with problems of shelter, gravity, entry, etc., be without certain recognizable characteristics, which inescapably derive from the forms of these problems?[22]

He admits that the proclaimed endless variety that could be produced by his transformational process collapsed because the process works within a very limited vocabulary. His ambition to create a universal process that would encompass all of architectural production fell short due to a paucity of universality.

His process is not computational per sé. There are no explicit rules that can be extracted from his process. The starting shape itself mandates certain rules of transformations according to its internal configurations. The rules are not set up *a priori* but are created as the process unfolds.

Eisenman's process is not combinatorial, but rather derived mainly from the internal logic of the kernel form. It is a topdown process that starts with a cube, which is supposed to function only as the initiator of the process, which is then subjected to a recursive application of transformational rules which unfolds according to the internal logic of the cube itself. The presence of the cube is very clear within the end product despite the transformational rules. It is a process of adding more refinements and details to the starting shape. All these transformations are circumscribed by the boundaries of the cube. Only later on, did Eisenman introduce new operations like decomposition to break out of this cube, a process of decomposition and fragmentation.

Is his process prescribed by his theory, or it is the other way around – theory prescribed by process? Because his aim was to displace the metaphysic of architecture, he employed such a process. Or is the process itself that prescribed his theory? Every syntactical approach to architecture would imply displacement of the 'metaphysic' of architecture.

Besides its dysfunctionality, did his process produce novel domestic experiences or novel formal language that could have not been produced otherwise? In the least, this process helped him break away from the previous architectural heritage if not completely.

The formal rule-based system that he had set up was a very self-constrained system that mandated its explosion or 'decomposition'. The strict closure of his system exploded in House X. In House X, the sequential and reversible reading of the building is impossible. It is not possible anymore to reconstruct the process by entering a reversed situation. House X marks the shift from linear to nonlinear processes in Eisenman's work. Thus, the object can be predicted neither from the starting point nor from each step throughout the process. "An object which while attempting to retain the boundaries of the syntactic domain nevertheless works to criticize the original forms of Euclidean geometry, the Cartesian spatial grid, in order to open up the system to the new, ideologically based shapes.

This object becomes House X."[23] This decentralized house, a set of four fragments with the loss of center, denotes the loss of unity, an empty center.

Unlike the previous houses, this process of decomposition does not start from zero – that is, the neutral cube – but from a given formal notions. Decomposition is used here more as an analytical tool that starts with a given shape to be deconstructed and fragmented.

What is interesting about Eisenman is his self-critical process through which he is always in constant flux, to an extent that he includes critical texts in his books or makes an imaginary dialogue between him and his critic.

4.3. George Stiny's Shape Grammars

Stiny and **Gips** began the Shape Grammars Project in 1972, based on the premise that design is a matter of formal composition, following the tradition of the Beaux-Arts, in the hopes of establishing "a science of design" and "a theory of formal composition." [24] Stiny was so enthusiastic in his ambitions as an advocate of the systematic design process, as to declare that using the rules will replace the traditional intuitive way of design and the designer will no longer need the 'creative inspiration,' the 'inventive flash,' or 'individual genius.' By so doing the designers will be able to answer the difficult question that concerns all designers: "where do designs come from?"[25] Putting exaggerated ambitions aside, shape grammars aimed first at externalizing the design process so that it can be modified and transmitted, and second at creating many variations to select from.

The computation is done visually with shapes rather than symbolically.[26] Shape grammars are a set of rules that can be applied on a starting shape to generate a set of designs. Shape rules have the form: A → B. The rule can be applied in a design whenever there is a shape that matches the left side of the rule. The rules can be addition, subtraction, or spatial transformations like translating, mirroring, and rotating. Within the same conditions, many variations can be produced by selecting different computations. Most of the synthetic applications of shape grammars have dealt with the design problem in a trial-and-error fashion through applying rules to generate candidate solutions, followed by an evaluation mechanism which determines whether the current derivation would fit the design problem.

Since their inception, shape grammars have been used widely for both analytic and synthetic applications. Computation with shape has an advantage over numerical computation; shape grammars rely on the basic geometrical components of points, lines, planes, and volumes to do the computation and this makes the process less abstract than working with numbers.

Several improvements have added more complexity and sophistication to shape grammars, mainly color and parametric grammars. Parametric shape grammars allow the designer to interfere in each step by adjusting the parameters, hence providing more control to the designer. Color grammars constrain the rules to behave in certain ways desired by the designer to respond to certain

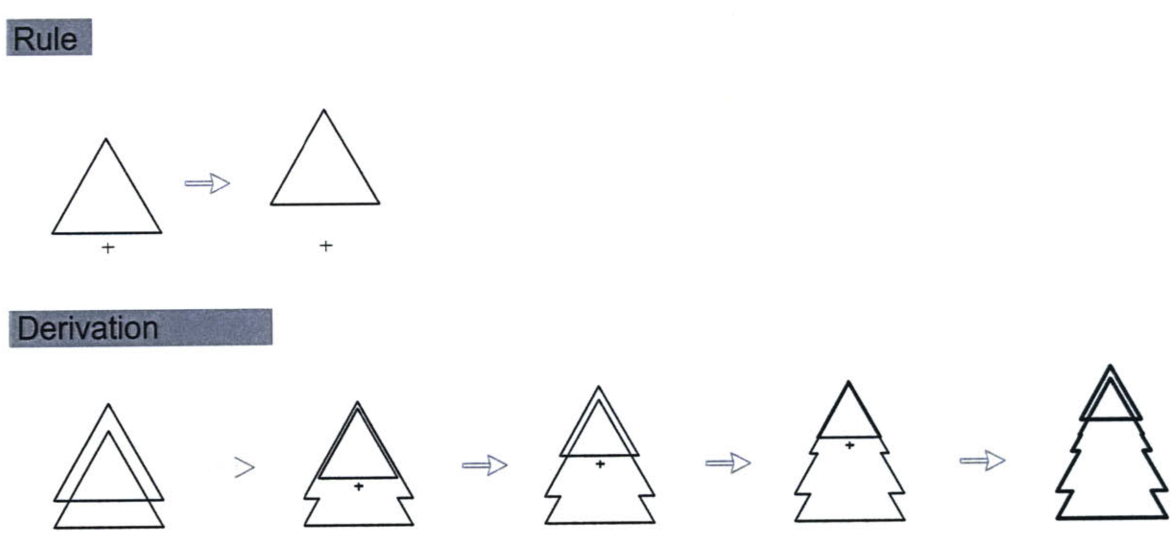


Figure 4.3. Shape Grammars.

constraints imposed by the design problem.

Difficulties in Design – Context

> The shapes and spatial relations used to compute designs often have implicit meanings and functions in the same way that, in a conventional design process, the lines a designer puts down on paper have meanings.[27]

Through analyzing the work of Kandinsky and Klee, Knight mapped out how computation addresses the dualisms that Kandinsky and Klee tried to expose: analysis-synthesis, form-content, calculation-intuition, emergence-predictability, and intelligibility-productivity.[28] I am going to adopt some of these dualisms to structure my argument.

Form-content:

As the design activity (intuition, inspiration, and guesswork) is shifted towards designing the algorithms rather than the form itself, knowledge about the context should be encoded *a priori* within the grammars. The grammars rather than the form should embody the context (aesthetics, expression, meaning, and purpose). The ways in which the rules are set up determine how the final shape would respond to this complex matrix of requirements. The rules restrict the range of combinatorial possibility. Since it is difficult to predict how the process is going to unfold, how can such knowledge be codified first, and then encoded in the rules? Most of the successful synthetic applications of shape grammars were limited to a few steps of computation that can be controlled and refined by the designer. This simple application of the rules gives the impression that the final design could have been achieved without computation. Every shape grammarian acknowledge the difficulty of such an act especially when the rules, most often than not, lead to unpredictable results. Accommodating this unpredictability

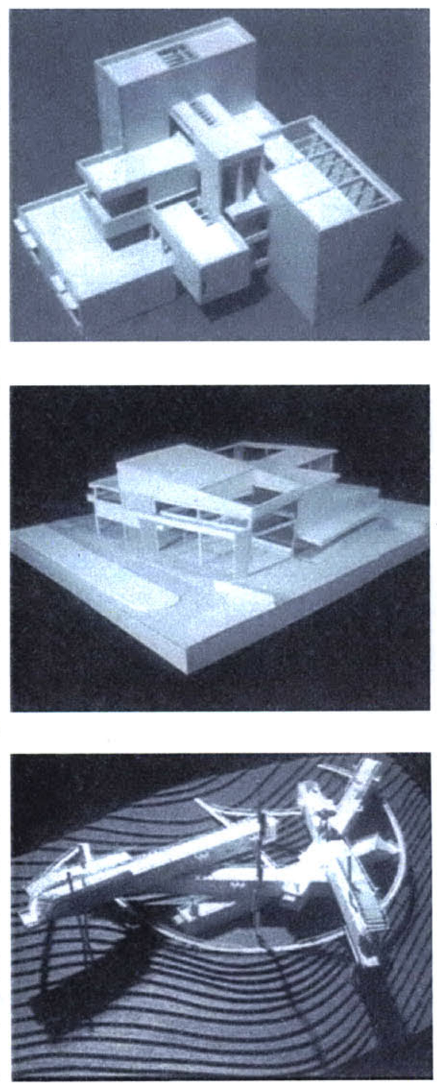


Figure 4.4. Projects designed using shape grammars formalism. (From Knight. "Classical and non-Classical Computation").

is accomplished by refining the rules until they fit. Another solution that has been experimented is combining grammars with genetic algorithms or simulated annealing for selecting the best variant. The difficulty of this option resides in the difficulty of codifying the selection criteria.

Mitchell, in his book "The Logic of Architecture,"[29] argues that grammars encode knowledge of how to compute or put together buildings that function adequately. "These rules encode knowledge of form, function, and the relationship of the two."[30] His aim was to theorize the plausibility of establishing a formal language for practical as well as poetic purposes.

Unlike Eisenman, who not only negated but wanted to displace meaning, shape grammarians were keen on finding ways to embed meaning within the rules.

Analysis-synthesis:

Where do the rules come from? Most of the synthetic applications of shape grammars so far had started analytically by extrapolating rules from a corpus of architectural objects that fall within the same formal language. These rules then serve as a starting point to generate designs that have the same 'style,' since these rules are imbued with certain formal and functional characteristics that give the generated designs the same formal identity. This would not have been possible without a process of approximation and obliteration of the subtle differences between many instances that belong to the same formal language. The process involves extracting and codifying these rules to be applied to a later one. These rules are taken for granted to work under any conditions and contingencies. It is a process of unlocking the secrets of great architects, a process of mastering these rules and embodying the original creator's authorship. The rules are supposed to encompass and generate all the variations. The question that perturbs this experiment is: Beyond its academic value, what is the value of generating an authentic Palladian Villas in this age? Can these grammars go beyond what Palladio had originally produced? The problem is that shape grammars already work within a closed grammatical system that can not go beyond Palladio.[31]

This is the exact opposite of typological process in which the emphasis is directed towards semantics rather than syntactics. The type arises as a result of fusing many instances which have formal or functional similarities. The type represents a response to a complex of ideological, religious, or practical demands in certain contexts. It was seen not as a model to be imitated, but as a principle allowing for infinite formal variations. It encapsulates historical experience. "The 'type' therefore, is formed through a process of reducing a complex of formal variants to a common root form."[32] Because of this, type is not neutral but has formal, functional, and symbolic connotations.

Shape grammars do not terminate in and of themselves; the designer has to interfere. This issue will be highlighted later when I discuss some of the biological processes in which the process terminates itself when the metabolic balance between the generated form and the environment is accomplished.

The computer applications of shape grammars highlight this problem. The speed of the modern computer allows many rules to be applied recursively for a large number of times. A designer may be able to handle up to five rules, but beyond that, it becomes complex. This makes it more difficult to encode knowledge about the context from the beginning.

Notes:

Is this shift of creativity towards designing the rules rather than the form worth it? Do shape grammars make life easier for architects? Do the variations help the designer? What does it mean to relinquish some of the designer's responsibilities to the rules?

(Endnotes)

[1] Piaget, Jean, 1896. *Structuralism*. New York, Basic Books.
[2] Madrazo Agudin, Leandro. 1995. *The Concept of Type in Architecture: An Inquiry into the Nature of Architectural Form*. [Zurich: The Institute], p.40.
[3] Chomsky, Noam. 1957. *Syntactic Structures*. 's-Gravenhage, Mouton.
[4] Madrazo Agudin. *The Concept of Type in Architecture: An Inquiry into the Nature of Architectural Form*, p.333.
[5] Picon, Antoine. "From 'Poetry of Art' to Method: The Theory of Jean-Nicolas-Louis Duran". *In* Durand, Jean-Nicolas-Louis. 2000. *Précis of the Lectures on Architecture; with, Graphic portion of the lectures on architecture; introduction by Antoine Picon; translation by David Britt*. Los Angeles, CA: Getty Research Institute.
[6] Ibid., p. 15
[7] Madrazo Agudin. *The Concept of Type in Architecture: An Inquiry into the Nature of Architectural Form*, p.222.
[8] Gandelsonas, Mario. "From Structure to Subject: The Formation of an Architectural Language". In Eisenman, Peter. 1982. *House X*. New York: Rizzoli, p.18.
[9] Ibid., p.8.
[10] Ibid., p. 26.
[11] Eisenman, Peter. 1987. *Houses of Cards*. New York: Oxford University Press, p. 172.
[12] Gandelsonas, Mario. "From Structure to Subject: The Formation of an Architectural Language", p.10.
[13] Eisenman, Peter. *House X*, p. 36.
[14] Ibid., p.36.
[15] Ibid., p. 40.
[16] Ibid., p.177.
[17] Eisenman, Peter. 1963. *The Formal Basis of Modern Architecture*. Ph.D. dissertation, p.13.
[18] Gandelsonas, Mario. "From Structure to Subject: The Formation of an Architectural Language", p.18.
[19] Eisenman, Peter. *House X*, p. 40.
[20] Ibid., P.40.
[21] Tafuri, Manfredo. "Peter Eisenman: The Meditations of Icarus". In Eisenman, Peter. *House of Cards*, p. 167.
[22] Eisenman Peter. *House X*, p.42.
[23] Gandelsonas, Mario. "From Structure to Subject: The Formation of an Architectural Language", p. 24.
[24] Stiny, George. 1980. "Kindergarten Grammars: Designing with

Froebel's Building Gifts," Environment and Planning B 3 (1980): p.461.

[25] Ibid.

[26] Knight, Terry. "Shape Grammars in Education and Practice: History and Prospects." <http://www.mit.edu/~tknight/IJDC/>.

[27] Ibid.

[28] Knight, Terry. 2003. "either/or → and". *Environment and Planning B: Planning and Design,* volume 30(3).

[29] Mitchell Mitchell, William J. 1990. *The Logic of Architecture: design, computation, and cognition.* Cambridge, Mass.: MIT Press.

[30] Ibid., p. 238

[31] Greg Lynn had talked about this subject extensively in his article, "New Variations on the Rowe Complex," in his book "Folds, Bodies & Blobs, Collected Essays."

[32] Argan, Giulio Carlo. 1963. "On the typology of Architecture". *Architectural Design*:12.

Chapter 5: Biological

While the linguistic model emphasizes the syntactical dimension of architecture and its ability to communicate meaning, the biological model puts emphasis on achieving the metabolic balance between the generated form and its environment. Environment is the key issue in this argument.

All of the evolutionary approaches base their argument on the fact that vernacular architecture produces more fitting and adaptable buildings. These buildings' forms have developed, through a persistent process of correction and over a long span of time, into a stable condition with respect to their contextual forces (environment and user requirements). The question that has concerned everyone who worked within this evolutionary process was how to transpose that model into our complex present-day context.

All of the processes mentioned below have a similar stance with respect to achieving fitness between the generated form and its context. However, two distinct approaches can be identified, which I term here as **bio-logical** and **bio-alogical**. The bio-logical is demonstrated by Christopher Alexander, one of the foremost architects to talk about the concept of achieving fitness between the generated form and its context. Alexander is the first to translate D'Arcy's argument – that 'the form is a diagram of forces' – into architecture. John Frazer, and later the Emergent Design Group (EDG) at MIT, developed processes that are based on a natural growth system and on Evolutionary Algorithms. These processes were aimed at creating processes that respond to the increasing complexity of any design problem. The bio-alogical is demonstrated by Greg Lynn's approach, which, although premised on anchoring the form within its context, aimed to legitimize an aesthetic quest and to seek novel formal constructs.

All the architects who worked within either of this biological approach did not produce architecture; rather, they produced abstract intricate shapes which were difficult to translate into architecture. Ironically, these architects wanted to respond logically to any given context, but ended up with diagrams that seemingly negated these contextual parameters.

5.1. Bio-logical

5.1.1. Christopher Alexander

> Here the design process is remote from the ensemble itself; form is shaped not by interaction between the actual context's demands and the actual inadequacies of the form, but by a conceptual interaction between the conceptual picture of the context which the designer has learned and invented, on the one hand, and ideas and diagrams and drawings which stand for forms, on the other.[1]

Context was a major concern for Alexander. Form and context formulate what he called an 'ensemble'. The goal of design was to achieve the 'fit' between these two constituents. In other words, the context dictates the form. "Every design problem begins with an effort to achieve fitness between two entities: the form in question and its context. The form is the solution to the problem; the context defines the problem."[2] This idea is biological par excellence. Alexander was intent on creating a direct correspondence, or one-to-one mapping, between the context and the generated form. He cites different examples to elucidate this point: the pattern formed by iron filings in a magnetic field, and the shape taken up by a soap film in response to internal and external air pressures.

He claimed that contemporary design processes were inadequate in producing forms that corresponded to the increasing complexity of design problems. Also, our human cognitive and creative capacity is limited, and hence we need a process to stimulate and enhance this creativity.

Contextualizing

In his book, "The Evolution of Designs,"[3] Philip Steadman dedicated one chapter to dissect the biological component of Alexander's argument. In his seminal book, "Notes on the Synthesis of Form," Alexander clearly argued for a rational, explicit design method to replace intuitive individualism. His rigorous theory was accompanied by a scientific method with particular mathematical means by which this might be achieved. His work was premised on the fact that unselfconscious process produces good (more fitting, better adapted) results while the self-conscious produces bad ones. For Alexander, an 'unselfconscious' process signified the design process that occurs in vernacular architectural contexts, while 'selfconscious' process signified the design process that takes place in our present-day and age when architects are educated to become specialized and professional. Alexander wrote, "I shall call a culture unselfconscious if its form-making is learned informally, through imitation and correction. And I shall call a culture selfconscious if its form-making is taught academically, according to explicit rules."[4] Unselfconscious societies developed more fitting and better adapted forms slowly over a long span of time.

To explain the adaptability of the form to its environment within the unselfconscious process, Alexander resorted to cybernetics and especially to Ross Ashby[5]'s machine, 'the Homeostat'. Homeostasis is the tendency of the body to self-regulate its internal equilibrium in the face of disturbances from the outside environment. The Homeostat machine was intended to simulate the homeostatic process of an organism. This system has what Ashby called 'essential variables' which express the range of values through which the machine can respond without making a radical change. When faced with a new condition beyond the essential variables, a series of switches move at random until they hit a configuration in which the machine adapt again. It is a process of searching for the best configuration in a trial-and-error process.[6]

In the context of vernacular architecture, when the primitive craftsman is faced with a 'misfit', he reacts by making some random modifications, but without imposing any designed conception on the form. It is a process of trial-and-error that awaits feedback from the environment. The system itself, as the Homeostat machine, is self-regulating. The unselfconscious process implies a very slow evolution through a series of persistent corrections. These corrections happen gradually. However, if many 'misfits' happened at the same time, the system will break down.

How can the evolutionary model be transposed from the unselfconscious to self-conscious processes? How can adaptability be achieved within a complex situation (self-conscious process) wherein all misfits are interdependent? How ca the time problem be overcome?

Alexander discussed some procedures that are required to accomplish this transposition. First, a representation or a model of that process is required to flesh out the form-context interaction that is necessary for achieving fitness – a virtual space to test the fitness of the form against its environment. The compressed evolutionary process will then be transferred to the head of the designer. The designer will consequently have "mental pictures" of both the form and the context; he or she can then test them against each other, as opposed to an unselfconscious craftsman who tests out this interaction physically. Alexander claimed that the designer will immediately resort to these mental images which are incomplete and incorrect. His aim was to correct these images in the designer's mind and to elevate these intuitive images into a higher level of understanding: what he called "formal pictures of mental pictures."[7]

Alexander strived towards giving a precise mathematical description of the design problem so that he can evaluate the fitness throughout the design process. He advocated the use of mathematics and logic, which for some architects has negative connotations associated with certain formalisms. "It is the business of logic to invent purely artificial structures of elements and relations... And then, because the logic is so tightly drawn, we gain insight into the reality which was previously withheld from us."[8] These logical structures of a design problem was thought by Alexander to facilitate a better understanding of its complexity, and that, in turn, will prescribe a form that fits that complexity. Abstracting the design process by converting the problem into mathematical symbols, according to Alexander, also helped the designer to feel neutral about the problem, which will eventually lead to generating new solutions.

The structure of the design problem can be determined by a process of hierarchical decomposition stemming from set theory. By decomposing the complex problem into 'misfits', the designer can come to understand the holistic structure of the problem and make changes to correct these misfits.

Like Durand, who wanted to systemize architectural knowledge by classifying and regulating building types, Alexander wanted to establish a cumulative scientific repertoire of patterns that allow behavioral tendencies to coexist without conflict. These patterns are pictures which can depict anything spatial. Design, for Alexander, was based on predicting all potential conflicts and then defining the patterns that prevent these conflicts, and then combining these patterns into forming a cohesive whole. In this sense, his approach is a rational, constructive, and evolutionary one.

Generative Machine

In his book, "Notes on the Synthesis of Form," Alexander hinted at deducing the form out of the diagram of forces. Later on, this concept, borrowed from D'Arcy Thompson who referred to the form as "a diagram of forces," was crystallized and further developed into his article "From a Set of Forces to a Form".

In that article, Alexander initiates his argument by asking: "Given a set of needs, how can we generate a form which meets those needs?"[9] Troubled by the notion of need and driven by a desire to objectify contextual parameters, he replaced it with the notion of force. He believed that the concept of force had the potential of encompassing the complexity of a given context. He defined force as: "an inventive motive power which summarizes some recurrent and inexorable tendency which we observe in nature."[10] By so doing, Alexander claimed that the notion of force enables us to encapsulate and summarize all forces embedded within a given system: human (in terms of needs), mechanical (in terms of Newtonian forces), thermodynamic (in terms of thermodynamic potential), and social (in terms of social forces).

In simple natural systems, there is a direct registration of these forces upon the form. Alexander mentioned the example of the bumpy sandy surface formulated directly by the interaction of five forces. In an artificial system, however, a form emerges to stabilize the contextual forces only by means of design. Thus, to design means that all forces within a system should be assessed to be later used in generating a form. This represents an unequivocal correspondence to D'Arcy's thesis.

Alexander listed three methods to generate a form out of this diagram of forces: **numerical, analogical,** and **relational**. He privileges the relational method over the other two, which are, according to him, too simple and cannot handle

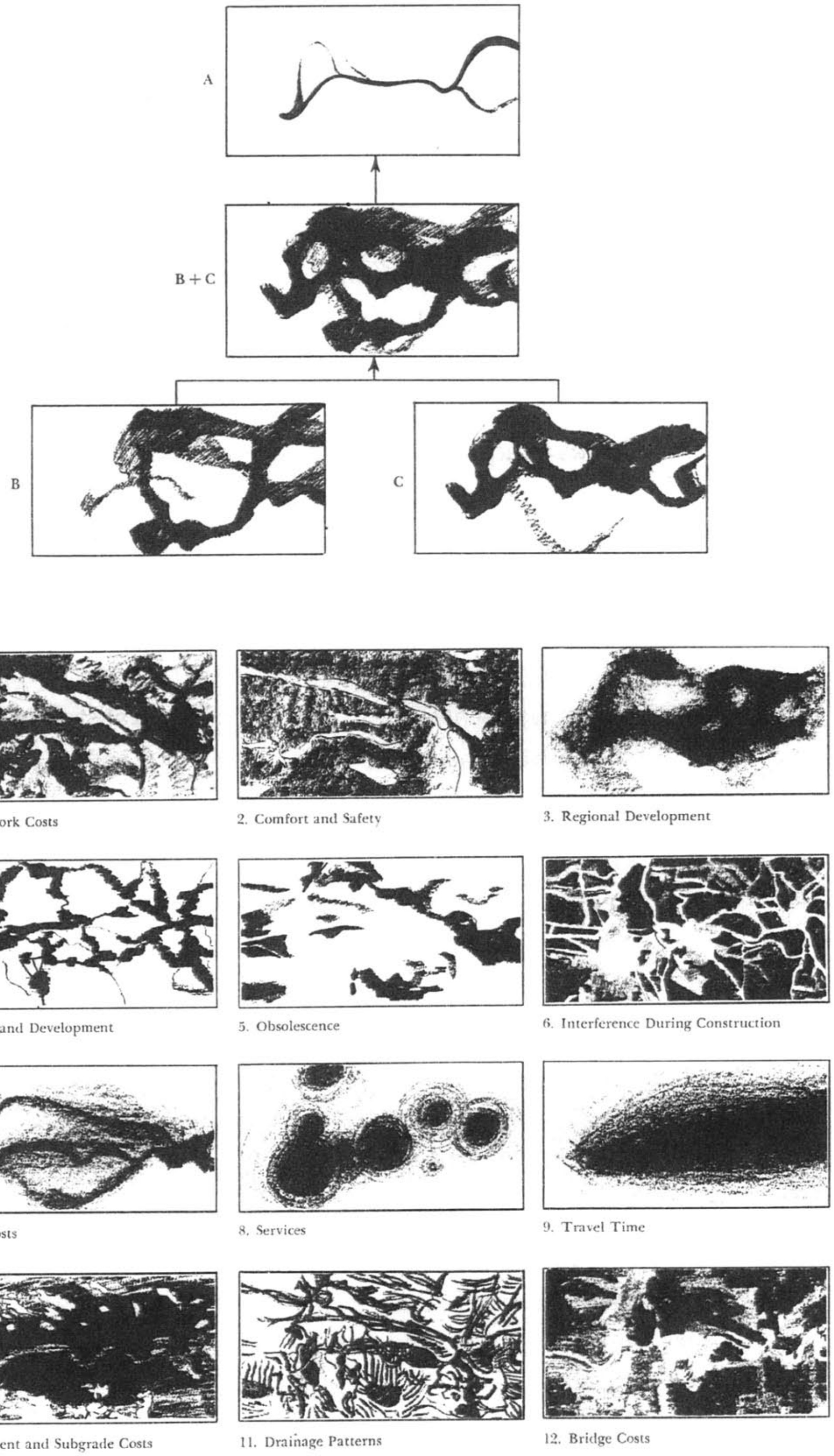


Figure 5.1. The relational method illustrated by Alexander.
(From Alexander Christopher. "From A Set of Forces to a Form.")

the complexity of the context.

The numerical method relies upon representing each force by "the variation of a one-dimensional numerical variable. One of these seeks minimization (or maximization). The others are held constant, and are called constraints."[11] This method is very limited because it works only within a representation of a single one-dimensional numerical variable, and most of the more subtle human forces (practical, psychological, and social) cannot be represented by this manner. The other two analog methods generate the form by physically simulating these forces and allowing the forces to act upon the form, such as Gaudi's string method. Alexander presented the relational method as a promising strategy to deal with uncalculated forces. The relational method relies on finding a common ground wherein all the forces can interact. The common thing about them is the tendency to seek out some specific kind of end-state through their physical potencies. The process is to abstractly define the physical relationship between the forces; then by fusing and by superimposition, the implications of these forces emerge.

The example he cites is very compelling. The goal of the project was to locate a twenty-mile stretch of highway in Massachusetts. He began by plotting 26 forces and their physical relational implications. By simply superimposing these diagrams, he obtained the sensuous shape of the highway. Although he admitted that this was a very simple example, the process of fusion was achieved simplistically through the superimposition of the diagrams. Previously he mentioned the difficulty of trying to assess the forces of a certain context. Even this last method is full of ambiguities that he himself does not foresee how to deal with it. The translation of the diagram into three dimensional constructs was done crudely. By moving to a further state of abstraction, the project was left in the form of abstract diagrams.

5.1.2. John Frazer

> Our model is, at any given time, the expression of an equilibrium between the endogenous development of the architectural concept and the exogenous influences exerted by the environment.[12]

Frazer, in his book "Evolutionary Architecture," posits very similar issues that had concerned Alexander for a long time. The first is achieving the metabolic balance, i.e., fitness, between the generated form and its environment. The second is seeking out another abstract environment rather than the physical one, which is very costly to compress the morphological evolution occurring therein. While Alexander's idea about this abstract space was vague, Frazer conducts his experiments in a computer virtual space. "'Imaginative use' in our case means using the computer – like the genie in the bottle – to compress evolutionary space and time so that complexity and emergent architectural form are able to develop."[13] The third is the lack of trust in traditional intuitive methods employed by architects; these methods, Frazer believes, are not enough to face the increasing complexity of design problems. The notion of 'misfit' is also used to refer to the conflicts between the increasing complexity of contextual forces in a 'self-conscious' culture.[14] Frazer's aim is not to generate novel forms so much as to generate an evolutionary architecture that exhibits metabolism.

His work is premised on the fact that architecture is a living, evolving thing that is subject to "principles of morphogenesis, genetic coding, replication, and selection."[15] Frazer goes so far as to call his architecture as being conscious of its environment.[16] The aim is to compress evolution in a simulated environment by creating virtual architectural models that evolve in harmony with the natural forces. The computer is used as an evolutionary accelerator. In his words: "[o]ur present search is to go beyond the 'blueprint' in architecture and to formulate a coded set of responsive instructions (what we call a 'genetic language of architecture') that may yield a more appropriate metaphor."[17]

He acknowledges the fact that natural evolution occurs without pre-knowledge of what is to come. He claims that harnessing some of these natural processes can bring improvements to the built environment. He looks at nature because of its perfection and variety brought about by the relentless experimentation of evolution.

Computer modeling was very crucial to achieve this process. It allows the forms to evolve in a virtual environment without the expenses that accompany physical construction. Another important point was environmental modeling. The environment has to be explicitly defined so that it interacts with the form. In trying to simulate environmental factors, Frazer constructed antennae that worked as transmitters or receivers of information and detected movement, sound, and color. They also included video systems for detecting cultural patterns.[18] However, all of these mapping procedures are not adequate to capture the complexity of a design problem.

Genetic Algorithms (GA) are used to steer the evolution process through selecting the fittest variant out of a huge number of mutations. As is well-known, Genetic Algorithms were developed for scientific problems that require search and optimization, but problems in architecture cannot be subject to this optimization. In science, the selection criteria can be codified, while in architecture, it is very difficult to codify the contextual parameters. The solution would be to use artificial selection achieved by the user's intuition after every generation. However, this will slow down the process of morphogenesis, and the myth of compressing evolution will be beyond reach.

As in shape grammars, the design activity is shifted towards designing the genetic code-script, rules for the development, a simulated environment, and most importantly, the selection criteria.[19] The concept for Frazer is implied in the way we design the generative rules.

What is interesting about Frazer's model is that evolution is process-driven rather than form-driven. What is modified during evolution is the genotype rather than the phenotype. The phenotype is the outward expression of the genotype. This is why selection plays a paramount role in evolution. "They are plans for action, not plans of form, whose formal destinies are entwined with the material in which they are manifested and only revealed in their interaction with unique sets of contingencies in the material world."[20] It is interesting to think of the body of a human as being an interface between the environment and the internal genetic code, the DNA. "[T]he genetic material inherited by an organism can be said to represent an implicit model of the environment that organism's parents were subject to, proffered as a 'best guess' at the world the offspring will be entering. Each generation of the organism's species is 'edited' by natural selection to produce a revised set of genes specifically suited to the current environment of the species."[21]

At the end of his book, Frazer goes so far as to propose evolving "architectural life from nothing, with no preconceptions, with no design…just blind tactics."[22]

The proof of the impossibility of mapping the context is mentioned in his book itself in the postscript written by Tim Jachna. Jachna states: "[t]he context into which a designed object is introduced consists of microstates of a virtual infinity of unsimulatable systems, each of which is unpredictable in behaviour and affected by the equally unpredictable behaviour of the others. Design may be seen as a process, but the moment of the intrusion of its product into the world is one of the ill-prepared abruptness and violence."[23]

5.1.3. Emergent Design Group[24]

> The development of an organism...may be considered as the execution of a 'developmental program' present in the fertilized egg. ...A central task of developmental biology is to discover the underlying algorithm from the course of development.
>
> - Artistid Lindenmayer and Grzegorz Rozenberg[25]

The Emergent Design Group was co-founded at MIT in 1997 by Peter Testa, Devyn Weiser, and Una-May Oreilly. This group aimed at bridging the different disciplines ofarchitecture, artificial intelligence, computational geometry, engineering, and material science. Their work comprises different software tools. I am going to talk specifically about GENR8 which stands as the last software that was developed by this group. Although their work lacks the theoretical rigor found in Frazer's theory, they were able to develop complicated tools.

GENR8 (Generative Form Modeling and Manufacturing) is a surface modeling tool that simulates organic growth of surfaces in a given environment. It is based on the natural biological growth processes to create novel surfaces. The concept was to combine the advances in generative design with the combinatorial aspects of Alife with an evolutionary search component. This project was modeled on complexity theory and simulated surfaces to grow in a bottom-up fashion, combining primitive elements in a non-linear way. The tool was developed without having a specific function in mind. It is intended to be a general tool that could have many different applications that are not limited to architecture. It is a growth system assisted by evolutionary techniques. This tool was incorporated as a plug-in inside Alias | Wavefront Maya.

Map L-system, which is an extension of L-system, was adopted as a growth model. L-system is a mathematical description of plan growth. L-systems are very similar to shape grammars; however, they are represented textually rather than spatially, and that explains the difficulty of designing rules with L-systems. They simulate growth through recursive application of these rules. In an L-system, the grammars are either context-free or context-sensitive. The context-sensitive rules are in the form:

$$B < A > C \rightarrow X.$$

The letter 'A can produce the letter X if and only if A is surrounded by B on the left and C on the right. The permutation produced by map L-system is huge to explore. GENR8 uses evolutionary computation to explore and select from this vast space.

This simulation of the **environment** significantly changes the growth of the system. As mentioned before, biological models mandated an explicit definition of the environment. In GENR8, the environment was modeled through three types of forces: attractors, repellors, and gravity. They affect directly the growth of the surface. They are defined as points in the space. Their positions and magnitude can be adjusted by the user, except gravity which is defined as a global force. These forces were defined mathematically following the Newtonian definition:

$$f = c * d^{-e}/m$$

where c is the constant; e is the exponent; d is distance; mis mass of the surface; and gravity as a uniform force not expressed in this equation.

These contextual forces were made to influence the grammars of the growth system. Other endogenous factors were added to the system – repelling points, fixed perimeter, fixed center, random noise. Noise affects the position of the points and the other factors affect the type of segments.

The evolutionary component of the process was achieved by employing an Evolutionary Algorithm (EA) called Grammatical Evolution (GE). A key issue in EA is the

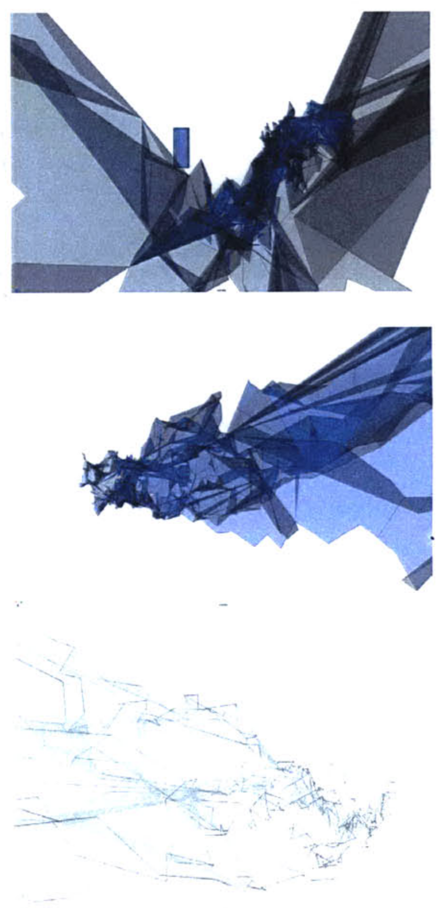


Figure 5.2. Surfaces generated by GENR8.
(From <http://web.mit.edu/edgsrc/www/>)

fitness criteria. These criteria have to be defined explicitly by using mathematical functions; or they can be defined interactively by the user at each population. However, the process becomes time-consuming for the user and limits the population size and the number of generations in practical use. Five different criteria were implemented: size, smoothness, soft boundaries, subdivisions, and symmetry. The user can manually set these parameters. To allow more control for the user, they introduced what they called the idea of interruption, intervention, and resumption (IIR) that allows the user to interfere in the process at any time and modify the direction of the process.

5.2. Bio-alogical

5.2.1. Greg Lynn

Greg Lynn, one of the most renowned contemporary architects, has been researching the potential use of computers in the design process since the early 90's. Lynn is one of the very few digital architects to follow a rigorous process which he always tries to externalize. *Alogical* is used here to denote the confusion that arises in Lynn's theory. It is not clear whether he is talking about a logical or an illogical approach to architecture. The example he cites about the shape of the boat hull is very provocative within its context. This example gives the impression that he is talking about a design problem that can be rationally solved by mathematical algorithms. He mentioned the example to show how, in other domains, the shape can register and internalize the forces of the site. The confusion stems from the fact that the forces he uses throughout his project are simplistically mapped out from the site. So, is he talking about a literal or metaphorical usage of the notion of force?

In his book, "Animate Form,"[26] Grey Lynn continues to examine the project launched by Alexander for conceiving the context as an active abstract space that orchestrates the form-making process. "This shift from a passive space of static coordinates to an active space of interactions implies a move from autonomous purity to contextual specificity."[27] Lynn's argument is based on the assumption that architecture, unlike other fields, has been anchored in a neutral space of Cartesian coordinates.

His argument does not diverge so much from Alexander's. However, there is an emphasis that the form registers and stores these forces leaving their traces embedded in the resulting form. The form becomes a mnemonic structure from which the forces that have inflected it can be deduced.

Lynn is not arguing for a neo-rationalist approach in architecture. He wants to legitimize his **aesthetic quest** by resorting to pseudo-scientificity. He is seeking a dynamically conceived architecture that implies animation and form evolution.

Unlike in the biological model where the emergent form stabilizes the contextual forces, terminating the process is not a key issue in Lynn's project. Lynn shows the stages of morphogenesis but does not show how the ending can be come about. Lynn vindicates an animated process in which

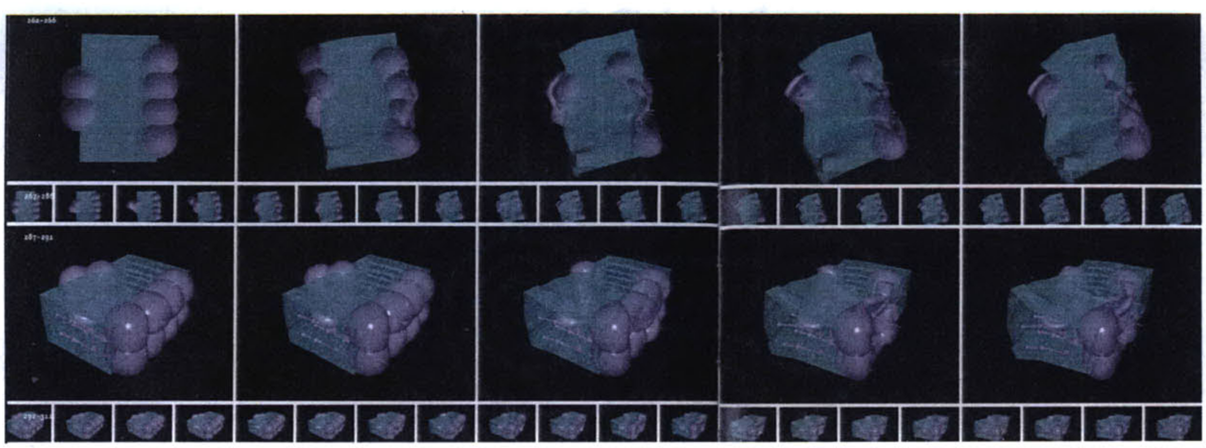


Figure 5.2. Greg Lynn's House Prototype Project.
(From Lynn, Greg. *Animate Form*).

not only the form is animate but also the forces.

Unlike in the previous biological processes, Lynn talks about the adaptability of the generated form according its context as a formal-driven process rather than a process-driven one. He is more concerned about how the form is going to respond **formally** to the complexity of a given context. He argues against the purist functional approach pioneered by the Modern architects. The form should be more sensitive and plastic to the contextual forces, rather than oppressing them by designing a pure form. He argues that deformation, inflection, and curvature are three ways of registering force on form. "Because topological entities are based on vectors, they are capable of systematically incorporating time and motion into their shape as inflection. Inflection or continuous curvature is the graphical and mathematical model for the imbrication of multiple forces in time."[28] In this topological definition of even the simplest forms such as a sphere, complexity is always present as potential.

Although he preaches a better understanding of "the appearance of these tools in a more sophisticated way than simply a new set of shapes,"[29] he is the one who understands the new digital tools as a new set of shapes.

His theory and process is to legitimize an aesthetic quest that is based on curves of calculus equations.

> The computer also introduces a new aesthetic vocabulary of curves based on calculus equations. Although we have been using calculus to analyze form in the design process we have never had a tool that allows us to intuitively design using the logical curves of calculus equations.[30]

Notes

The abstract space that Alexander was looking for to conduct his testing can be found in different software environments.

Lynn's work was, to a large extent, mandated by the software he is using: Maya Alias | Wavefront. This software was designed specifically to enhance visual effects in the movie industry. It was not intended to simulate complex phenomena; however, it simulates different phenomena like collisions of rigid or soft bodies. Different natural forces such as wind, gravity, and fluids were modeled based on the laws of physics. These forces create lifelike conditions. Dynamics and particles were used to enhance this realistic effect.

As illustrated before, Alexander was very keen on elucidating his usage of the notion of force, in his effort to try to devise an objective definition of the notion of need that can encompass all the contextual parameters. However, Lynn does not offer any explanation of why he is using a Newtonian definition of force to express complex contextual parameters. He assumes that all these forces are quantifiable. For instance, in his project, "House Prototype in Long Island," he maps these forces according to their visual attributes by giving them attracting or repelling values that are tempered with various parameters for decay, acceleration, and turbulence. The foundations of the existing building were mapped using a vortex force; the oak tree and the neighboring houses were mapped using repelling radial forces. These forces interacted with one another to produce a "gradient field of attraction and repulsion across the site."[31] Particles, then, were deployed in this gradient field to make these forces visible. These particles were mapped on spline elements. The function of the house was introduced only through the initial pre-deformed shape, "H" plan.

Clearly connected to the biotechnical approach, Lynn also designed the Embryological Houses. He described it as "a strategy for the invention of domestic space that engages contemporary issues of brand identity and variation, customization and continuity, flexible manufacturing and assembly, and most importantly an unapologetic investment in the contemporary beauty and voluptuous aesthetics of undulating surfaces rendered vividly in iridescent and opalescent colors."[32]

Many of the variations in any Embryological House are based on an adaptation to contingencies of lifestyle, site, climate, construction methods, materials, spatial effects, functional needs and special aesthetic effects. Their plasticity denotes their ability to be shaped in response to any given context and to accommodate multiple functional and material requirements.

(Endnotes)

[1] Alexander, Christopher. 1964. *Notes on the Synthesis of Form*. Cambridge, Harvard University Press, p.77.

[2] Ibid, p.15.

[3] Steadman, Philip. 1979. *The Evolution of Designs: biological analogy in architecture and the applied arts.* Cambridge; New York: Cambridge University Press.

[4] Alexander, p. 36.

[5] Ashby, William Ross. 1966. *Design for a Brain: The Origin of Adaptive Behavior.* London: Science Paperbacks.

[6] Steadman, Philip. *The Evolution of Designs: biological analogy in architecture and the applied arts*, p.174.

[7] Alexander, Christopher. *Notes on the Synthesis of Form*, p. 76.

[8] Ibid., p.8.

[9] Alexander, Christopher. 1966. "From a Set of Forces to a Form." In Kepes, Gyorgy. 1966. *The Man-Made Object*. London, Studio Vista.

[10] Ibid., p. 96.

[11] Ibid., p. 98.

[12] Frazer, John. 1995. *An Evolutionary Architecture*. London: Architectural Association, p.103.

[13] Ibid., p.18.

[14] Ibid., p.106.

[15] Gordon. "Introduction". *In* Frazer, John. *An Evolutionary Architecture*.

[16] Frazer, John. *An Evolutionary Architecture*, p. 103.

[17] Ibid., p.11.

[18] Ibid., p.75.

[19] Ibid., 65.

[20] Jachna, Tim. "Postscript." *In* Frazer, John. *An Evolutionary Architecture*, p.113.

[21] Ibid., p.116.

[22] Ibid., p.101.

[23] Ibid., p.109.

[24] More information can be found at: http://web.mit.edu/edgsrc/www/.

[25] Quoted from Flake, Gary William. 1998. *The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaptation*. Cambridge, MA: MIT Press, p.77.

[26] Lynn, Greg. 1999. *Animate Form*. New York: Princeton Architectural Press.

[27] Ibid., p.11.

[28] Ibid., p.23.

[29] Ibid., p.17.

[30] Lynn, Greg. Lynn and Hani Rashid Architectural Laboratories.

[31] Lynn, Greg. *Animate Form*, p.143.

[32] Lynn, Greg. *In* Greg Lynn and Hani Rashid Architectural Laboratories, p.84.

Chapter 6: Speculations

Two mains issues can be extrapolated from the two models that I traced in my previous two chapters: first, a biological fallacy stemming from the biological model, and second, the inadequacy of the rules stemming from the linguistic model.

6.1. Biological: The Biological Fallacy

In the biological model, the form is the result of a logical process, or what is referred to as a biotechnical determinism. "The aesthetic of architectural form ... was achieved without the conscious interference of the designer but as something which nonetheless was postulated as his ultimate purpose."[1] These processes preached pure technology and objective design methods at the expense of the iconic significance of forms. There is no intentionality in the making of form. Beyond this biological determinism, the problem of the biological processes is twofold: first, these processes equate cultural evolution with organic evolution; and second, these processes presume that the context is reducible to codified statements.

6.1.1. Darwinian vis-à-vis Lamarckian Theory of Evolution

Steadman, in his book, "The Evolution of Designs,"[2] talks about the biological fallacy in architecture. According to him, this fallacy stems from the fact that cultural evolution can be explained through organic evolution. The difference between organic evolution and cultural evolution can be elucidated through the difference between the Darwinian and Lamarckian theories of evolution.

According to Darwinian and neo-Darwinian theory, variations and permutations happen randomly without any direction. Evolution can only be explained by natural selection, which plays a paramount role. The permutations that are selected transmit these newly acquired characteristics to their offspring. Evolution is without a plan. It is an 'elective' theory of evolution where the environment controls the evolutionary process through selection. [3]

On the other hand, Lamarck's theory implies that variations try to direct evolution to become better adapted to a changing environment. This theory implies a teleological character in the evolution process. It is an 'instructive' theory of evolution. The environment itself teaches the organisms to better adapt. "Language, unique to human species, provides the channel by which the accumulated experience of each generation in coping with the problems of life can be passed on to the next".[4] Through culture, human beings are able to transmit this accumulated experience.

Architectural form, which is a cultural artifact, cannot be subjected to the Darwinian definition of evolution. Forms evolve in a complex teleological fashion in which many parameters, including cultural, environmental, and social, interact with each other to direct this evolution.

In projects done by Frazer and the Emergent Design Group, Genetic Algorithms were used to select the fittest variant out of a huge number of permutations based on very clear selection criteria. Genetic Algorithms are modeled after the Darwinian model of evolution, in which only natural selection determines the direction of evolution. GA can be useful in other domains for optimization where the selection criteria are objective and amenable to absolute definition.

One of the main consequences of adopting the Darwinian model is neutralizing and marginalizing the individual designer. In Steadman's words: "Just as Darwin inverted the argument from design, and 'stole away' God as designer, to replace Him with natural selection, so the Darwinian analogy in technical evolution removes the human designer and replaces him with the 'selective forces' in the 'functional environment' of the designed object."[5]

6.1.2. Reductionism

All the processes in the biological model exhibit a great deal of reductionism. Reductionism is manifested in rendering two entities – the design problem and the designer's cognitive process – reducible to explicit encodable instructions, so that they are amenable to computational algorithms. The computationalist approach requires reductionism to function. To sum up these reductions, I am going to use Liddament's argument in which he notes three different levels of reductionism: ontological, epistemological, and methodological.

Liddament, in his article "The Computationalist Paradigm in Design Research,"[6] shows that although computational tools are powerful in scientific domains to solve many problems, they do not adequately fit the actual design activity. He acknowledges the fact that the computationalist paradigm presents itself as a "scientific approach with a correspondingly rigorous methodology."[7]

Ontological Reduction

Ontological reduction is encountered when the complexity of a given design problem is systematically reduced into mere physical forces that have attracting and repelling attributes. The success of idealizing objects in scientific applications does not mean that it can work for all domains, especially in design. This reduction involves eliminating the "messy and often ill-defined entities of the 'real' world to theory-based 'idealizations' and theoretical entities."[8] Architects are usually concerned with the particularities and the specific requirements rather than with the generalization of any design problem. These complex particularities are not easily reducible to theoretical entities as can be done to a greater extent in science. Design problems defy generalizations and reductionism. Many authors have referred to design problems as 'wicked'[9], ill-defined, and indeterminate, defying systemization and scientific methods.

Irreducible complex design problems are not amenable to systemization strategies via computational algorithms. Complexity theory offers a more accurate description of dynamic real-life phenomena than the traditional Newtonian cause-and-effect worldview. Each of these systems is at its simplest description and cannot be reduced to general laws that govern its behavior. Non-linear systems defy reduced description. These unsimulatable nonlinear systems are in their briefest description, and their descriptions cannot be compressed. This shows that even those dynamic systems which used to be simulatable no longer possess this quality. What should be deduced from complexity theory is the impossibility of reducing the contextual parameters into ideal fields of forces. Complex systems are constituted of many independent and varied elements that interact in intricate organizational configurations. Because of their non-linearity and sensitive reciprocity with the environment, complex systems are hard to predict in the long run. They are also difficult to isolate, model, or reproduce for experimental purposes. We need design processes that can cope with such complexity and not design processes that model complexity theory.

Only very few design problems have defined problem 'domains' that can be solved by the successful applications of certain algorithms. Most design problems are also intermixed with cultural and social factors that cannot be systemized and subjected under the control of an algorithm.

6.2. Linguistic: The Inadequacy of the Rules

The linguistic model seems more promising, especially considering that the emphasis is directed towards designing the rules rather than falsely aiming for achieving fitness between the form and its environment. The research that has already been done with shape grammars is seminal in this regard, offering more malleable grammars that mediate between a complex designer who aspires for control and the unanticipated results produced by grammars. However, much needs to be done to create rules that will open up more formal possibilities while simultaneously tempering the nuances of a given context -- rules that will liberate us rather than trap us.

6.2.1. Rules or No Rules?

> The admirable structures found in nature – crystals, atomic systems, flowers – reveal configurations of forces that are arrested at some level by the constraints of their theme and left sufficiently alone to realize their form perfectly.[10]

Do we need explicit rules? Does using rules indicate rationality, order, and determinism, and does the lack of rules in turn indicate irrationality, disorder, and nonlinearity? Can irrationality be produced through rules?

Once, while working on a project, I became utterly lost and disoriented because I had so many options and directions to pursue – a vast space that I had to dive into and navigate through. It was a state of amorphism. Constraining this space before I dove in seemed a very pleasant idea. The process of adding constraints meant subjecting this space under rules that will exclude a huge part of the space and leave a portion that complies with the newly established rules.

Claude Lévi-Strauss has already referred to this process of ordering and structuring in his theory of language formation. Mario Gandelsonas explicated Lévi-Strauss's theory in this manner:

> The establishment of society can be seen as the

> establishment of order through conventions, or more specifically, the establishment of a language through symbolic codes. Before order, before language, there exists a primal chaos where there are no rules for marrying, building, eating; in this chaos, which precedes society, there is only an infinite field of potential for manipulation of the individual and collective realms from the verbal to the sexual. The systematization and institutionalization of rules in these domains, the making of rules, involves at once a repression of chaos, of the amorphous, and an invention of social codes of a 'language' of kinship relations, a 'language' of myth, or a 'language' which expresses the spatial organization of a tribe.[11]

Two things can be extracted from this: first, stepping into a higher order is inevitable; second, that I had made a sacrifice by constraining my space and negating a huge part of it. According to that theory, stepping up into a higher order involves sacrifices. Sacrifice involves the establishment of rules, of an order.

My confusion was increased after I read Sanford Kwinter's article titled "The Computational Fallacy,"[12] in which he touches upon the same issue that the shift to a higher mechanical order implies suppressing the embedded intelligence found in natural materials. "The movement of all (advanced) technological societies has been one from archaic matter intelligence (empirical, qualitative, multi-spectral) to mechanical matter intelligence (numerical, dissociated)," he writes, "but only incompletely and each in its own way." All natural materials, for him, possess an embedded material intelligence that is being suppressed by mechanical complexes that are unifunctional and deterministic, while fundamental, free, and unprocessed matter has magical qualities of material intelligence.

After my own attempts at exploring these constrained spaces created in the wake of new rules, I found that I am over-constrained by my rules; therefore, I wanted to break out free from them. While I was trying hard to break from these fixed orders, I read Alexander's "Notes on the Synthesis of Form," in which he wrote that when one steps up into a higher order he cannot break out of it because he cannot see the world without these new orders. He calls this phenomenon 'loss of innocence'.[13] This meant that, for me, I will automatically evaluate anything I now create based on the newly acquired rules, and I will not accept anything without this process. I am in a higher order now and I cannot step down. The only thing I can do is to improve my rules.

However, architecture, at some point, becomes too crapulent with rules to the extent that it can not move. Only then will it necessarily change back into an amorphous condition.

Greg Lynn proposed an alternative view. Lynn, in his theory and praxis, seeks to change and break out of these fixed orders and go back to an amorphous condition wherein local differences can be addressed -- a condition that is more pliant that can internalize the forces of a given context. Establishing fixed orders is always achieved at the expense of repressing local differences of program, structure, form and culture.[14] He preached for vague 'anexact' forms that are neither exact nor inexact. Unlike eidetic ideal forms, these forms are pliable enough to address particularities and differences.

6.2.2. Rules: Do They Constrain or Liberate?

Lionel March cleverly coined it as a title for one of his articles: "Rulebound Unruliness." For him, using rules does not constrain the designer but in fact liberate him:

> In a shape grammar, a rule is no fetter – as school children might view school rules – but, on the contrary, shape rules liberate. They

> provide the language in which the designer speaks. They give the designer 'style.' Freedom comes from following the rule.[15]

Within a broad class of spatial systems, shape grammars offer the chance to build both rule-bound and sometimes unruly systems. For March, every conceivable shape can be generated through geometrical constraints.

Jackson Pollock's paintings present an ideal example in this context. Beneath his seemingly arbitrary paintings lies an intricate order, which explains the appeal of his paintings. Pollack is very renown for his drip painting technique. He is an example of someone who implicitly used rules to generate unruly systems, as illuminated by Richard Taylor, a physicist, who has dedicated much time into analyzing Pollack's paintings. He used fractal analysis and image processing techniques and discovered that the formulated patterns on Pollack's drip paintings are in fact fractals with a size range, the largest pattern being more than 1000 times larger than the smallest.[16]

It is a matter of finding the right plateau between the two extremes: rigidity and plasticity of rules, rigidity that excludes contingencies and plasticity that permits these contingencies to be internalized without filtering. Can we establish rules without making any sacrifices of meaning or function or the humanistic touch?

6.3. Digital Architecture vis-à-vis Generative Architecture

> What appears on the surface as a hard, rational discipline of design turns out rather paradoxically to be a mystical belief in the intuition process.[17]

To follow up on our previous discussion regarding Greg Lynn's bio-alogical form-generation processes, most contemporary, the so-called digital, architects adopt the "alogical" part and leave out the "bio." These architects aspire to be logicians, but at the same time their processes are completely illogical. Following the mathematical logic of the computer, most contemporary architects tend to rationalize their praxis either to legitimize their work, or to set themselves against the previous generation who praised intuition, individualism, and creativity as the main impulse behind architecture. Some metaphors are being used to conceal the intuitive methods with a rationalizing mask: "abstract machine," generative diagram, and mathematics. Because of this, words like 'digital' and 'generative' have become synonymous with each other, as if one cannot be mentioned without implicitly associating with the other. However, this is far from the truth. These architects employ an algorithmic (procedural) process that is being implemented non-algorithmically. In other words, they use procedural logic that is not rule-based.

There is still, among these architects, a reluctance to internalize the logic of the computer, to comprehend the order imposed by the computer. Architects are still shifting between the old condition – the architect as an artist – and the new condition, the architect as a scientist or a mathematician. The computer is urging everyone to imitate the scientific methodologies to resolve design problems. Architects in the early 50's and 60's who wanted to embrace computer technology were trying hard to subject traditional design methods to systemization and mathematics. Now, it seems that architects want to subject computers to their intuitive logic.

Computer software packages offer a very malleable environment within which to easily carve out complex calculus-based forms. Most approaches are trial-and-error based and do not follow a clear rationale. Dynamism and free manipulations of form gives the impression of continuous transformations. This malleability and continuous transformations of form give the impression of a generative process.

What is at stake now is a form-driven architecture rather than a process-driven one. These architects are seeking formal solutions to complex design problems. Rather than being defined as the properties that remain unchanged when the object undergoes transformations and deformation, topology is understood in its formalistic sense.

Many contemporary digital architecture praxes such as Ocean North and UN Studio have been engaging in generative diagrams as a method to mediate between design problems and new computer software. Using generative diagrams indicates a yearning towards objectivity and science, described by Picon as "a new realism."[18] Diagrams come to fill the vacuum and to conceal the traditional intuitive process by a shell of rationalism and realism. It is a pseudo-scientific approach that tries to validate the necessity of architecture by generating architecture through deep and thorough analysis.

For these architects, the generative diagram is not reductive but has the potential to embrace the complexity of a given design problem, and to engirdle the project within a layer of reality. These diagrams are dynamic systems that emphasize structural organization and relationships at the expense of typology and semantics..

Generative diagramming contributes to the production of complex geometries derived from site information and deformed by site-specific forces or introduced information. These geometries, when negotiated towards real-life situations, produce spaces which are adaptable, flexible, and programmable, yet articulated and rich.

These generative diagrams are usually accomplished through employing animation techniques such as particles to explore new design possibilities. Using particles as a contextual mapping process gives way to depicting the continuum of data across the site. They register the flow of the context's forces, rather than indexing discrete nodes. Advanced software packages like 3D Max and Maya are used as engines to simulate the environment by he use of particles and forces. Translating this information into architecture remains a mysterious quest. Generative diagrams do not proffer a formal mechanism to transfer this mapping process into form, and thus the diagram, more often than not, undergoes a crude translation into form .Despite the claim that a generative diagram is not a reductionistic reading of the context, this method of mapping is very subjective and does not reflect the real forces that exert their influence upon the form. Despite the claim that the role of the subject is reduced to pre- and post-production phases within this process, the subject still dominates the process as in rule-based systems,.

An interesting issue to note here is the authoritative position held by the artifacts produced using seemingly scientific methods such as generative diagrams, which attributes iconic power to the creations of technology.

One of the main repercussions of using digital tools is what Picon refers to as "the destabilization of form."[19] The free-form manipulations allowed within virtual spaces created by computers lends itself to the belief that the resultant form is a product of an arbitrary stop, or a snapshot, in a continuous flow of transformations. What, however, is the guarantee that the resultant form is the best variant we can hope to attain? It is like writing a script, a continuum of possibilities, and the resultant form is a singularity within this flow.

6.4. Scientific Methods: Inductive Fallacy

As many contemporary architects seek to emulate scientific methods in their design processes, it seems that they fall into the same misconception within the philosophy of science illustrated by Karl Popper as 'inductive fallacy.' Scientific works are known to be authoritative, objective, and original. These architects depend on that model of induction to formulate their theories. They emphasize on thorough analysis through which form should emerge, negating by this all preconceptions a designer might have. In this manner, they render the subject as a passive observer.

Does science bring with it the death of the scientist? Science transforms the nature of the scientist, from being mystical to being a reasoning figure. Science does not diminish the role of the individual. On the contrary, huge emphasis is placed on the scientist's creativity. The notion of the scientist being a passive observer can be a testament to the induction fallacy clarified by Karl Popper.[20]

The inductive view suggests that through patient observation a pattern or a law will emerge that will impose itself on the scientific observer. This is not what usually happens. According to Popper, the scientist is a part of the experiment, and he imposes his own hypothetical explanations on the phenomenon in question. In every cycle, the scientist tries to direct the experiment towards his hypothesis's demands.

This inductive conception of scientific methodology formulated the basis upon which Alexander had proposed his systematic design method. Producing a hypothesis out of a given body of data is similar to producing an architectural form out of specific requirements.

This approach became typical of the work borne by the design methods movement. "First 'data' were collected and assembled into the 'programme'; meanwhile all premature urges to define the form and shape of the building were suppressed."[21] It was only through analysis that the form would emerge. All premature preconceptions should be purged in favor of thorough analysis. As Hillier and Leaman noted, " '[r]ationality' in design was virtually equated with purging the mind of preconceptions, to make way for a problem solving method which linked a procedure to a field of information."[22]

As Steadman suggests, it is important to demarcate the boundary between what is amenable to scientific treatment and what is in the realm of cultural, aesthetic and moral factors and values.

(Endnotes)

[1] Colquhoun, Alan. 1981. *Essays in Architectural Criticism: Modern* Architecture *and Historical Change.* Cambridge, MA.: MIT Press, p.45.

[2] Steadman, Philip. 1979. *The Evolution of Designs: Biological Analogy in Architecture and the Applied Arts.* Cambridge; New York: Cambridge University Press.

[3] Ibid., p.187.

[4] Ibid., p.187.

[5] Ibid., p.189.

[6] T. Liddament. 1999. "The Computationalist Paradigm in Design Research." *Design Studies,* Vol 20, No 1.

[7] Ibid., p.44.

[8] Ibid., p.45.

[9] Buchanan, R, "Wicked Problems in Design Thinking," In Margolin, V and Buchanan, R (eds.). 1996. *The Idea of Design.* Cambridge MA: MIT Press.

[10] Arnheim, Rudolf. 1977. *The Dynamics of Architectural Form: Based on the 1975 Mary Duke Biddle Lectures at the Cooper Union.* Berkeley: University of California Press, p.165.

[11] Gandelsonas, Mario. "From Structure to Subject: The Formation of an Architectural Language". In Eisenman, Peter. 1982. *House X.* New York, NY: Rizzoli, p.7.

[12] Kwinter, Sanford. 2002. "The Computational Fallacy." *Thresholds* Vol. 26, p. 90.

[13] Alexander, Christopher. 1964. *Notes on the Synthesis of Form.* Cambridge, MA: Harvard University Press.

[14] Lynn, Greg. 1998. *Folds, Bodies & Blobs: Collected Essays.* La Lettre Volee, p.48.

[15] March, Lionel. 1996. "Rulebound unruliness." *Environment and Planning* B, 23, p.398.

[16] Richard Taylor. 1999. "Fractal Analysis of Pollock's Drip Paintings." *Nature*, Vol 399, p.422.

[17] Colquhoun, Alan. 1981. *Essays In Architectural Criticism: Modern Architecture and Historical Change.* Cambridge, MA: MIT Press, p.46.

[18] Picon, Antoine. "Architecture, Science, Technology, and the Virtual Realm". *In* Picon, Antoine and Ponte, Alessandra (ed.); Foreword by Ralph Lerner. 2003. *Architecture and the Sciences: Exchanging Metaphors.* New York, NY: Princeton Architectural Press.

[19] Picon, p.303.

[20] Popper, Karl Raimund. 1972. *Objective Knowledge: An Evolutionary Approach.* Oxford, Clarendon Press.

[21] Steadman, Philip. *The Evolution of Designs: Biological Analogy in*

Architecture and the Applied Arts, p.206.

[22] Hillier Bill, quoted in Steadman, Philip. *The Evolution of Designs: Biological Analogy in Architecture and the Applied Arts*, p.207.

Chapter 7: The End of the Beginning

From my explorations into the nature of generative design, I came to a conclusion that there is no way I can build a machine without my operative presence. Hence, I have to design a machine that will allow an intimate relationship between me and itself. After I design the machine, I have to relinquish some of my responsibilities to it. By so doing, I will get the machine under my control but simultaneously allow it to behave in an unpredictable fashion. "By negotiating the degree of discipline and wildness, one can cultivate an intuition into the behavior of computer-aided design systems and the mathematics behind them."[1] Only then will the machine extend myself and expand my cognitive ability. The issue then boils down to how to design good machines. To design good machines, we have to understand how to work algorithmically:

- Can I find rules that will liberate me without sacrificing function, meaning, and the humanistic touch? Or should I accept the sacrifice when working under the hegemony of rules?
- Can these rules address the contextual parameters of a given design problem without a loss?

The problem that pertained to my case studies and other generative systems I have studied here is the lack of a deep conceptual understanding of rules. Working with rules implies a completely different way of thinking; this friction cannot be accommodated unless we reach a point where we control the rule-making process. At that point, the rules will cease to be a trap and allow us to extend our cognitive and spatial capabilities.

These rules, if achieved, would be capable of bridging the gap between the designer and the computer. A computer will be able to deploy these rules quickly and precisely. What Alexander says in this regard is seminal:

> Anybody who asks, how can we apply the computer to architecture? He is dangerous, naïve, and foolish. He is foolish, because only a foolish person wants to use a tool before he has a reason for needing it. He is naïve, because as the thousand clerks have shown us, there is really very little that a computer can do, if we do not first enlarge our conceptual understanding of from and function; and he is dangerous, because his preoccupation may actually prevent us from reaching that conceptual understanding, and from seeing problems as they really are.[2]

Looking for these rules is my next step.

(Endnotes)

[1] Lynn, Greg. *Animate Form*, p.20.

[2] Alexander, Christopher. 1967. "The Question of Computers in Design". *Landscape*, pp. 8-12.

Bibliography

- Alexander Christopher. 1964. *Notes on the Synthesis of Form*. Cambridge, Harvard University Press.

- Alexander Christopher. "From A Set of Forces to a Form." In Kepes, Gyorgy. 1966. *The man-made object*. London, Studio Vista.

- Alexander, Christopher. Autumn 1967. "The Question of Computers in Design". *Landscape*, pp. 8-12.

- Arnheim, Rudolf. 1977. *The Dynamics of Architectural Form: Based on the 1975 Mary Duke Biddle Lectures at the Cooper Union*. Berkeley: University of California Press.

- Arnheim, Rudolf. 1971. *Entropy and Art; an Essay on Disorder and Order*. Berkeley, University of California Press.

- Ashby, William Ross. 1966. *Design for a Brain: The Origin of Adaptive Behavior*. [London]: Science Paperbacks.

- Boden, Margaret A. 2004. *The Creative Mind: Myths & Mechanisms*. London; New York: Routledge.

- Broadbent, Geoffrey. 1988. *Design in Architecture: Architecture and the Human Sciences*. London: D. Fulton.

- Broadbent, Geoffrey. 1990. *Emerging Concepts in Urban Space Design*. London; New York: Van Nostrand Reinhold (International).

- Chomsky, Noam. 1957. *Syntactic Structures*. 's-Gravenhage, Mouton.

- Colquhoun, Alan. 1981. *Essays in Architectural Criticism: Modern Architecture and Historical Change*. Cambridge, Mass.: MIT Press.

- D'Arcy Wentworth Thompson. 1992. *On Growth and Form*. Cambridge University Press.

- Durand, Jean-Nicolas-Louis. 2000. *Précis of the Lectures on Architecture; with, Graphic portion of the lectures on architecture; introduction by Antoine Picon; translation by David Britt*. Los

Angeles, CA: Getty Research Institute.

- Eisenman, Peter. 2003. *Giuseppe Terragni: Transformations, Decompositions, Critiques.* New York: Monacelli Press.

- Eisenman, Peter. 1999. *Diagram Diaries*. New York: Universe: Distributed to the U.S. trade by St. Martin's Press.

- Eisenman, Peter. 1987. *Houses of Cards*. New York: Oxford University Press.

- Eisenman, Peter. 1982. *House X*. New York: Rizzoli.

- Eisenman, Peter. 1963. *The Formal Basis of Modern Architecture*, PhD dissertation. London: Cambridge University.

- Flake, Gary William. 1998. *The computational beauty of nature: computer explorations of fractals, chaos, complex systems, and adaptation*. Cambridge, Mass.: MIT Press.

- Frazer, John. 1995. *An evolutionary architecture*. London: Architectural Association.

- Hofstadter, Douglas R. 1980. *Gödel, Escher, Bach: an eternal golden braid*. New York: Vintage Books.

- Hillier, Bill. 1996. Space is the Machine: a configurational theory of architecture. Cambridge; New York: Cambridge University Press.

- J.Holland. *Adpatation in Natural and Artificial Systems*. University of Michigan Press 1975.

- Jones, J. Christopher. 1981. *Design Methods: seeds of human futures*. New York: J. Wiley.

- Knight, Terry. Shape Grammars in Education and Practice: history and prospects, <http://www.mit.edu/~tknight/IJDC/>.

- Knight, Terry and Stiny, George. 2001. “Classical and non-Classical Computation”. In *arq*, Vol 5, No 4.

- Knight, Terry. *2003.* “either/or → and”. *Environment and Planning B: Planning and Design,* volume 30(3).

- Lynn, Greg. 2002. Architectural Laboratories; [compiled and edited by Véronique Patteeuw].

- Lynn, Greg. 1999. *Animate Form*. New York: Princeton Architectural Press.

- Lynn, Greg. 1998. *Folds, Bodies & Blobs*: Collected Essays. Brussels: La Lettre Volee.

Rotterdam: NAi; New York, NY: Available in North, South and Central America through D.A.P., Distributed Art Pubs.

- Madrazo Agudin, Leandro. 1995. The Concept of Type in Architecture: an inquiry into the nature of architectural form. Zurich: The Institute.

- March, Lionel. 1996. "Rulebound Unruliness". *Environment and Planning B: Planning and Design*, volume 23.

- March, Lionel (ed.). 1976. *The Architecture of Form*. Cambridge University Press.

- Hillier, Bill. 1996. *Space is the machine: a configurational theory of architecture*. Cambridge; New York: Cambridge University Press.

- Mitchell, William J. 1990. *The Logic of Architecture: design, computation, and cognition.* Cambridge, Mass.: MIT Press.

- Mitchell, William J. 1977. *Computer-Aided Architectural Design*. New York: Petrocelli/Charter.

- Negroponte, Nicholas Peter. 1970. *The Architecture Machine: toward a more human environment.* Cambridge, Mass.: MIT Press.

- Piaget, Jean, 1970. *Structuralism*, translated and edited by Chaninah Maschler. New York, Basic Books.

- Picon, Antoine and Ponte, Alessandra (ed.); foreword by Ralph Lerner. 2003. *Architecture and the sciences: exchanging metaphors*. New York: Princeton Architectural Press.

- Popper, Karl Raimund. 1972. *Objective Knowledge: An Evolutionary Approach*, Oxford, Clarendon Press.

- Rowe, Colin. 1976. *The Mathematics of the Ideal Villa*. Cambridge, Mass.: MIT Press.

- Schön, Donald A. 1988. "Designing: Rules, types and worlds." *Design Studies* 9, no. 3:133-43.

- Simon, Herbert. 1969. *The Sciences of the Artificial*. Cambridge, Mass.: MIT Press.

- Steadman, Philip. 1979. *The Evolution of Designs: biological analogy in architecture and the applied arts.* Cambridge; New York: Cambridge University Press.

- Steadman, Philip. 1983. *Architectural Morphology: an introduction to the geometry of building plans.* London: Pion.

- Stiny, George. 1980. "Kindergarten grammars: designing with Froebel's building gifts". *Environment and Planning B 3: 461.*

- Stiny G, Gips J. 1978b. "An Evaluation of Palladian Plans." *Environment and Planning B: Planning and Design* 5 199-206. Cambridge, MA: MIT Press.

- T. Liddament. January 1999. "The Computationalist Paradigm in Design Research". *Design Studies* Vol 20, No 1.

- Taylor, Richard. 1999. "Fractal Analysis of Pollock's Drip Paintings." *Nature*, vol 399, p422.

- Wertheimer, Max. 1959. *Productive Thinking*. New York, Harper.

- Yehuda E. Kalay (ed.). 1987. *Computability of design*. New York: Wiley.

Special Note: Unless specified all images are the property of the author.

S-001 - 8