

## A FRACTAL STUDIO

Chris I. Yessios  
Department of Architecture  
The Ohio State University  
Columbus, Ohio 43210

### ABSTRACT

The experience of using computer aided architectural design tools in a second year graduate studio is presented. These tools had to be developed as the search for design solutions evolved. The computer has been used for the exploration and generation of architectural forms and very little as a drafting/rendering machine. The generative algorithms were based on fractal geometries, arabesque ornamentations, DNA/RNA biological processes, etc. The design problem was a Biological Research Complex. The whole experience raised some interesting pedagogical questions, which are also discussed.

### INTRODUCTION

This is a report on an atypical graduate design studio course at the Ohio State University Department of Architecture. It was atypical for a variety of reasons, but primarily because the computer was used as an explorer and generator of architectural forms. The generators were based on fractal geometries, arabesque ornamentations, DNA/RNA biological processes, transitional processes and Fibonacci series. Thus the title of the paper would need to be longer to make it completely descriptive. However, as it is, it describes well the manner in which the studio course itself evolved.

There was a basic architectural program, a team of students and instructors (the players), a deadline, an early realization that no conventional design method could produce a desirable solution, an early suspicion that the computer might be useful, a rather advanced architectural modeling oriented CAD system, but no software which could even come close to what we needed to do. As ideas started shaping up, so did our computer tools. The existing modeling system offered us a base. Additional modules of software were gradually added (almost as fractal generators are applied to the basic shapes). One module led to another, each time a member of the group started wondering: can we do this?

The generative process of the software itself had to stop at a certain moment, due to the deadlines we were faced with. But it is by no means complete yet and we expect to continue our explorations and the further development of the software, through some other studio course, in the near future. At this time, we do not fully understand where it might lead us. We do not even fully understand the potential of what is already in place. Actually, the potential

appears to be virtually infinite. It is leading us to compositional schemes which we could never had conceived on our own, but the computer is able to unfold for us. And yet, we programmed the computer; we told it what to do. This is not as paradoxical as it may sound. It certainly underlines the potential of the machine as a "reinforcer" of our creative processes.

#### THE PLAYERS, THEIR ROLES AND THE PROJECT PHASES

Peter Eisenman, who I assume needs no further introduction, has been a Visiting Professor at our school for the past three years. This was his studio for the Spring and Summer quarters. He was on campus one day a week.

Ben Gianni is a junior architect and design faculty in our school. He has been a former student of Eisenman and has also worked at his office in the past. He was the daily instructor and critic. Thomas Leeser and Mark Wigley are architectural designers working at Eisenman's office; Thomas for a number of years, Mark this past year. They were visiting critics to the course. They came a couple of times at the beginning of the Spring term and more frequently towards the end. They offered critique to the students and exchanged ideas. They also used the computer themselves, as they explored their own solutions to the design problem.

Seven 6th year graduate design students were joined by two students who are specializing in computer aided architectural design (CAAD). In our school, the Graduate Program in CAAD is a Master level specialized curriculum which includes about seven courses taken directly in the Computer Science Department, and some five courses in CAAD taken in the Department of Architecture, in addition to the regular architectural design/theory core courses. It also requires the completion of an independently developed thesis project. The CAAD students who joined this studio had completed all their course requirements, except for their thesis project.

I have been responsible for the CAAD courses and research in our school for a number of years. I have an architectural background and have taught studios in the past. My main role was that of an "interpreter": translating intuitive design notions to workable tools. The two CAAD students also made major contributions to these interpretations and they trained all the other students how to use the computer. But their main role was in design, just like the other students.

When I first met Eisenman, some three years ago, he was on the negative side with respect to the use of computers in architectural design. He is now a strong advocate. I am flattered to believe that demonstrations of our work and some discussions we had contributed to changing his mind. But it is primarily his own inquisitive nature which made him recognize the potential, when he saw it. What he sees, of course, does not have much to do with the computer usages commonly encountered in architectural offices. He sees them as generators of architectural forms. As this has also been a main interest of mine over the past few years, we somehow agreed to join forces.

The course was one of five graduate design studios the students could choose from. Participation by the CAAD students was optional and two decided to take it. The students had to design a Biological Research Facility for the

University of Frankfurt, West Germany. It was actually a closed international competition to which Eisenman was invited to participate. Thus, as the students were searching and working on their design solutions, work was also going on at Eisenman's office in New York. This offered the opportunity for a continuous intellectual intercourse between a professional environment which had to be concerned with the realities of the project and the associated deadlines, and an academic environment which could afford to experiment and explore. As it developed, the whole design process consisted of two phases.

Phase I was mostly exploratory and roughly coincided with the Spring quarter (April-June). During this phase, the computer tools evolved and a variety of more or less complete design solutions were produced, with frequently specific directives provided by the studio critics. Phase II was the final decision making and production phase. It roughly coincided with the Summer quarter (July-August). In this phase the computer was used as a production tool. After a design solution was finalized, 3-D models were generated and perspectives were rendered. But most of the final presentation, which included a physical scale model, was done by traditional manual methods at Eisenman's office, in New York. Thus the four students who continued the studio course into the Summer quarter moved and worked in New York.

#### THE ARCHITECTURAL PROGRAM

It would be beyond the scope of this presentation to give complete details of the program of the building. It consists primarily of laboratories (some of a restricted character), office spaces and lecture/seminar rooms. The essence of the design problem without question lies beyond the list of the spatial requirements. The following paragraphs from the program text are indicative of the flavor of the solution required.

"The concept of the interdisciplinary biology center is based on the idea of a conceptual and methodological multiplicity of close contact points under one roof.

...The planned extension of the building on the site should intensify the current research cooperation between the departments (including the already existing department of chemistry). Spatial connections will enable spontaneous exchange of thought and mutual support and the development of new cooperative projects.

...Suggestions will be favored that allow the possibility of micro and macro expansions. ...All expansions should have the same functional and environmental qualities as those specified in the program."

How does one address these requirements? Eisenman's suggestion: through fractal geometries, arabesque ornamentation and DNA/RNA biological processes. They are all generative processes, which, if properly regulated and left loose, should be able to produce the equivalent of an interdisciplinary biology center, which can expand in response to as yet unknown future requirements. But in no way should these processes be copied in an iconic manner. They should, instead, be simulated as processes which may lead to a correctly functioning building of a possibly abstract form.

These guidelines were given by Eisenman at the very beginning and became part of the program requirements. At least to the extent they could be, since they were quite general and abstract. They needed refinement, specificity and identification of relevance. This then was the design problem which the students and the rest of us had to address.

## THE FIRST PHASE AND THE EVOLUTION OF THE TOOLS

The design process was initially allowed to be relatively unstructured. Group and individual activities were intermixed. A schematic functional diagram was made available right from the beginning, but was almost immediately pushed to the side and all of us had to research and learn (or re-learn) fractals, arabesque, DNA/RNA, Fibonacci series and any other generative process which was potentially relevant.

The group quickly agreed (or was persuaded by Eisenman) that the problem was not one of arranging spaces and securing appropriate and sufficient linkages. It was rather a problem of designing the generative process which would be capable of growing to whatever level is necessary by current programmatic requirements. As needs for further growth would later manifest themselves, the generative process would exit its frozen state to produce the next state of the building.

At the very beginning we all agreed that the computer must be useful in searching such a solution, but we also agreed that it should be used only if it proved to be truly productive. As the specifics of the design methodology started to reveal themselves through brain storming sessions and frequently heated discussions, there were bad news for the computer. In spite of the availability of some rather advanced modeling software, we had nothing which might even remotely approach the kind of procedures we had started to identify. [1] At the same time it was becoming apparent that what we were talking about could only be done on the computer. Thus we almost had no choice. The design process had to include the development of new computer tools. There is no way I can give a complete report on the ups and downs and all the stages our work went through. Our tools evolved a piece at a time and in parallel with the rest of the design process. Following is a summary description of where they are at the time of this writing.

The fractal, as a generative system, consists of an initial state of a shape (the base) and one or more generators. The generator, from the practical point of view, is a production rule: replace each and every line segment of the base with the shape of the generator. For example, given the base and the generator shown in Figure 1, the first twenty states generated by twenty applications of the generator are also shown, one step at a time.

Our implementation is interactive and allows the fractal to be generated one step at a time or at multiple increments, it allows up to five generators to be applied, generators can be changed (replaced, deleted, inserted) at execution time, the generative process can go forward as well as backward, thus allowing the designer to return to earlier states. Any state of the fractal can be saved for later retrieval. The fractal process can be applied to pre-stored or interactively drawn shapes.

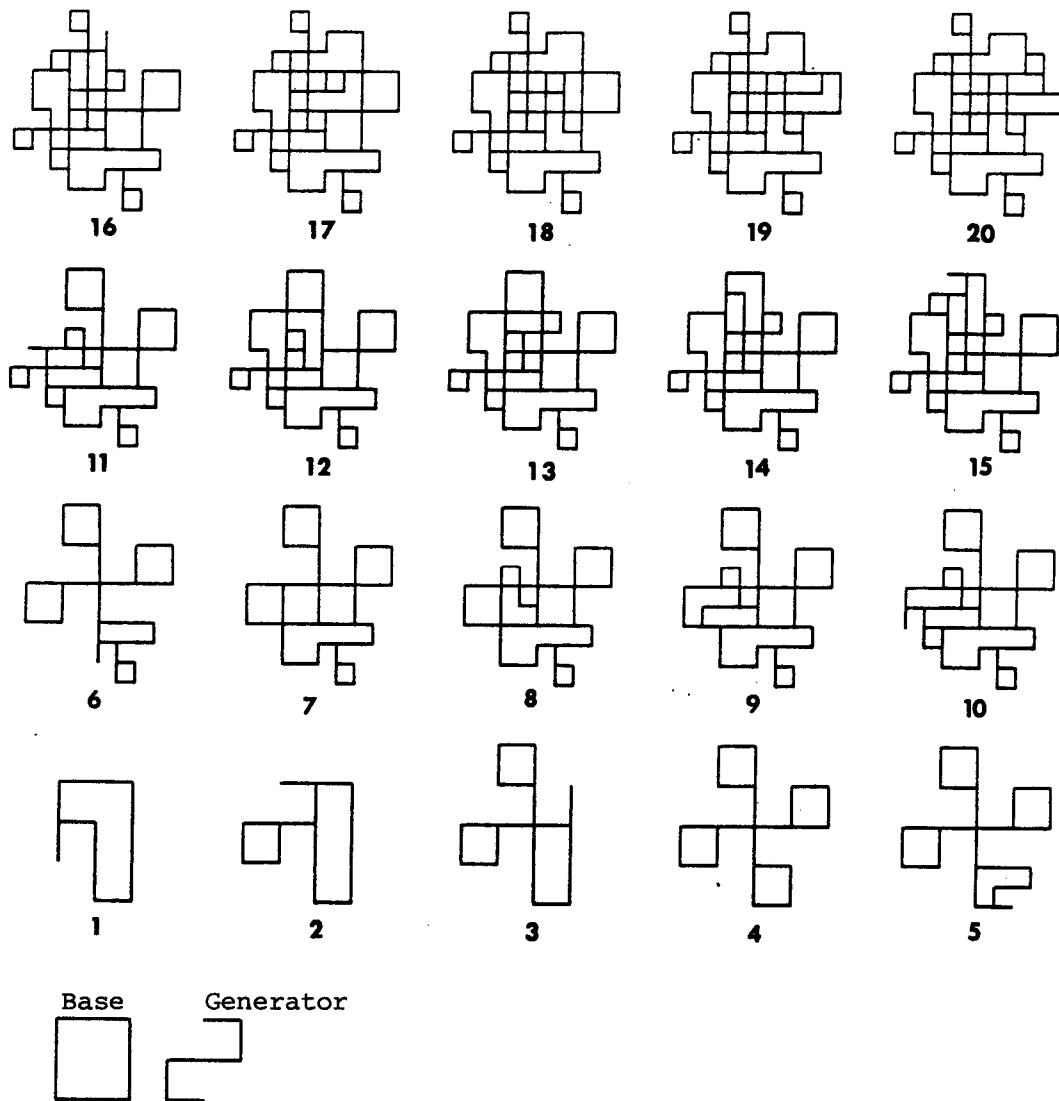


Figure 1: The fractal process.

The theory of fractals has been rather extensively documented by Benoit Mandelbrot. [2 and 3] Our challenge was to make a design tool out of it. A fractal process, if left unrestrained, will go on for ever. In addition, if applied in a "pure" fashion, it will create an interesting shape but will never produce a building. A building typically has to respond to a multiplicity of processes, superimposed or interwoven. Therefore, the fractal process needs to be guided, to be constrained and to be filtered. We preferred the biological term of "mutation". The fractal process has to be mutated by the utilitarian requirements of the functionalities of a building.

Some of these concepts are further illustrated in Figure 2. The initial shape (base) can itself be the product of a fractal or some other generative process. Applying generators I and II individually results to distinctly different shapes. The two generators can then be mixed uniformly to produce the fractal shape in Figure 2(e). Or the mixture of the generation can be controlled and regulated by the designer to produce a shape such as the one shown in Figure 2(f).

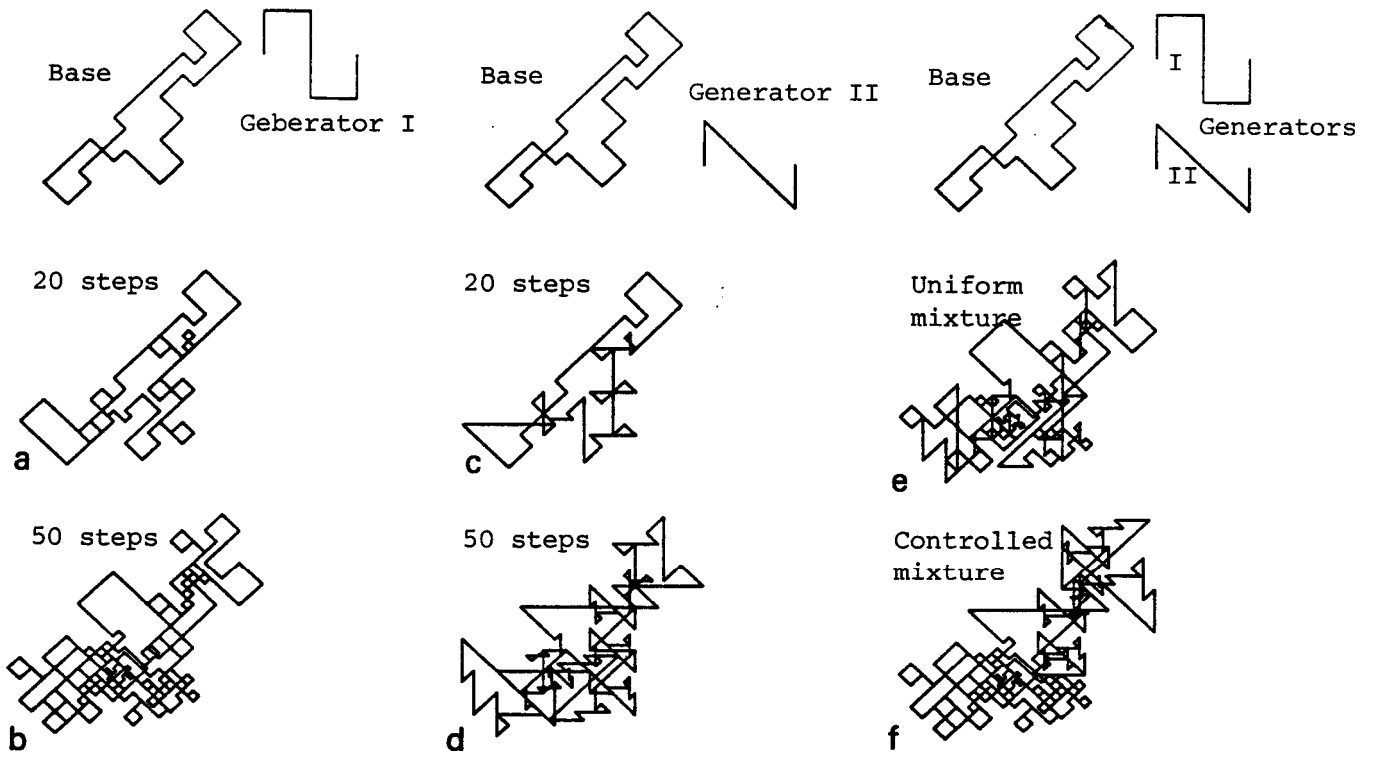


Figure 2: Applying two generators, one at a time & mixed.

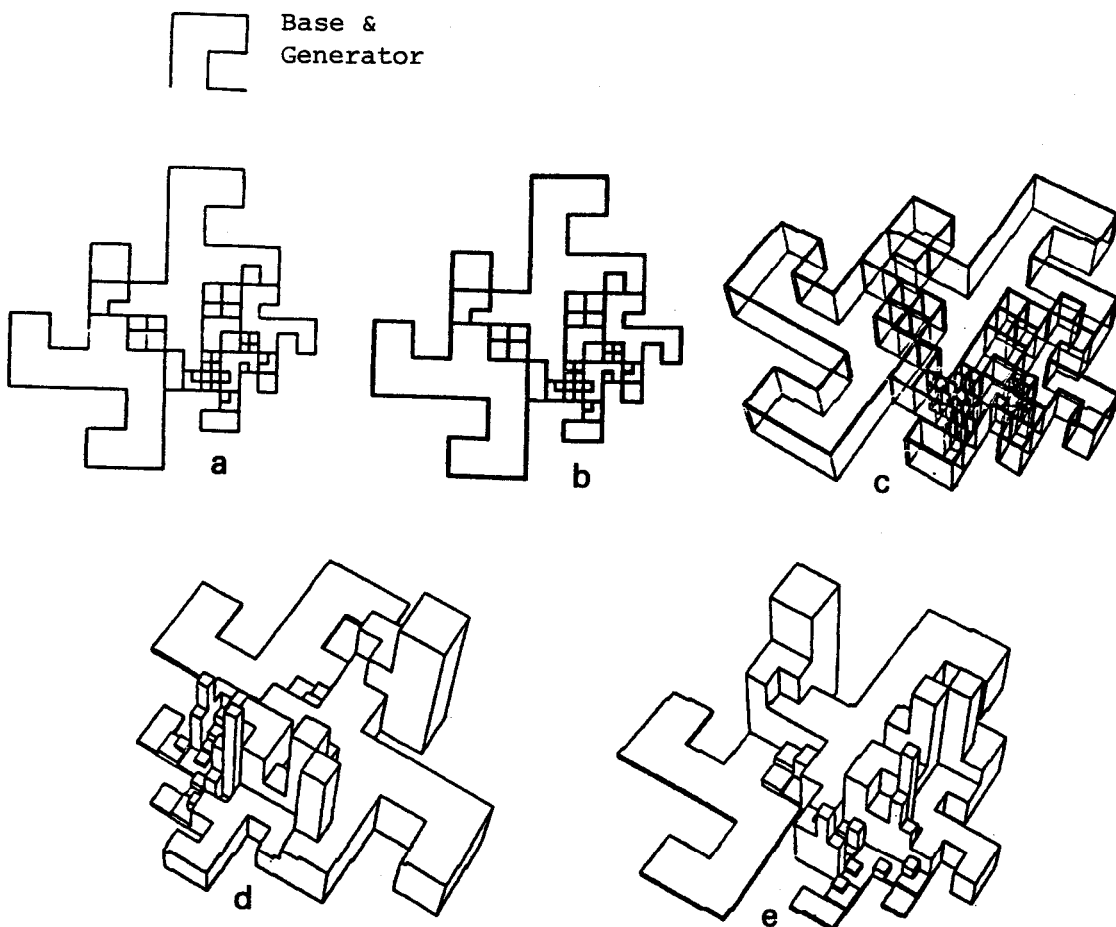


Figure 3: Fractal designs in two and three dimensions.

Another interesting example is shown in Figure 3. Here the base and the generator are one and the same shape. The fractal generated after 30 steps (3(a)) has been filtered and transformed into a structured drawing (floor plan) which next becomes the base for generating a 3-D building model. An interior model is shown in 3(c) and two views of the exterior model are shown in 3(d) and (e). How were the heights determined? The user can make his/her own choice, but in the example the Fibonacci series was applied.

The examples I have used appear to be quite irregular shapes, but they do not have to be. An appropriate choice of generators can produce completely symmetric compositions. But symmetries and regularities are more inherent in the arabesque ornaments. At least at first glance, an arabesque ornament (such as the one shown in Figure 4) appears to be highly symmetric and to consist of locally finite shapes. But when we take a closer look and trace the "laces" it consists of, each has a very irregular shape and traverses a long distance. It some times eventually closes into itself. Other times it never ends. One such lace has been highlighted in Figure 4.

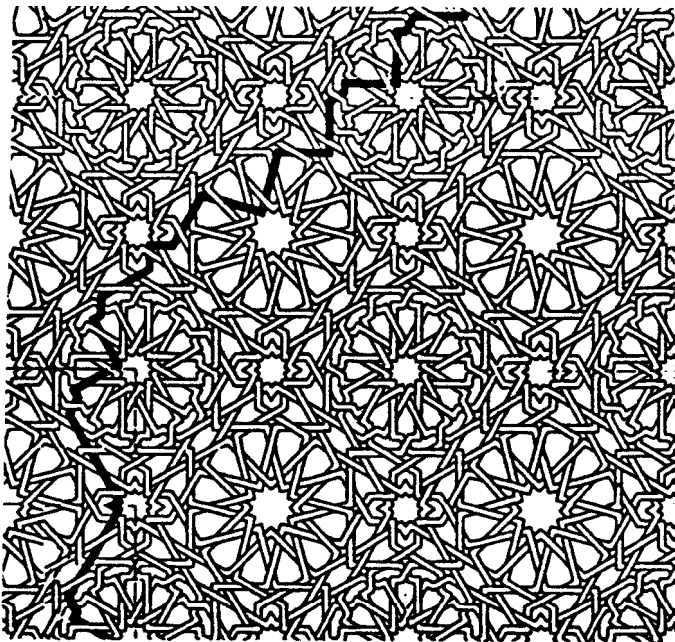


Figure 4: A typical arabesque pattern.

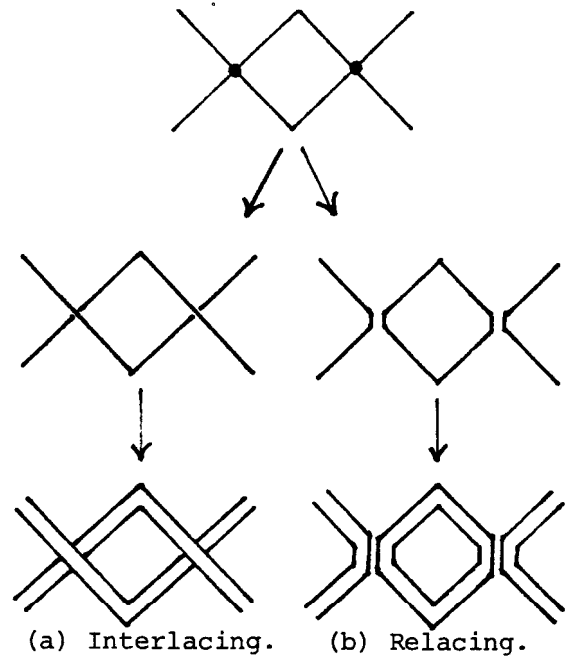


Figure 5: Arabesque operations

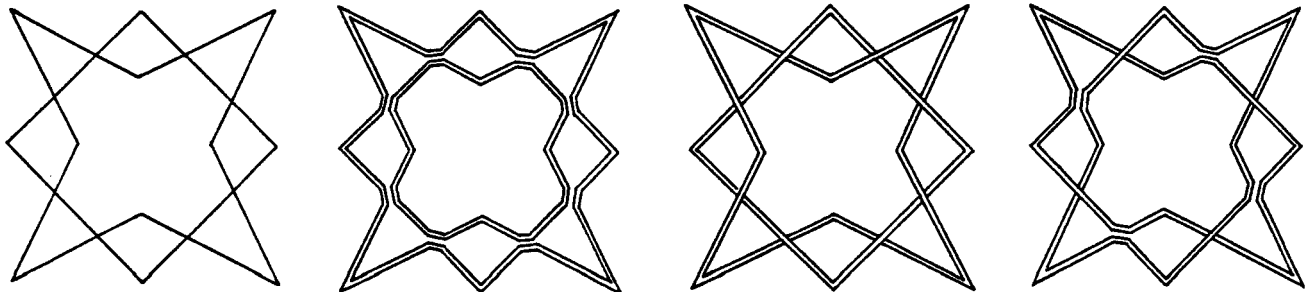


Figure 6: (a) Relacing only. (b) Interlacing only. (c) Relacing & Interlacing.

The arabesque process typically starts with a symmetric arrangement of simple basic shapes, such as squares, diamonds, circles etc., which overlap each other to form a repetitive pattern. [4] Then any one or both of two operations may be applied. The relacing operation causes two intersecting line segments to be rerouted so that each jumps to the other. After they are rerouted, the angle at the point of intersection can be rounded in a variety of ways (Figure 5(b)). The interlacing operation causes one of two intersecting line segments to pass under the other. When the under/over condition is applied at every other point of intersection, a "weaving" effect is derived. It is more visible when the laces are represented with double lines, as is typical with arabesque ornaments (Figure 5(a)). Examples of arabesque applied to two overlapping shapes (a diamond and an asterisk) are shown in Figure 6, where in 6(a) only relacing has been applied, in 6(b) interlacing only and in 6(c) both relacing and interlacing, at even intervals.

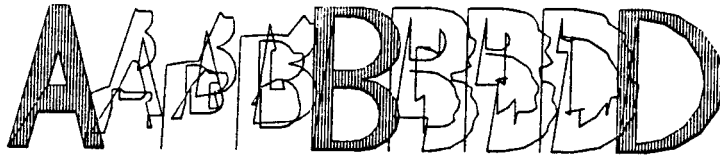
As a generative process the arabesque can go to infinity, but it is typically applied as a process which closes into itself. While traditionally the arabesque is applied to regular and symmetric patterns, there is no reason it cannot be applied to irregular shapes as well. Actually, fractal shapes can be very nicely interlaced and relaced. Again, our implementation of the arabesque procedures aimed at a high degree of interactivity, thus allowing the user to control the flow of the generation.

Given two shapes, arbitrarily different or self-similar, can the computer "discover" what set of rules transforms one to the other? As buildings need to expand, can such procedures tell us how to go from the current state to the new state of the building design and do it in a way consistent with the design principles applied in the original building? Would such techniques address the problem of designing highly expandable buildings, such as required by our current program? We cannot claim that we have found a universally applicable method, but we conducted some interesting explorations.

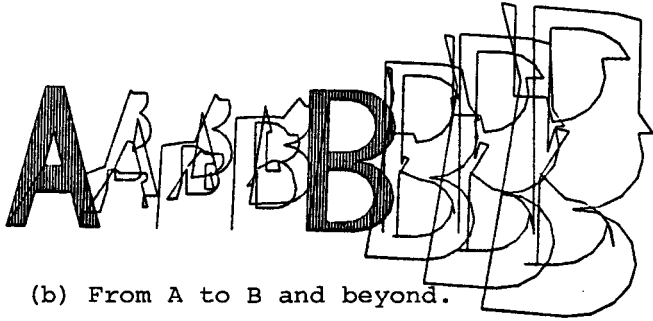
The transition process transforms an initial shape A to a target shape B by applying any number of in-between steps. The points of shape A are mapped into shape B and vice versa. Then each point of A has a corresponding point in B and vice versa. A point in A can go to its corresponding point in B in a variety of ways. For example, the transitions illustrated in Figure 7 apply a simple linear interpolation. More than two shapes can be "chained" as illustrated in Figure 7(a), where the letter A goes to the letter B and then to the letter D (in 4 steps or 5 states). The third state is a shape which is the average of A and B (50% A and 50% B). State 2 is more A than B and state 4 is more B than A.

Once the rules of the transition have been established, the transition can be allowed to continue beyond its target, to infinity. In Figure 7(b) A goes to B as before, but continues beyond B for another three steps. In 7(c) D goes to B and again continues for another three steps. Figure 7(c) also illustrates that going from B to D is geometrically equivalent with going from D to B (the reverse), provided the same steps and transition rules are applied. Figure 8 contains some more examples of transitions which start to illustrate the potential of the transitions as a design tool. To borrow Eisenman's terms, is it rhythm in what at first glance appears to be chaotic? Or is it "catachrisis", meaning we have gone too far?

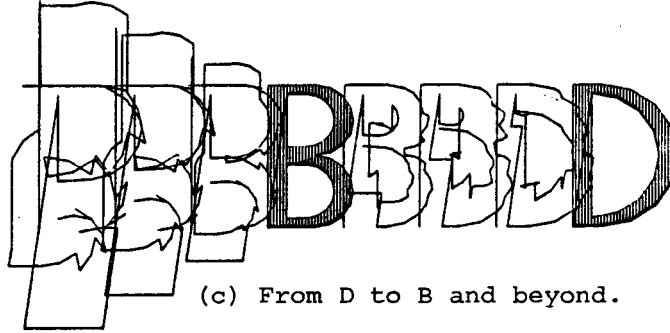




(a) From A to B to D.



(b) From A to B and beyond.



(c) From D to B and beyond.

Figure 7: Transitions.

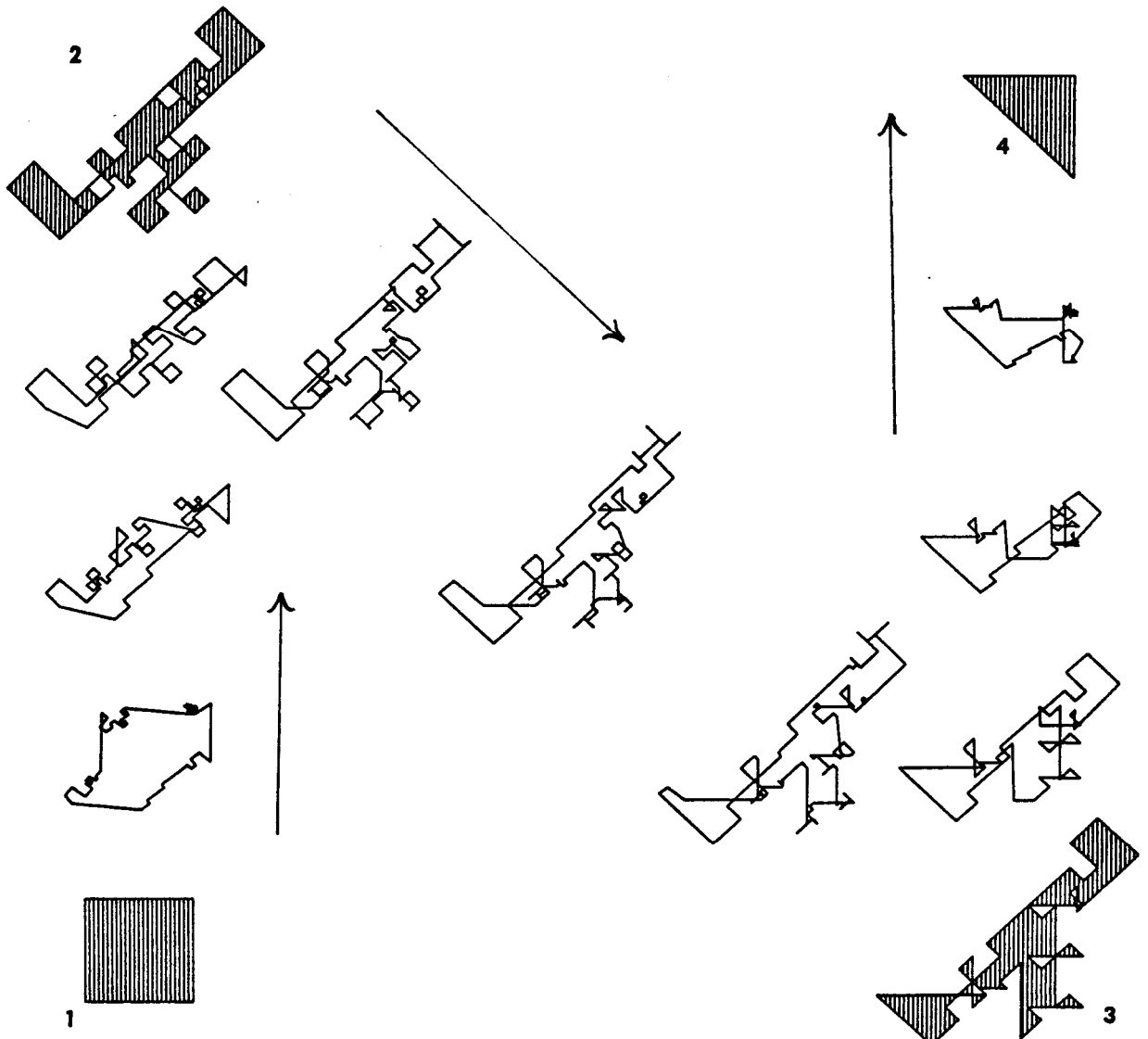


Figure 8: From square (1) to fractal (2) to fractal (3) to triangle (4).

I do not show anything relating to the DNA/RNA process, even though I mentioned it as another generative process which we explored. We have not yet been able to produce the tools which would apply the biological processes of replication, transcription and translation to spatial configurations. At least not in a way which does not lead to icons of spirals, which were ruled undesirable right from the beginning. Yet, the students were encouraged, even required, to apply the logic of the biological processes as they worked with the other generative processes and as they dealt with the programmatic requirements of the building. A variety of preliminary designs were produced by the end of the first phase and some are illustrated in Figure 9.

#### THE SECOND PHASE AND THE FINAL SOLUTION

Space limitations will only allow me to highlight the logic and the symbolism which went into the final solution. The following are taken directly from the text which accompanied the competition submission . [5]

... The project form is the result of the action of the three most basic processes by which DNA constructs protein: replication, transcription, and translation, on the geometric figures that biologists use to explain these processes... (Figure 10(1)). The particular shapes of the inner faces of these figures indicate that they are capable of locking together as two pairs (Figure 10(2)). The blueprint for every protein is encoded in long sequences of these paired figures which form a double stranded chain (Figure 10(3)). Using an analogy between biological construction and architectural construction, this chain can be transposed into architectural form in such a way that it produces an architecture which becomes complicit with the discipline which it houses.

In this project, the biologists's figures were overlaid upon the site in a row beginning at the main entrance and following the precise sequence of the DNA chain ..., which produces the necessary tensile strength of biological structure (Figure 10(4)). The five pairs of figures which lay closest to the existing chemistry building became the laboratories for our project. In this way, an initial architectural structure was produced out of the figurative code for biological structure.

The project was then produced by subjecting that figurative code to the very processes which it describes. These biological processes were interpreted architecturally by the use of fractal geometry. ... The processes of fractal geometry can also be described by the similar processes of replication, transcription, and translation. Through this analogy, the original five pairs of figures could be successively transformed by fractal processes in a way that articulates the three basic DNA processes.

The basic steps of these combined DNA and fractal processes, as well as the final building form, are shown in Figure 10 (5 through 7). Color renderings will be shown at the Conference.

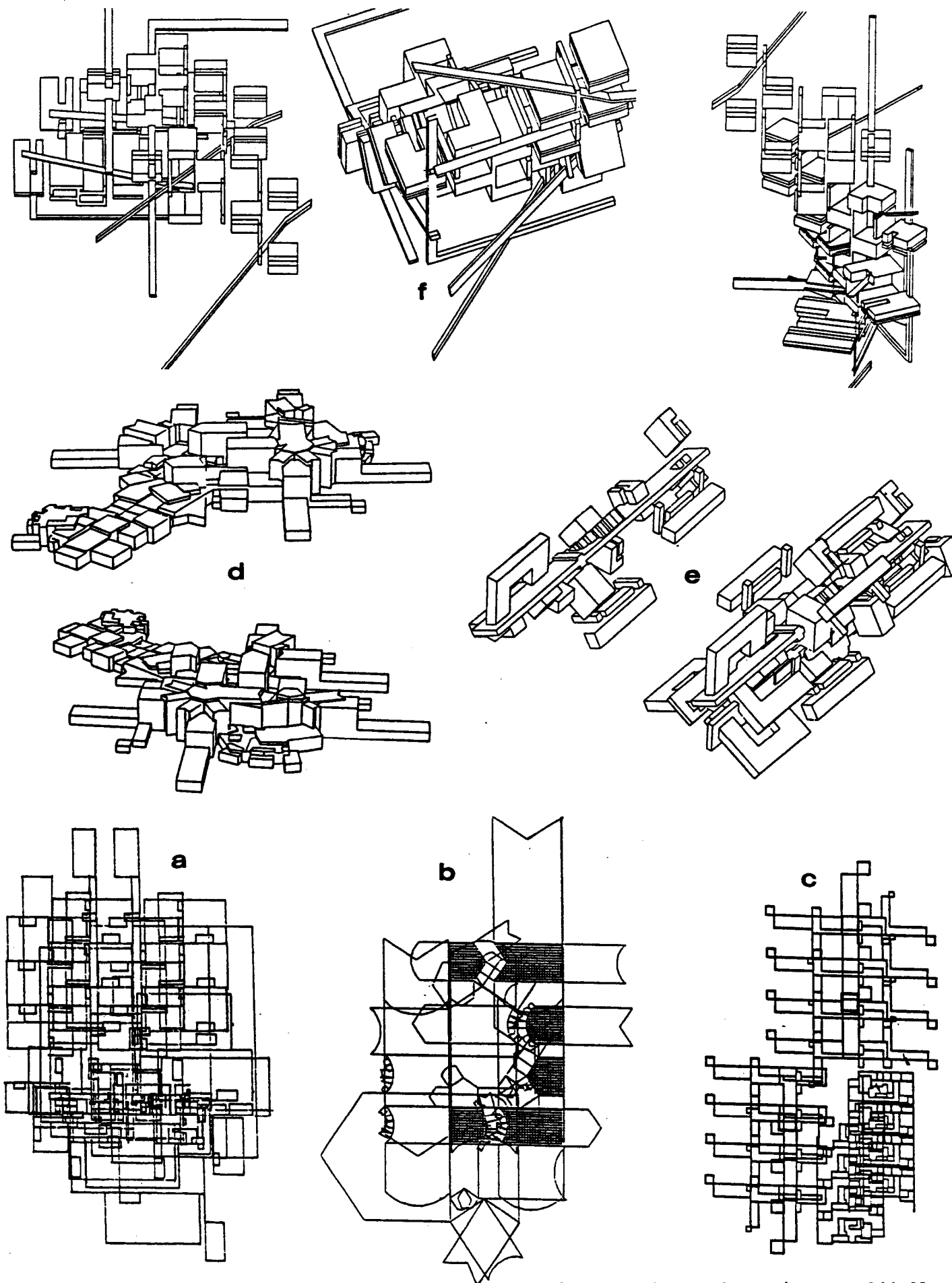


Figure 9: Fractal and DNA/RNA based design explorations by the students in ARCH 844.01.

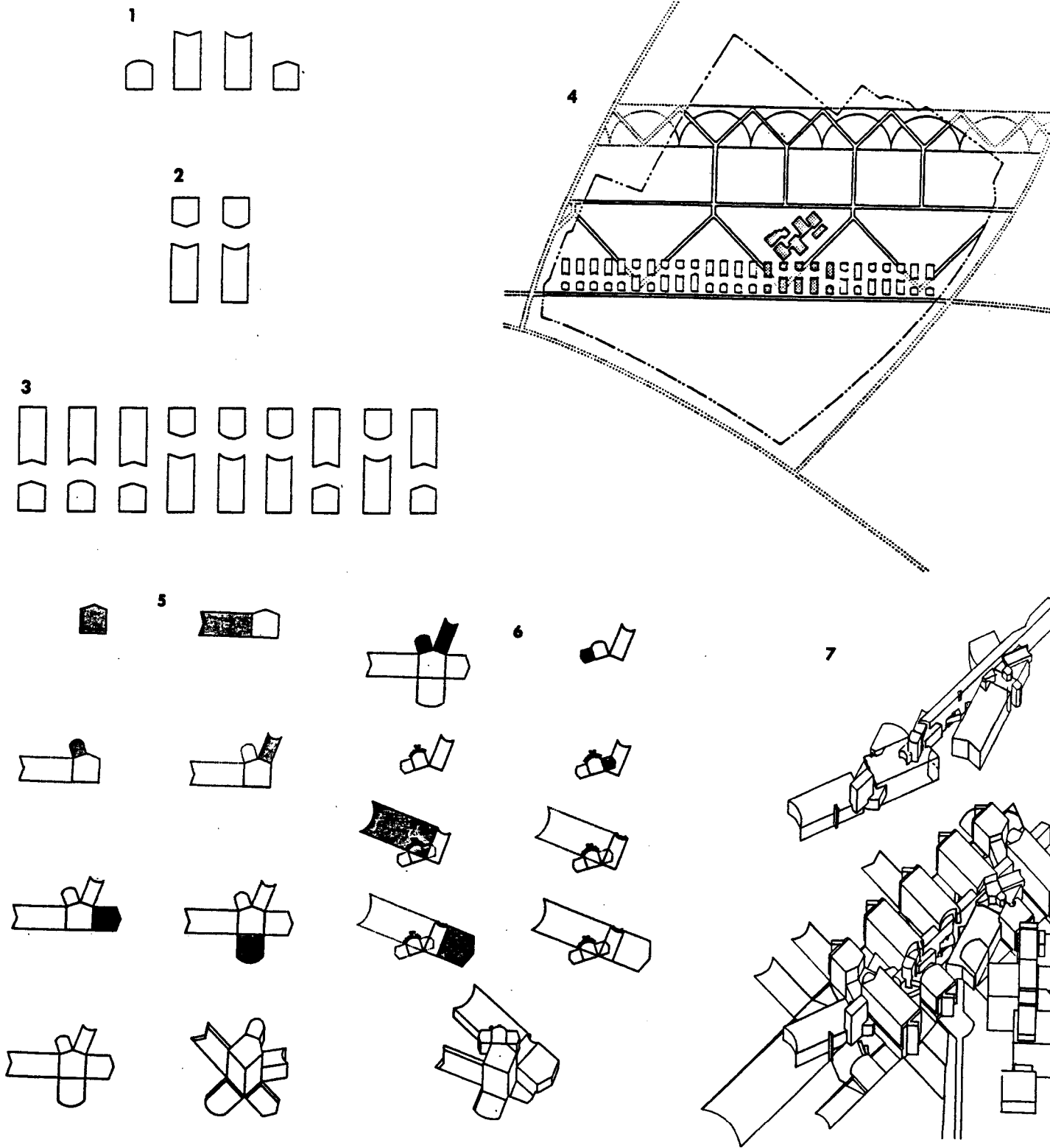


Figure 10: Highlights of the final design and how it was derived, through fractals & DNA

## CONCLUSION: SOME PEDAGOGICAL ISSUES

Regardless of whether or not the competition will be won, this studio has been an extremely valuable experience for all of us; students and instructors alike. Not only was it highly educational, but also enjoyable. In addition, it raised some interesting pedagogical questions.

In our schools, we all try to figure out how much computers and in what ways we should incorporate them in our architectural curricula. Some recurring issues are tool-using versus tool-building, and how much computer programming, if any, is appropriate. I shall avoid drawing explicit conclusions with respect to these issues and would rather let the readers do it on their own. But certain points need to be made.

All the "traditional" design instructors and students in the course were honestly declaring that they were computer illiterate. Yet, I found most of them, Peter Eisenman in particular, to have a high sense of algorithmic processes, which is what computer programming for architects is really what it should be all about. As they had to rely upon somebody else to write the code for them, I have no doubt that some of their original concepts were partially lost or distorted. On the other hand, whenever some variation was added to the initial concept, there was very little interest by the originator to use it. In general, each designer tended to use primarily those parts of the computer tools which he/she participated in proposing. It is probable that the whole design process might have been even more productive and imaginative if the designers were also the developers of the tools they considered desirable.

Another interesting development has to do with the role the CAAD students ended up playing. It was initially assumed that they would be the "interpreters" who would develop the required computer tools, with my supervision. As the course proceeded, they were on the one hand absorbed by the design process itself and on the other hand they had to train all the other students how to use the computer. Actually, the course would have never worked without their latter contribution. But this left them very little time to involve themselves with the development of the tools, most of which I had to do myself. At a certain point they also seemed to declare that they had no interest in developing computer tools. But they were expecting to design by using the computer, a truly interesting contradiction.

As we constantly complain about the current state of commercial CAD and some of us research the methods for producing truly design oriented machines, after an experience such as the one reported here, one cannot help it wondering: Will there ever exist a universally satisfactory CAD system? Is it time to start researching what might be called "artificially creative" systems?

- REFERENCES:
- [1] Yessios, C. I., ARCHIMODOS: User Manual, CAAED Lab, The Ohio State University, Columbus, OH, 1984.
  - [2] Mandelbrot, B. B., The Fractal Geometry of Nature, W. H. Freeman and Company, San Francisco, 1983.
  - [3] Peitzen, H-O. and Richter, P. H., The Beauty of Fractals: Images of Complex Dynamic Systems, Springer, Berlin, 1986.
  - [4] Critchlow, K., Islamic Patterns,
  - [5] Eisenman, P. et al., Biozentrum der Universitat Frankfurt am Main, Design Competition, New York, NY, 1987.