Artificial intelligence and Architectural Design

Celestino Soddu, Enrica Colabella

CAAD Futures '95, Singapore 1995

Introduction

An experimental approach to the design action, building a new tool that increases the design creativeness with the preview of a universe of possible evolutions.

The design of architecture is based on the ability in thinking the multiplicity of possible incoming scenarios. The quality of this design is connected to the way used in checking the evolution when it occurs, and we must do that following the increasing complexity.

The complexity cannot be born ex-novo, generated by a unique action.

The complexity is the result of a developmental process. The complexity acquires its peculiar characteristic in front of the temporality of a developing path, using the stratification of multiple choices in front of the manifold different moments and opportunities. If we want to build the complexity, we must use a multiplicity of extemporaneous and subjective keys of approach, applied on different and, maybe, apparently contradictory fields.

Then, the quality of a design is strongly linked to the logic approach that we adopt in the operative action. The quality is linked to the procedures that we have activated in developing the first hypothesis. The quality is not originated by the first idea that is, normally, strongly contingent and axiomatic.

We must use the first idea like a catalyst to start up the process and, after this beginning, we can also forget it.

The achieved quality/complexity of the professional results in architectural design is measured by the presence of a concurrent manifoldness of different possible approaches, of possible ways of custom, of possible performances that the subjectivity of the users discovers like a mirror of own uniqueness.

But the quality of a project is also based on the recognisability of the idea of architecture, of the environment, that each architect pursues in the time of his design process, using, as design field, the space of randomness generated by his discovering of a set of multiple possibilities in front of the same request. The designer can use these spaces of randomness like evolution field to increase the complexity and the performances of his project, and so to respond to his subjective conceptual needs.

In this operative scenario, that identifies in the structure of the approaches the key to controlling the quality of the results, the CAAD, and simulation tools present, from a part, indubitable advantages for what involves the quickness of the iterative processes and the preventive control of the results.

From the other side, these tools have increased the possibility of bypassing some steps of this process, and then of degenerate it. This fact can happen when we use the CAAD tools like the drafting machines and the typological patterns, whose potentiality of control is

performed like interdiction of some possible field of evolutions and not like the way to amplify the incoming possibilities of the design.

The tools like CAAD and Rendering have definite a first substantial change in approaching the design activity. In the beginning, it seemed foreseeable that, with the handmade sketches, the ambiguity necessarily disappear. And, with the ambiguity, one of the essential characters of the sketch when it is used as a tool of design reasoning, to evaluate possible new evolution paths.

This hypothesis has been demonstrated false. We have considered the ambiguity of the sketch as semantic manifoldness, as an adaptive tool to discover possible design evolutions. This potentiality remains, and the sketch on paper becomes a more effective tool because it acquires autonomy.

Not only. In this field of approach, now we can also use the ambiguity of the computerised simulation, of the virtual reality.

The virtual reality has opened another front that can use the ambiguity like an operational tool.

It presents the potentialities to activate unexpected reflections: the manifoldness of the possible parallel scenarios reported with the quality of the final representation, the multiple and parallel representation of the same design idea.

Following this possibility, we are now only at the first steps. The actual CAAD follows the layout of the antecedent tool, the drafting machine, and the typological references. This is strongly coercive because these tools adopt integrate and optimised pre-designed forms.

But respect to these old tools, the CAAD, as all the software, has an internal availability to the evolution that allows us to get a total overturn: from the power of interdiction to the adaptivity, from the closing performed with static models to the ability in supporting the subjectivity and the differences.

We must use, in the next future, these potentialities of the CAAD, its structure strongly diversified among interface and executor, the possible metamorphosis of the interface from a set of buttons to click, into an artificial intelligence shell. This interface must learn from the designer and must amplify his potentiality with the generation of a set of possible scenarios in harmony with the designing idea of the architect.

This is not a far scenario. The experimental tools realised following this approach exist and they are usable.

The software that we have realised (the last version of BASILICA) operates like artificial intelligence interface to generate a multiplicity of virtual scenarios, a diversified universe of possible and the parallel reality that is the representation, in real time, of the designing idea. This tool follows and increases the architectural idea of each designer, and his creativity. Like the ambiguity of old sketches, this tool supports the fields of possible designing evolutions with the manifoldness of unpredictable virtual scenarios. These scenarios, generated in real time, are always different, but they are also always identifiable as one of the possible representation of the same designing idea. The tool works with the morphogenetical code of the subjective idea concerning the architectural environment that the designer builds step by step with his natural designing approach.

The incoming CAAD tools, however, are going to be characterised in two types. We must not get confusion among software that is performed like executors (the CAD as electronic drafting machines, or the tools for the Rendering, to create electronic maquettes) and the software that are performed like an interface between man and drafting machine.

These interface tools identify the logic procedures in approaching the design process. Today these tools are integrated with CAD software and support only the management of the design approach based on typological date base.

In many of these interfaces, tools persist, in this moment, a carelessness and superficiality that affect the advantages of using information technology in architectural design. To attract customers making easy the interface, the CAAD producers follows simplification that inhibits the access to potentiality already present in the tool. To get these potentiality, the ways we can follow are two: first, the increasing complexity of the user interface (in other sectors this approach has generated monsters, like super remote-control devices with hundreds of buttons of varied forms and colours); second, building a dynamic interface trainable from the designer, during his "normal" professional activity.

This new type of interface must be a tool of artificial intelligence able to apprehend from the designer using the application of relations between input and output, between requests and possible formal matrixes. So we can leave the use of axiomatic typological references to enter in the field of the morphological and subjective ones. Leaving also the false optimisation of "objective" database that pre-defines a nonexistent inferential relation among functional requests and formalised architectural events. Defeating the most interesting bringing in of information technology in architectural design: the ability of management of the complexity.

The interface can be a software that builds, following the logical associations selected by the designer during his work, a subjective meta-design, an explicit and operative structure of his designing idea.

Supporting each step of the designer, the interface generates a universe of possible and parallel scenarios, representing a fan of possible evolution paths. The designer, choosing one (or more) of these scenarios, defines, in progress, the character, the recognisability of his own conceptual approach. In the phases of learning and, afterward, when the tool supports the creativeness for the realisation of the project, the interface uses the performer tool, the CAD for the technical representation and the Rendering shell for the simulation.

In other words, the designer, developing his designing activity, can also design his drafting machine, builds his operative tool, his subjective meta-design, making explicit (and supported by the computer) his logical approach, his identity as a designer.

THE EMULATION OF DESIGN PROCEDURES

A tool that operates in the field of design processes needs to construct, with algorithms, a set of logical procedures that are, in practice, the logical procedures of discovery. These procedures are not analytical.

They are similar to the epistemological structure of scientific discovery.

To use information technologies in the architectural design we need to have a software able to simulate the increasing complexity of the system in front of the progressive stratification of ever different and not necessarily coherent approaches.

The software we have developed, Basilica is an intelligent interface between the designer and the project. It helps the architect using a controllable and changeable logical sequence of morphogenesis. Following the selected approach, the tool can generate a set of different formalised scenarios in response to the progressive multiplicity of questions.

Every generated scenario needs to be different and clearly characterised one. This is because the design procedure is a developing path. In order to simulate it, Basilica uses a nonlinear, unpredictable, and increasingly complex sequence. Using this approach, Basilica can generate a sequence of possible scenarios that can reply not only to the pre-coded questions but also to the unpredictable conceptual needs of the architect.

The algorithms of this software are not organised with a database of shapes and do not refer to a deductive request/reply structure. This is because, in the design processes, every formalised reply exceeds the field of the pertinence of the request. The architects use the random margins between requests and formalised replies to answer to their conceptual needs, to perform their idea of the architecture. The role of these algorithms is to simulate the human procedure of design evolution, of discovering the possible fields and, in the same moment, replying in advance to the unknown and unpredictable answers with a set of possible virtual scenarios.

In other words, Basilica simulates the logical design procedures shaped as a formalising engine that changes every answer into possible choices using a formal matrix. During every cycle, the developing project system gets to an increasingly complex level. As it normally happens during any design process, the software changes what before was random into what after is necessary, because it has just happened.

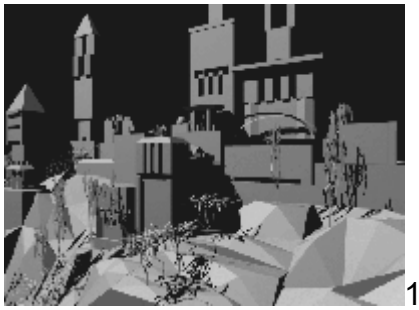
For this reason, we assimilate these algorithmic procedures to the inferential procedures of AI. So we can manage, evaluate and control the increasing complexity and adaptivity of the project, that is the increasing of power to get out intersubjective replies through the explicit stratification of a sequence of subjective contributions.

THE LOGICAL STRUCTURE OF "BASILICA"

The logical and operative structure of the simulation system of Basilica is based on the use of the main cycle, with auto-organization capability, and a set of ever growing secondary cycles. All are bound together.
Each cycle represents a whole structure in simulating the decision choices. It operates the transformation of the answers into possible shapes. This device is designed by
1. The use of a paradigm to control the auto-organization procedures. This tool represents and controls the gained complexity and, in the meantime, is adaptive to incoming developments. This is the device that allows us to reply to a question that puts one of the possible formal matrixes into the paradigm.
2. The identification and sharing of the random margins between questions and shaping reply. The system uses and represents these margins as "operable fields" for the design choices. This improves the project evolution.
3. The set of possible formal matrixes, which are abstract shapes but usable in giving body to a set of possible performances. These formal matrixes are not a database. They are generated by the interconnected cycles, by a set of simultaneous devices operating in a series of different fields, like geometry, dimension, materials, technology, complexity, and so on.
Therefore, every formal matrix is the extemporary transformation of the contaminations and resonances into a set of different subsystems performed as a subsequent paradigm/random margin/formal matrix. All are in a subsequent homothetic complexity that looks like a fractal shape. At the end of every cycle (and of the related and multiple progressive nidifications) the result is:
1. an increasing complexity, and the related passage into a more evolved representation of answers, together with the proliferation of the same answer.

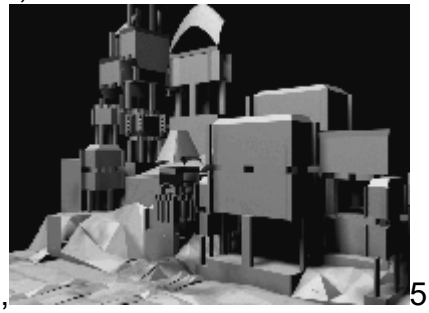
2. The production of needs, for the reason that every event we design was born also using subjective and random approaches. It was not necessary before but it became necessary after the choice: it is a part of the project history. This happens also if we, later, remove it because we consider this event as an obsolete one.
At first, Basilica was only a research software to simulate, using AI procedures, the decision approach to design. Now we can use it for the management of architectural and environmental design, as well as for teaching. To use Basilica as learning tool we have built a user interface to evaluate and control, in different but simultaneous fields, the morphogenesis of the paradigm structure, its evolution and the sharing of the random margins to operate subjective choices oriented towards the intersubjective performances of the project.
The designer can build his own paradigm, change the structure of the relationship between possible events, change the geometry and stratify multiple possible geometries within the paradigm, define the quantity of possible exceptional events and the relationship between these events and the normal structure of the architecture. And, also, the designer can choose the time of evolution that he wishes to simulate. Mainly, Basilica is a tool to design the morphogenetic code of the artificial environments that we can manipulate through a set of quantic parameters.
With Basilica, it is possible to design an artificial DNA able to generate a set of ever different, unpredictable and individually characterised artificial events. Every scenario, that is a 3D computerised virtual model of architecture, is recognisable as an individual of the same species. So we can identify, for example, a species as a typical design of an architect, the universe of possible solutions that the same architect can generate when faced with the same design problem. We have used our software in professional experiences, designing the incoming scenarios of the town environments and of new architectures.
For example, we have designed the DNA of the Italian Medieval towns. The problem was how to control in progress the identity and the recognisability of these environments in front of the increasing complexity of the actual needs. The challenge was to save the differences identified as a morphogenetic code of this type of towns.

The images (Figure 1-5 are a sequence of views of different 3D models generated directly with Basilica, that are different but belonging to the same "species". There are a set of nowhere virtual medieval towns that lived a parallel virtual time.

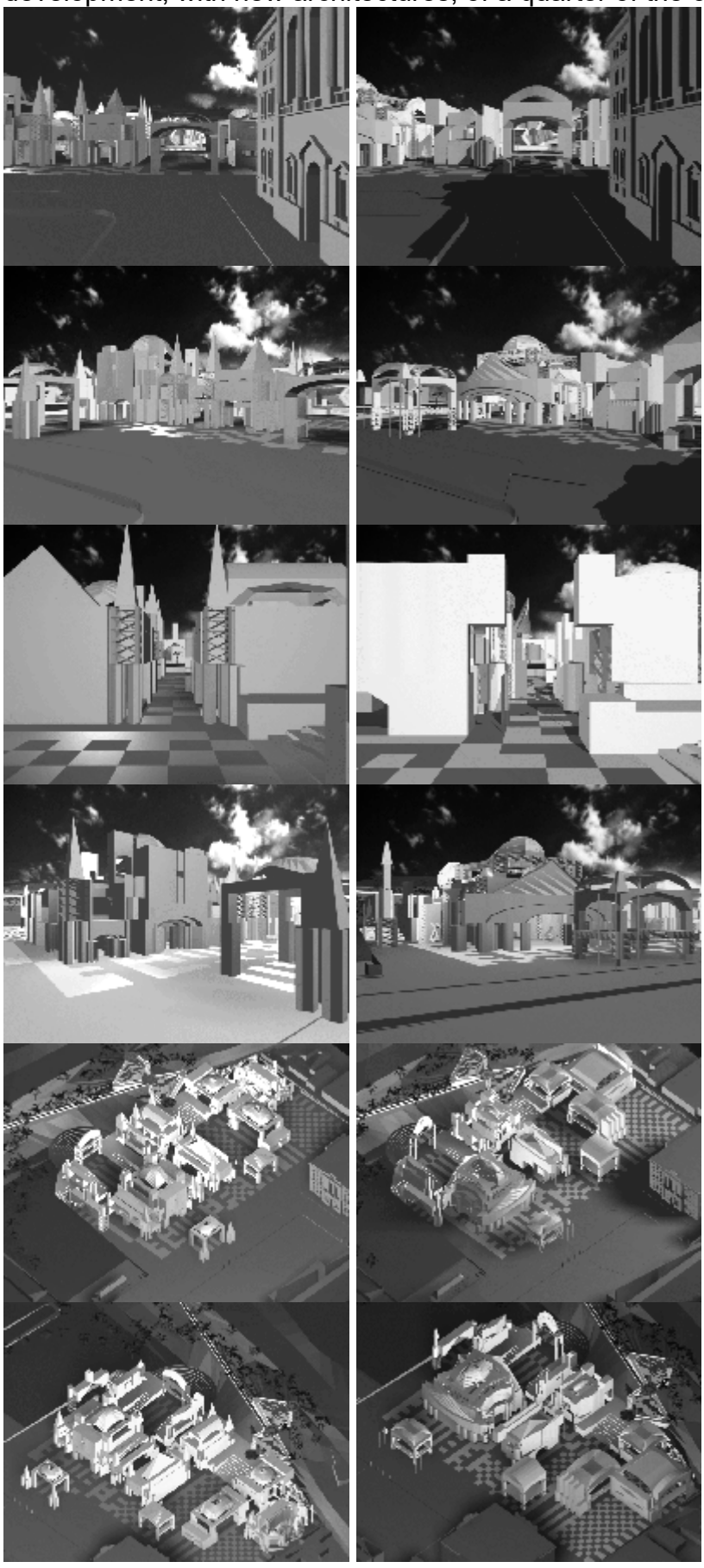

1,

2 ,

3,

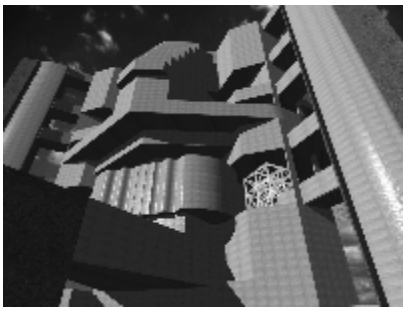
4,

5)

We have also used this program in a more usual professional experience: the development, with new architectures, of a quarter of the centre of Rome (Figure



6-17).
The sequence of these images shows two different parallel scenarios generated automatically with our software, and belonging to the same morphogenetic code that we have designed. The first 3D model is one of the multiple prototypes and the second is the scenario we have used in our project. This scenario has been generated the last. It belongs to the last, more complex and "interesting" generation of scenarios.

The last two images

(Figure 18-19) show an architectural project we have done for Taiwan. We have gained the complexity of these buildings, and I think also this quality, using a sequence of 3D models generated with our software.