Expressive Form

A conceptual approach to computational design

Kostas Terzidis

LONDON AND NEW YORK
Chapter 6
Algorithmic Form

An algorithm is a computational procedure for addressing a problem in a finite number of steps. It involves deduction, induction, abstraction, generalization, and structured logic. It is the systematic extraction of logical principles and the development of a generic solution plan. Algorithmic strategies utilize the search for repetitive patterns, universal principles, interchangeable modules, and inductive links. The intellectual power of an algorithm lies in its ability to infer new knowledge and to extend certain limits of the human intellect.

While many algorithms have been invented and implemented for architectural design in space allocation and planning problems, their implementation in aesthetics and formal theories has been, generally, limited. Most of the theories related to form pertain mainly to subjective interpretation and perception. In contrast, algorithmic logic involves a deterministic approach to form and its shaping forces; it suggests rationality, consistency, coherency, organization, and systemization. What makes algorithmic logic so problematic for architects is that they have maintained an ethos of artistic sensibility and intuitive playfulness in their practice. In contrast, because of its mechanistic nature, an algorithm is perceived as a non-human creation and therefore is considered distant and remote. Traditionally, the dominant mode for discussing creativity in architecture has always been that of intuition and talent, where stylistic ideas are pervaded by an individual, a “star,” or a group of talented partners within the practice. In contrast, an algorithm is a procedure, the result of which is not necessarily credited to its creator. Algorithms are understood as abstract and universal mathematical operations that can be applied to almost any kind or any quantity of elements. For instance, an algorithm in computational geometry is not about the person who invented it but rather about its efficiency, speed, and generality. Consequently, the use of algorithms to address formal problems is regarded suspiciously by some as an attempt to
overlook human sensitivity and creativity and give credit instead to an anonymous, mechanistic, and automated procedure.\textsuperscript{3}

While most algorithms are tailored to automate tedious manual methods, there is a certain category of algorithms that are not aimed at predictable results. Their inductive strategy is to explore generative processes or to simulate complex phenomena. Such inductive algorithms can be regarded as extensions to human thinking and therefore may allow one to leap into areas of unpredictable, unimaginable, and often inconceivable potential. For instance, artificial neural networks are systems of algorithms that simulate the human brain’s functions. They are being used in classification, forecasting, and modeling applications. In making determinations, neural networks use several principles, including gradient-based training, fuzzy logic, genetic algorithms, and Bayesian methods. What distinguishes these algorithmic processes from common algorithms is that their behavior is often non-predictable and that frequently they produce patterns of thought and results that amaze even their own creators. Similarly in design, shape grammars, mathematical models, topological properties, genetic systems, mappings, and morphisms are algorithmic processes aimed at exploring uncommon, unpredictable, and uncharted formal properties and behaviors.\textsuperscript{4}

Two opposing human activities that are central to algorithmic composition as a mode of thought are invention and discovery. Invention is defined as the act of causing something to exist by the use of ingenuity or imagination: it is an artificial human creation. In contrast, discovery is the act of encountering, for the first time, something that already existed. Both invention and discovery are about the origin of ideas and their existence in the context of human understanding. These two intellectual mechanisms result from a logic which tends to argue whether the existence of certain ideas or processes is one of the following: either a human creation or simply a glimpse of an already existing universe regardless of the presence of humans. The most paradigmatic example of this polemic is that of geometry itself: the existence of geometry can be regarded as either a descriptive revelation of properties, measurements, and relationships of existing forms or as an arbitrary, postulate-based mental structure that exists only in the human mind. For instance, Euclidean geometry was developed originally to measure distances on the surface of earth and yet, in Euclidean geometry, Platonic primitive shapes, such as squares, circles, and triangles, do not exist \textit{per se} in nature though they represent idealized approximations of natural objects. Likewise, architecture can be regarded as either a simulation of the laws and structure of nature or as a world of fantasy and imagination.\textsuperscript{5} In any case, algorithms are encapsulations of processes or systems of
processes that allow one to leap and venture into the world of the unknown, whether natural or artificial. They are not the end product, but rather a vehicle for exploration.

Computation is a term that differs from, but is often confused with, computerization. While computation is the procedure of calculating, i.e., determining something by mathematical or logical methods, computerization is the act of entering, processing, or storing information in a computer or a computer system. Computerization is about automation, mechanization, digitization, and conversion. Generally, it involves the digitization of entities or processes that are preconceived, predetermined, and well defined. In contrast, computation is about the exploration of indeterminate, vague, unclear, and often ill-defined processes: because of its exploratory nature, computation aims at emulating or extending the human intellect. It is about rationalization, reasoning, logic, algorithm, deduction, induction, extrapolation, exploration and estimation. In its manifold implications, it involves problem solving, mental structures, cognition, simulation, and rule-based intelligence, to name but a few.

Digitization is the conversion of analog information into digital information. Computerization, by definition, involves digitization. This is not necessarily the case with computation. Because many mental processes can be analyzed, codified, systematized or even synthesized without the use of computers, computational methods do not have to involve digitization. Nonetheless, their implementation on a computer system allows explorations of complexities that extend the limits of human prediction. For instance, even though the work of Gaston Julia in the 1920s and, consequently, Benoit Mandelbrot in the 1970s were conceived and expressed in paper, they would have never been visualized, understood, and explored further without the use of computational graphics.

Computing is a term used to denote the act of making a mathematical calculation or a computation. Computing is often equated with computation since both employ the same methods. Essentially, both terms share the same meaning. Grammatically, the term computation involves the suffix “-tion” that denotes a state, condition, or quality of a procedure. Similarly, the term computing employs the suffix “-ing” that implies an action of implementing a procedure. While the two terms are linked through a state-action relationship, the noun “computation” implies a set of theories and methods, whereas the noun “computing” suggests an active investigation within the sphere of computation. In any case, these two terms are entirely different, distinguished, and set apart from the terms computerization or digitization.
The dominant mode of utilizing computers in architecture today is that of computerization: entities or processes that are already conceptualized in the designer’s mind are entered, manipulated, or stored in a computer system. In contrast, computation or computing, as a computer-based design tool, is generally limited. The problem with this situation is that designers do not take advantage of the computational power of the computer. Instead some venture into manipulations or criticisms of computer models as if they were products of computation. While research and development of software involves extensive computational techniques, mouse-based manipulations of three-dimensional computer models are not necessarily acts of computation. For instance, it appears, from the current discourse, that mouse-based manipulations of control points on non-uniform rational b-spline (NURBS)-based surfaces are considered by some theorists to be acts of computing. 9 While the mathematical concept and software implementation of NURBS as surfaces is a product of applied numerical computation, the rearrangement of their control points through commercial software is simply an affine transformation, i.e. a translation.

An alternative choice is being formulated that may escape these dialectically opposed strategies: algorithmic design. It involves the designation of software programs to generate space and form from the rule-based logic inherent in architectural programs, typologies, building code, and language itself. Instead of direct programming, the codification of design intention using scripting languages available in three-dimensional packages (i.e. Maya Embedded Language (MEL), SdMaxScript, and FormZ) can build consistency, structure, coherency, traceability, and intelligence into computerized three-dimensional form. By using scripting languages designers can go beyond the mouse, transcending the factory-set limitations of current three-dimensional software. Algorithmic design does not eradicate differences but incorporates both computational complexity and creative use of computers. For architects, algorithmic design enables the role of the designer to shift from architecture programming to programming architecture. Rather than investing in arrested conflicts, computational terms might be better exploited by this alternative choice. For the first time perhaps, architectural design might be aligned with neither formalism nor rationalism but with intelligent form and traceable creativity.

There are still some misconceptions about the role of the computer in the process of design. Design, like many other mental processes, at the information-processing level has nothing specifically neural about it. The functional equivalence between brains and computers does not imply any structural equivalence at an anatomical level (e.g.,
equivalence of neurons with circuits). Theories of information processes are not equivalent to theories of neural or electronic mechanisms for information processing.\textsuperscript{10} Even though, physically, computers may appear to be a set of mindless connections, at the information level they are the materialization of mathematical and syllogistic procedures.\textsuperscript{11}

The word “tool” is often used to describe the synergistic interaction of designers with computers. A tool is defined as an instrument used in the performance of an operation. The connotative notion of a tool implies control, power, dominance, skill, and artistry. A pen, for instance, is a device that allows one to perform or facilitate the manual or mechanical work of writing or drawing. The capabilities, potency, and limitations of a tool are known or estimated in advanced. This is not the case with computers performing inductive algorithmic computations. Neither is their capacity or potency understood, nor can their limitations be pre-estimated. Indeed, designers are frequently amazed by processes performed by algorithmic procedures, over which they have no control and of which they often have no prior knowledge.

Since the mid-1970s, beginning with shape grammars and computational geometry and continuing through topological properties and morphism, designers and theorists have been concerned with the use of algorithms as a mechanism for exploring formal compositions. These theories have attempted either to automate and enhance existing manual techniques or to explore new uncharted territories of formal behavior. Various methods have been employed in the search for new forms: formal analysis involves the investigation of the properties that describe an architectural subject. Composition, geometrical attributes, and morphological properties obeying Galilean, Newtonian, and, lately, molecular and organic principles are extracted from figural appearances of an object. In contrast, structural analysis deals with the derivation of the motivations and propensities which are implicit within form and which may be used to extract their generative processes. Morphism employs algorithmic processes for the interconnection between seemingly disparate entities and the evolution from one design to another.

Unlike computerization or digitization, the extraction of algorithmic processes is an act of high-level abstraction. It is often equated with rationalism, determinism, or formalism, but more importantly these resources are ultimately in the service of transcendency. Transcendency is the quality of lying beyond the ordinary range of perception. It is the quality of being above and beyond in response to timelessness and spacelessness. Algorithmic structures represent abstract patterns that are not necessarily associated with experience.
or perception. Furthermore, the observable outputs of algorithms should not be equated with the processes that created them. Marcos Novak makes a distinction between topology and curved surfaces. Topology, he points out, “means simply the study of those relations that remain invariant under transformations and deformations. A notion of continuity is indeed implied in this definition, but the continuity is abstract.”

Similarly, in Architectonics of Humanism Lionel March “seeks an order beyond appearances” as he attempts to “uncover the ‘many modes of numbering’” and “looks for the ‘warring and opposing elements’, which go to make an original microcosm echoing universal harmony.” Algorithmic processes result from events that are often neither observable nor predictable and seem to be highly intuitive. These events are made possible by abstraction and ingenuity. For instance, the discovery (or invention) of “hyperspace” resulted from an algorithmic inductive process of projections that map three-dimensional points into four-dimensional ones, yet both the projections and the results are neither predictable nor observable. In this sense, algorithmic processes become a vehicle for exploration that extends beyond the limits of perception.

When comparing contemporary practicing architects such as Thom Mayne, Frank Gehry, and Peter Eisenmann it is necessary to overlook many significant and distinguishing differences in order to identify at least one common theme: the use of the computer as an exploratory formal tool and the increasing dependency of their work on computational methods. The most paradigmatic examples of the last ten years invest in computationally generated parts and diagrams. Through computation, architecture transcends itself beyond the common and predictable. In contrast, computerization provokes Whorfian effects: through the use of commercial applications and the dependency on their design possibilities, the designer’s work is at risk of being dictated by the language-tools they use. By unknowingly converting to the constraints of a particular computer application’s style, one runs the risk of being associated not with the cutting-edge research, but with a mannerism of “hi-tech” style.

In Diagram Diaries, Peter Eisenman’s concept of an architectural diagram as an explanatory, analytical, generative, or representational device is directly linked to the principles of human understanding and interpretation. This human-centric approach is implicit within the sphere of subjective phenomena and personal interpretations. Within that realm, any logic that deals with the evaluation or production of form must be, by default, both understandable and open to interpretation. The problem with this approach is that it does not allow thoughts to transcend beyond the sphere of human understanding. In fact, while it praises and celebrates the uniqueness
and complexity of the human mind, it also becomes resistant to theories that point out the potential limitations of the human mind. In contrast, algorithmic form is not about perception or interpretation but rather about the process of exploration, codification, and extension of the human mind. Both the algorithmic input and the computer’s output are inseparable within a computational system of complementary sources. In this sense, a diagram becomes the embodiment of a process obtainable through a logic of mutual contributions: that of the human mind and that of the machine’s extendibility.

Similarly, Euclidean geometry was understood as an extension of human perception. The divinity of its nature can be ultimately linked to its ability to infer abstract concepts that appeal to the mind rather than the eye. Like religion, it was the revelation of an abstract system of relationships that transcended above and beyond the material objects it represented. Similarly, algorithmic form is an extension of human understanding. The mere existence of the term “unimaginable” can be linked to the ability of algorithms to surpass the sphere of human control and prediction. Like meta-structures, algorithmic forms are manifestations of inductive processes that describe, extend and often surpass the designer’s intellect.

There is often confusion about ownership of algorithmic forms. Intellectual property is defined as the ownership of ideas and control over the tangible or virtual representation of those ideas. Traditionally, designers maintain full intellectual property rights over their designs or manifestations thereof, based on the assumption that they own and control their ideas. This is not always the case with algorithmic forms. While the hints, clues, or suggestions for an algorithm may be intellectual property of the designer-programmer, the resulting tangible or virtual representations of those ideas is not necessarily under the control of their author. Algorithms employ induction, regression, randomness, recursion, cellular automata, probability, Markov chains, or quantum computation, to name a few, the outcomes of which are unknown, unpredictable, and unimaginable. If there is an intellectual root to these processes it must be sought in a world that extends beyond human understanding. Both the notions of “unknown” and “unimaginable” escape from human understanding since both negate two of the last resorts of human intellect, those of knowledge and imagination. In fact, as Novak points out, while the clause “if-then” is a syllogistic structure that leads on to new knowledge, the clause “if-then-else” involves the alternative “else” that may point to the opposite of knowledge, that is, to “that which does not follow from its roots, or, indeed, that whose roots can no
longer be traced, or have become irrelevant, or are unknown, or follow from principles outside previous understanding.”

A paradigm shift is defined as a gradual change in the collective way of thinking. It is the change of basic assumptions, values, goals, beliefs, expectations, theories, and knowledge. It is about transformation, transcendence, advancement, evolution, and transition. While paradigm shift is closely related to scientific advancements, its true effect is in the collective realization that a new theory or model requires understanding traditional concepts in new ways, rejects old assumptions, and replaces them with new. For T.S.Kuhn, scientific revolutions occur during those periods when at least two paradigms coexist, one traditional and at least one new. The paradigms are incommensurable, as are the concepts used to understand and explain basic facts and beliefs. The two live in different worlds. The movement from the old to a new paradigm is called a paradigm shift.

Traditionally, the dominant paradigm for discussing and producing architecture has been that of human intuition and ingenuity. For the first time perhaps, a paradigm shift is being formulated that outweighs previous ones. Algorithmic design employs methods and devices that have no precedent. If architecture is to embark into the alien world of algorithmic form, its design methods should also incorporate computational processes. If there is a form beyond comprehension it will lie within the algorithmic domain. While human intuition and ingenuity may be the starting point, the computational and combinatorial capabilities of computers must also be integrated.

Notes

1 The work in this area is as old as computer-aided design. An early attempt was MIT’s BUILD system, which would take a building program, indicating dimensions and connections for each space. The computer then arranged the spaces, thus solving the problem. This approach has since been used extensively for solving complex design problems that are related to arranging parameters in optimum ways. These approaches focus on the functionality of the design scheme and do not take into account aesthetic or artistic parameters. In areas such as the design of computer chips, nuclear plants, or hospitals, automatic spatial allocation plays a very important role, even today. See Dietz, A., Dwelling House, Construction Cambridge: MIT Press, 1974, and Yessios, C., Parent B., Brown, W. and Terzidis, C., “CKAAD Expert: A Computer and Knowledge Aided Architectural Design Expert”, in Design Theory 88, NSF Grantee Workshop on Design Theory and Methodology, Rensselaer Polytechnic Institute, Troy, NY, 1988, pp. 1–8.
2 Greg Lynn reveals that “because of the stigma and fear of releasing control of the design process to software, few architects have attempted to use the computer as a schematic, organizing and generative medium for design”. See Lynn, G., *Animate Form*, New York: Princeton Architectural Press, 1999, p. 19.

3 In response to this discrepancy, the ancient Greeks devised a “fair” method of acknowledgement of authorship. The Pythagorean theorem, the spiral of Archimedes, and Euclidean geometry are attempts to give proper credit to the authors regardless of the status of their subjects as inventions or discoveries.

4 The term algorithmic is often connected with complexity. While the objective or result of an algorithm may be complex, the strategy itself does not necessarily follow that complexity. For instance, chaos is the study of how simple systems can generate complicated behavior.

5 Perault, the architect of the peristyle of the Louvre, argued that architecture is a fantastic art of pure invention. He asserted that architecture really exists in the mind of the designer and that there is no connection to the natural world. In addition, architecture as an imaginative art obeys its own rules which are internal and personal to each designer, and that is why most creators are vaguely aware of the rules of nature and yet produce excellent pieces of art. A similar point was also argued by Giovanni Battista Vico. In his work *The New Science* (1744), Vico argued that one can know only by imagining. The twisting of language and meaning can lead one to discover new worlds of fantasy. He argued that one can know only what one makes. Only God can understand nature, because it is his creation. Humans, on the other hand, can understand civilization, because they made it. “The world of civil society has certainly been made by men, and its principles are therefore to be found within the modification of our own human mind.”

6 In its colloquial sense, computerization refers to the process of furnishing with a computer or a computer system.


9 See Cuff, D., “One Educator’s Thoughts on Design Software’s Profound Effects on Design Thinking and Teaching”, *Architectural Record*, vol. 9, 2001, pp. 200–206. In this article, Cuff considers that computing is “one of the most important transformations of the contemporary profession” and that today “computing has become a populist skill”.


11 Lynn, G. in *Animate Form*, op. cit. p. 19, describes machine intelligence “as that of mindless connections.”

12 Novak continues:
A cube is not less topological than a blob. However, when working algorithmically, what remains invariant is the algorithm, so that a new notion of topology, ‘variable topology’ is introduced. While the variations in the space of the parameters and control structures that implement the algorithm may be continuous, the product of the algorithm may be to show tears and discontinuities and ever fracture into a cloud of particles or an explosion of shards.


15 The prefix meta- indicates one level of description higher. If X is some concept, then meta-X is data about, or processes operating on, X. For example, meta-syntax is a syntax for specifying syntax, meta-language is a language used to discuss language, meta-data is data about data, and metareasoning is reasoning about reasoning (definition taken from the Free On-line Dictionary of Computing).

16 Sir Karl Popper argued that the world as a whole consists of three interconnected worlds. World One is the world of physical objects and their properties-with their energies, forces, and motions; World Two is the subjective world of states of consciousness, or of mental states-with intentions, feelings, thoughts, dreams, memories, and so on, in individual minds. World Three is the world of objective contents of thought, especially of scientific and poetic thoughts and of works of art. World Three is a human product, a human creation, which creates in turn theoretical systems with their own domains of autonomy. See Popper, Karl R., The Logic of Scientific Discovery, New York: Harper & Row, 1968.


19 Peter Eisenman referred to the idea of an electronic paradigm shift in architecture in 1992. He wrote:

During the fifty years since the Second World War, a paradigm shift has taken place that should have profoundly affected architecture: this was the shift from the mechanical paradigm to the electronic one. This change can be simply understood by comparing the impact of the role of the human subject on such
primary modes of reproduction as the photograph and the fax; the photograph within the mechanical paradigm, the fax within the electronic one.