pages: 1 -63

MORPHOGENESIS AND MORPHOGENETIC IT TOOLS IN ARCHITECTURAL DESIGN

Krystyna Januszkiewicz [*], Natalia Paszkowska-Kaczmarek [**]

[*] West Pomeranian University of Technology in Szczecin, Faculty of Architecture, ul. Żołnierska 50, 71-210 Szczecin, Poland
[**] Urbicon Spółka z o.o., 71-303 Szczecin, Poland
[*] E-mail: kjanuszkiewicz@zut.edu.pl ORCID: 0000-0001-6880-0862
[**] E-mail: npaszkow@gmail.com

DOI: 10.24427/aea-2025-vol17-01

Abstract
The article focuses on morphogenetic design using IT tools and their application in architectural design. The generative and creative potential of IT media has opened a new dimension in architectural design, especially in architecture aimed at imitating the works of Nature, its form-forming processes, behaviors, or ecosystems. At the turn of the 20th and 21st centuries, new applications of generative tools for architecture were made available, based on systems and mechanisms occurring in Nature, bringing designers closer to creating architecture consistent with the natural environment, not only in terms of visuality, but also in terms of the functioning of the building based on the model of a living organism. The biomimetic approach to designing architectural objects and their complexes is currently moving to a higher level of material, structural, and performative integration. It presents what morphogenesis is and its role in the creation of a new organism, as well as other emergent phenomena occurring in Nature. The instrumentalization of these processes in the IT space is considered, as well as the application of design tools based on these processes. The tools that imitate form-forming processes occurring in Nature are presented: Cellular Automata, L-systems, evolutionary and genetic algorithms, as well as mathematical object-tools (a specific type of sets) such as: Fractals, Voronoi Diagrams, Shape Grammars, which can describe the geometric results of natural form-forming processes.

Keywords: architecture; design; IT technologies; morphogenesis; morphogenetic tools

INTRODUCTION

Research into natural structures, conducted in various fields of science, has provided the basis for defining the geometry of form and its behaviour. The developed mathematical models and computer instrumentalisation of the processes of evolution, morphogenesis and emergence, together with the proposed methods and techniques, now make it possible to apply these patterns in architectural and structural design, as well as in materials design. It is a key concept that is equally important for theory and computer-aided design methods. The application of generative morphogenetic tools at the turn of the 20th and 21st centuries brought about a significant change in the use of computers' ability to instrumentalise complex formative processes and material behaviours. Today, we are looking for a different model of relations with nature and the universe. Global climate and anthropogenic changes, along with the associated threats, are forcing a departure from the capitalist relationship between Man and Nature. One alternative is morphogenetic design, as the logic of morphogenesis and emergence is not limited to the methodology of architectural design. These issues extend to broader areas of the built environment.

The aim of this article is to present how imitations of natural formative processes will change the conceptual side of architecture and influence its material practice. The basics of morphogenetic design and morphogenesis are presented in the light of current research. In describing the principles of natural morphogenesis, attention is drawn to how they have been transferred into an integral computational process coupled with a CAD system. In the 21st century, new applications of morphogenetic tools for architecture have become available, opening up new design methods. The methodological approach adopted here combines descriptive and analytical techniques with logical reasoning in order to determine, through comparative analysis, the changes and effects of these changes in architectural design.

1. MORPHOGENETIC DESIGN AND MORPHOGENESIS

Morphogenetic design is a mimetic process that creates artificial objects made of inanimate matter by giving them the characteristics and properties of living organisms. It uses generative design tools based on biological models derived from mathematical models of evolution and the biological sciences of evolutionary development. They combine the processes of embryonic growth and evolutionary development of species, offering designers tools to mimic morphogenetic formative processes. Evolutionary calculations allow the species pattern and process, as well as form and behaviour, to be linked to spatial and cultural parameters. Architecture and the built environment designed using morphogenetic tools are intended to be a remedy for rebuilding our relationship with Nature.

In the natural world, every form emerges from a process. It is a process that produces, develops and maintains biological and non-biological forms or structures, and which consists of a complex series of exchanges between an organism and its environment. Morphogenetic design tools attempt to mimic these processes. They are therefore generative tools that originate in the exact sciences and are used to produce 2D and 3D patterns and forms with complex geometry. They are mathematical models describing states or phenomena that occur in the natural world, although they may only be a mathematical operation. Their name (Latin *generare* – to give birth) refers to methods of using mathematical symbols and relationships to generate states of increasing complexity according to established rules. In computational environments, generative morphogenetic design tools can be divided into two groups:

- tools imitating formative processes occurring in nature: cellular automata, L-systems, evolutionary and genetic algorithms,
- mathematical tools-objects (a specific type of sets) such as: Fractals, Voronoi diagrams, shape grammars, which can describe the geometric results of natural form formation processes.

In architectural design, the tools of both groups often complement each other. They include both digital techniques and computational models, which are used to derive and transform forms, raising them to a higher level of formal and structural complexity and environmental efficiency.

1.1. Morphogenesis as a biological process

Morphogenesis (from the Greek *morphê* shape and genesis creation, literally „the generation of form") is the biological process that causes a cell, tissue or organism to develop its shape. It is one of three fundamental aspects of developmental biology along with the control of tissue growth and patterning of cellular differentiation. An important aspect of natural morphogenesis is that the processes of formation and materialisation are always inseparable and inextricably linked.

Originally, the term morphogenesis was used in biology, and the first recorded instances date back to the second half of the 19th century, when its equivalents were morphogenie (German, 1874) and morphogénie (French, 1862). In the 20th century, the term morphogenesis was also adopted in geology in relation to geomorphology, the science of the origin of landscape forms, or the processes that led to the formation of these forms [W. Stankowski 2019]. In biology, the word 'morphogenesis' is often used in a broad sense to refer to many aspects of development, but when used strictly, it should mean the formation of cells and tissues into specific shapes (Fig. 1). That is, the developmental processes that determine the shape of the embryo in successive stages of development and ultimately the shape of the adult organism. Incidentally, in biological sciences, the study of forms through their categorisation or morphology was the first tool of zoology, preceding the theory of evolution. Today, morphology has transcended its historical limitations to become morphogenesis, and research focuses on the forces that generate living forms and how their environments came into being.

Morphogenesis is one of several processes characteristic of living organisms. Natural morphogenesis, as a process of growth and evolutionary development, generates systems that acquire complex articulation and specific form (gestalt) and performative abilities through the interaction of internal material characteristics and external stimuli, i.e. forces and influences from the environment. In natural morphogenesis, therefore, formation and materialisation are always intrinsically and inextricably linked. Furthermore, in biology, the term 'morphogenesis' is used to refer to structural changes in tissues as the embryo develops, or to the basic mechanisms responsible for structural changes.

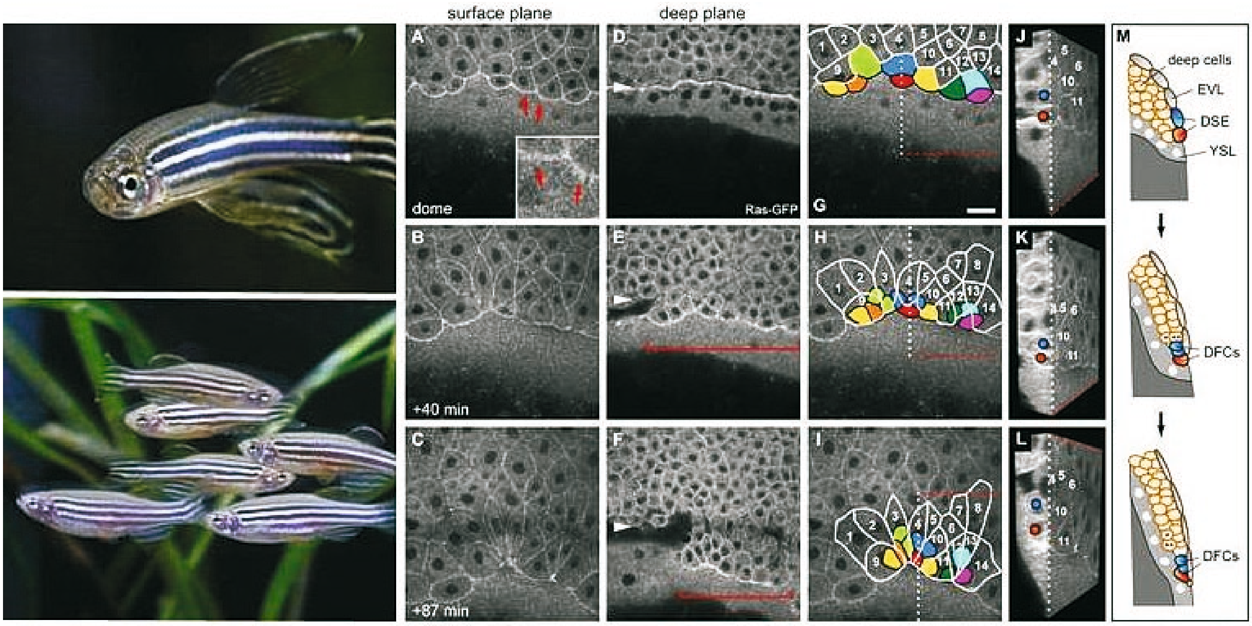


Fig. 1. Zebrafish (danio rerio), course of dorsal fin morphogenesis on a micro scale; source: Oteíza et al. 2008

Morphogenesis processes can be interesting and inspiring for architects, even though literally transferring biological structures or processes to an architectural design is usually unfeasible, however much it would be significant or desirable. In addition to morphogenesis, there are also processes such as growth, repair, adaptation and ageing. Transferring knowledge about these processes to architectural design can also be productive, especially with regard to architectural structures with dynamic capabilities.

A better understanding of biological morphogenesis may be useful in architectural design because:

- architectural design aims to solve problems that have often already been solved by Nature; architectural design increasingly strives to incorporate concepts and techniques such as development or adaptation, which find their counterparts in Nature;
- architecture and biology share a common language, as both attempt to model processes such as growth and adaptation (or morphogenesis).

1.2. Morphogenesis as a generative geometric and computational process

Research into natural structures, conducted in various fields of science, has provided the basis for defining the geometry of form and its behaviour. The developed mathematical models and computer instrumentalisation of the processes of evolution, morphogenesis and emergence, together with the proposed methods and techniques, now make it possible to apply these patterns in architectural and structural design. This is a key concept that is equally important for both theory and computer-aided design methods [K. Januszkiewicz 2010, p. 160]. The development of tools that generate form, its shape and the interrelationships between its parts in relation to environmental conditions has led to conceptual changes in the way we think about buildings, their behaviour and material structure. Today, computer processing power allows us to visualise and control the processes of form formation and behaviour, as well as material self-organisation during the design process.

At the beginning of the 19th century, in the context of botanical studies, the poet and writer Johann W. von Goethe (1749–1832) defined morphology as the study of form and presented this in his work entitled *Die Absicht ist eingeleitet*, written in 1807, and published in 1817. He combined the theory of characters Gestalt, i.e. the construction of form with the process of formation (Bildung), which constantly succumbs to form [E. Trunz 1960, pp. 54–56]. In this way, he brought biological sciences closer to visual arts and architecture. The integral processes developed in biological morphogenesis concerning the formation and development of material form (gestalt) are particularly important because architecture, as a material practice, continues to rely mainly on design approaches characterised by a hierarchical relationship that prioritises the definition and creation of form over its subsequent materialisation. This suggests that the hidden potential of a given technology can be developed from an alternative approach to design, one that derives from morphological complexity and performative capabilities without distinguishing between the processes of form creation and materialisation.

In architecture, morphogenesis (computational morphogenesis or digital morphogenesis) is understood as a group of methods that use digital media not as tools for representation or visualisation, but as generative tools for deriving form and its transformation, often in an effort to express environmental processes in built form.

In the age of information technology, the discourse on morphogenesis in architecture also includes concepts such as emergence, self-organisation, and form invention (*form-finding*). Among the advantages of biology-inspired forms, their proponents cite the potential structural benefits of redundancy and differentiation, and the ability to support multiple functions simultaneously [S. Roudavski 2009, p. 348].

1.3. What are artificial generative morphogenetic systems?

Generative systems are formalised mechanisms capable of producing alternative design solutions. These systems enable the creation of complexity at levels higher than their initial specification, as interacting elements of a given complexity generate aggregates with significantly greater behavioural or structural complexity. For example, Generative Design System software utilises this principle and, under the supervision of an architect, enables the generation of complex forms and patterns from simple specifications [J.P. McCormack et al. 2005]. However, it should be noted that the use of a generative design system is only possible after the project's intentions and objectives have been defined, which is necessary in order to define rules, relationships and algorithms.

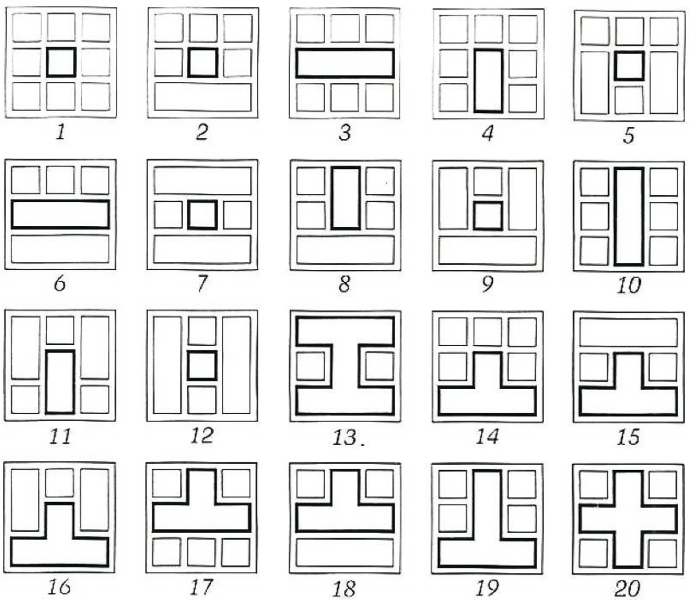


Fig. 2. William J. Michell, Shape Grammars Based on the Writings and Projects of Andrea Palladio; source: Stiny et al. 1978

Generative morphogenetic systems are artificial systems that mimic the formative processes and behaviours found in natural (biological) systems. Morphogenetic generative design is the use of imitations of these processes, which are incorporated into mathematical and computational models managed by algorithms, sets of rules and principles, in order to obtain design solutions. Morphogenetic design replicates them. Morphogenetic design mimics the evolutionary approach to the natural world to provide thousands of solutions to a single architectural and engineering problem. The rules for generative systems can be defined in various ways, e.g. by verbal grammars, diagrams, geometric transformations or command scenarios. Generative systems have varying degrees of control, from automated to manual, performed step by step [K. Januszkiewicz 2012b, p. 44]. Due to the possibilities of representing design solutions, generative systems can be divided into three broad groups:

- analogue systems, where the properties of the system are used to represent the properties of the designed objects. Such analogue systems include, for example, mechanical and electrical systems;
- iconic systems helpful in creating alternative design solutions by assigning
- operations such as adding, subtracting, transforming and moving those parts that are saved;
- symbolic systems use symbols such as words, numbers and mathematical formulas to represent possible solutions at the output [Y.E. Kalay & W.J. Mitchell 2004, s. 326].

The idea of using generative systems in design has its roots in the past. Design patterns and principles have been implemented in the history of architecture and art for many generations. The characteristics of such systems can be found in many historical examples, including painting, architecture, and design methods. For example, an analysis of Greek and Roman architecture shows the consistency of a design that was developed using logical design principles. Palladio, a famous Renaissance architect, developed a design process based on such logical design principles throughout his architectural work. In the 1970s, Stiny and Mitchell were able to extract a set of such rules of form and grammar from Palladio's writings and designs [G. Stiny et al. 1978, pp. 5–18]. These grammars were able to create many variations of Palladio's designs (Fig. 2).

William Mitchell, known for his research on evolution, compiled a classification of generative systems from Aristotle to Richard S. Lull (1867–1957). He demonstrated that these systems also played an important role in philosophy, literature and music. In architecture, Mitchell's classification refers to Leonardo da Vinci (1452–1519) and Jean-Nicolas-Louis Durand (1760–1834). In his studies conducted between 1803 and 1805 and published under the title *Précis des Lecons d'Architecture données à l'ecole polytechnique*, Durand proposed an innovative method of drawing projections and elevations by repeating and translating structural parts such as walls, facades, etc. These elements were to form a system in which new wholes could be designed. The aim was to design and build efficiently and quickly in Europe and the overseas colonies in the styles fashionable in the 19th century [K. Januszkiewicz 2012a, p. 48]. This approach to design, when applied in practice, would facilitate the preparation of two-dimensional technical drawings of buildings, which usually feature similar spatial solutions and are rich in historical decorative details.

It should be mentioned that systems thinking had already appeared earlier in other scientific disciplines. A system was treated as an object in which a set or groups of elements could be distinguished, interconnected in arrangements forming certain superordinate wholes. Nevertheless, systems thinking long stood in opposition to the reductionist concept that had characterised the approach of scientists and engineers for centuries. It was something new, something that opened up new areas of research. However, it was not until the 1930s that the need to create a unified approach to the system became apparent, especially in the natural sciences and later in the exact sciences. After World War II, Buckminster Fuller (1895–1983) in the USA, and the creator of systems theory Ludwig von Bertalanffy (1901–1972) in Europe, were active promoters of this then new way of seeing the world and solving complex problems.

The methodology of systems design entered architecture in the late 1960s and was mainly concerned with solving functional problems and spatial denotation. The basic principles of design were determined according to the definition of the problem, the description of the goal and the design task in a specific area of architectural creativity. Logical models, algorithms and mathematical models were used, as were the computer-aided design, simulation, optimisation and multi-variant design solution techniques available at the time. This was a consequence of the use of mathematical and logical apparatus within the scope of the then possible usefulness of computers in design [K. Januszkiewicz 2012a, pp. 54–65]. These new aspects of architectural form creation were explored in Poland by Adam M. Szymski. Drawing on his knowledge of human sciences, he conducted comparative analyses of the creative process and systemic design processes, laying the foundations for computer-aided design methodology [A.M. Szymski 1997].

Between 1968 and 1995, John H. Frazier and his research team developed the first design models using generative and evolutionary tools. The corresponding generating systems were based on a common strategy called 'seed generation' (for a design) or initial configuration. Each model defined a set of tasks to be performed by the design team, and in each case, one of the tasks required generative or evolutionary design tools [J.H. Frazer 1995]. By introducing minor modifications to the transformation process or only to the shape of the grain, it was possible to generate alternative designs. These models corresponded to the capabilities of electronic equipment and the state of knowledge in the field of computer science at that time [K. Januszkiewicz & N. Paszkowska-Kaczmarek 2023].

Currently, in order to study the impact of various factors on form, architects are turning to generative IT systems. They borrow them from other disciplines and use them to design buildings and materials. The most popular ones are: Cellular Automata, Voronoi diagrams, L-Systems, fractals, shape grammars, and evolutionary and genetic algorithms. Contemporary generative IT systems can also be divided according to the type of imitated and simulated processes: (i) formative processes occurring in biological creations, (ii) shape or pattern generation, (iii) material behaviour simulations, (iv) physical phenomenon simulations. The use of generative tools in design requires architects to take a different approach to the creative process than before, as these are primarily computational tools. This is a significant change, as architectural theory and practice have so far focused primarily on form rather than on the process that gives rise to form. In the 21st century, designers are learning from Nature how to use energy and materials sparingly, finding effective engineering solutions and structural patterns for new building materials in their creations. They are also learning how natural and built environments can best interact with each other.

1.4. The need for research on morphogenesis and morphogenetic tools

In striving for greater environmental, functional, material and energy efficiency in building forms than ever before, imitating Nature's creations is one of the options available today. Morphogenesis in architectural aspects should be developed in such a way as to meet the following requirements and expectations:

- computational design tools need to be developed and integrated into CAD/CAM/CAE systems in order to demonstrate research on the application of abstracted biological principles in the creation of structures that respond to changes in the environment;
- it is expected that non-unified complex structures will be increasingly designed in architecture in response to growing interest in parametric modelling, CNC fabrication and personalisation;
- it is necessary to be able to design structures that have no direct precedents in architecture. However, such precedents exist in nature, where structurally complex living organisms have been adapting to their environment for millions of years.

The instrumentalisation of morphogenesis and other formative processes, as well as the environmental behaviour of natural structures, will change the approach to the design of buildings. It will contribute to an understanding of form, material and structure not as separate elements, but rather as complex interrelationships that are embedded in and explored by computational integral design processes [E. Trunz 1960, p. 53].

2. EFFORTS TO IMITATE BIOLOGICAL PROCESSES AND BEHAVIOURS IN COMPUTATIONAL ENVIRONMENTS

A biomimetic approach to architectural design is not fully possible without the exchange of ideas and techniques between architecture and disciplines such as biology, physics, chemistry and mathematics. The focus is mainly on the natural processes of formation and adaptation that occur in nature, their instrumentalisation through mathematical models and computational techniques, their simulations and digital visualisations. Mathematics continues to provide operational tools for science to create mathematical models that describe simple and complex real-world phenomena. Such modelling is used to understand a given process by replacing it with a simplified system that reflects only selected features of the process. The mathematical description of the model is presented here in the form of a system of algebraic or differential equations. The processes under study are described by mathematical models with complex parameters, and the variables contained therein are subject to changes both in time and space [J. Gutenbaum 2003]. The digitisation of computational processes has made it possible to describe many complex, often non-linear phenomena of reality using mathematical models. Computer graphics has become helpful in visualising the course of modelled processes [A. Menges 2006, p. 53].

2.1. Emergence

In science, the term *emergence* (Latin *emergo* – to emerge, to rise) refers to the production of forms and behaviours by natural systems (ecosystems) that have irreducible complexity, as well as to the mathematical approach necessary for modelling processes in computational environments. Emergence, like self-organisation, is a completely different feature of system behaviour [S.A. Kauffman 1996].

Dynamic systems in nature, living systems and physical systems, including climate and geological forms, exhibit diverse organisational and behavioural characteristics that are crucial for research into emergence. These are ecosystems that are diverse in terms of components, relationships and information. These relationships are complex and operate in various hierarchical systems, and the resulting effects tend to take time to emerge. There are many definitions of evolutionary and developmental processes that unfold over time.

The definition presented by Tom de Wolf and Tom Holvoet is widely cited. They proposed the following understanding of emergence: A system exhibits emergent behaviour when there are coherent elements that can emerge (property, behaviour, structure, etc.) at the macro level and which arise dynamically as a result of interactions between parts at the micro level. Such elements enter into new relationships with respect to individual parts of the system *[T.D.* de Wolf & T. Holvoet 2005, pp. 3–4]. This reveals a tendency for systems to self-organise and strive for ever greater complexity. Evolutionary development arises from the dynamics of systems. Any physically occurring system that can be described using mathematical tools or heuristic principles is perceived as a dynamic system. Dynamic systems theory classifies systems based on mathematical tools rather than the visible form of the system.

Emergent transformations of individual biological forms are not detached from their structure and material. Natural structures contain complex material hierarchies, and it is in these hierarchies that the efficiency of emergent processes lies. Form (its geometry), structure and material interact with each other, and the behaviour of these three interacting components cannot be predicted solely on the basis of an analysis of any one of them [M. Hensel et al. 2010]. However, each higher level of ownership can be described as a consequence of lower levels of ownership. The systems from which form emerges and the systems within complex individual forms are maintained as a result of the flow of energy and information through the system. The flow pattern has constants that are adjusted to maintain equilibrium through "feedback" with the environment. Natural evolution is not a single system but is divided into many co-evolving systems with partial autonomy and occurs through interaction. A newly formed form (as a whole) can be a component of a system emerging at a higher level – then what is a system for one process can be an environment for another [M. Weinstock 2004, p. 13].

Although emergence is a concept strongly associated with evolutionary biology, it also occurs in other disciplines such as complexity theory, cybernetics, and, more generally, systems theory and artificial intelligence research. Nevertheless, it is a term that is increasingly used in architectural discourse. Emergence is important for architectural design and calls for a serious revision of the way designs are created. It means the emergence of qualitatively new properties, forms and behaviours resulting from the interaction between simpler elements in a dynamic process.

2.2. Evolution

The term evolution (Latin *evolvere, evolutio* – development, growth) refers to the process of change over time. Ecosystems adapt and evolve at different levels and at different rates. Adaptation, i.e. the process of genetic change in a population, and evolution allow organisms and ecosystems to survive in their environment, which is unique locally and cyclically variable with dynamic behaviour. Every living form emerges from two strongly interrelated processes, operating in maximally diverse periods of time: (i) from the rapid process of embryological development from a single cell to an adult form; (ii) the long, slow process of evolution of different species of forms over many generations. Biological evolution means changes in the characteristics of entire groups of organisms occurring in successive generations. [W. Ullrich 1973].

The change in the characteristics of successive generations occurs as a result of the elimination of some individuals (genotypes) from the current population through natural or artificial selection. Together with new mutations, this continuously affects the current gene pool of the population, and thus shapes its average phenotype at any given moment. Complex forms and systems of nature arise as a result of evolutionary processes. In addition, living forms grow, and growth is a complex process in which the contribution of the genotype is variable and the influence of the environment and phenotypic dependencies are different. In nature, the genotype encompasses the genetic constitution of an individual, while the phenotype is a product of the interaction between the genotype and the environment.

In 1859, Charles Darwin published *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*, in which he presented his theory of evolution. The theory of evolution revolutionised the natural sciences, then philosophy and the social sciences, and eventually encompassed all fields of science. In the 1930s and 1940s, a synthetic theory of evolution was developed as a result of combining Darwin's theory of evolution with genetics, which is still being developed and supplemented with new research findings.

The perfection and diversity of natural forms is therefore the result of constant evolutionary experimentation. Through profligate prototyping and ruthless rejection of failed experiments, Nature has evolved into a rich biodiversity of interdependent plant and animal species that are in metabolic equilibrium with their environment. The analogy to evolutionary architecture should not be treated merely as an implication of form development through natural selection. Other aspects of evolution, such as the tendency towards self-organisation, are equally or even more significant [J. Frazer 1995]. *Incidentally, in nature, the genotype encompasses the genetic constitution of an individual, while the phenotype is the product of interactions between the genotype and the environment. The emerging properties and capabilities of natural forms result from generative processes that act on successive versions of the genome. This genome is compact data that is transformed into biomass with a complex structural structure. The compelling goal is to instrumentalise the natural process of evolution and growth, model the basic characteristics of emergence, and then combine them within a computational framework.*

Evolutionary processes in biological systems have provided the initial concepts for evolutionary programming and evolutionary algorithms. The foundations of evolutionary programming were developed by Lawrence Fogel (1928–2007) and further developed by his son David B. Fogel. In evolutionary computation, an initial set of candidate solutions is generated and iteratively updated. Each new generation is produced by stochastically removing less desirable solutions and introducing small random changes. In biological terms, the population of solutions is subjected to natural selection (or artificial selection) and mutation. As a result, the population will gradually evolve towards increased efficiency, in this case the selected algorithm matching function. The goal of evolutionary programming was to develop artificial intelligence in relation to the development of the ability to predict changes in the environment. The environment was described by a sequence of symbols, and the evolutionary algorithm was to create a new symbol. The initial symbol maximised the function of the task to be solved and assessed the accuracy of its execution [T. Bäck & D.B. Fogel 1966].

Evolutionary programming is of significant importance in the design of architectural objects and complexes which, according to futuristic mimetic concepts, should evolve alongside the natural environment in the Anthropocene era. It is expected that artificial intelligence components embedded in buildings will create a network of reactive interconnections and bring about evolutionary changes in material and operational structures.

2.3. Form and behaviour emerge from the process

Biological forms and their behaviour emerge from a process that produces, develops and maintains the form and structure of biological organisms (and non-biological things), and this process consists of a series of complex exchanges between the organism and its environment. This is because an organism has the ability to maintain continuity and integrity by changing aspects of its behaviour. Form and behaviour are closely related. The processes of self-generation of form, as well as form itself, are described by basic patterns. Geometry plays a local and global role here, and is also related to the dynamic pattern and the pattern of form in self-organising morphogenesis. The form of an organism influences its behaviour in the environment, and specific behaviour will produce different results in different environments. Behaviour is non-linear and context-dependent [M. Weinstock 2004, p. 14].

Organisms can be viewed as systems that develop complex forms and patterns of behaviour through the interaction of their components in space and time. The dynamics of biological form development, due to growth and form along with morphogenesis, has taken centre stage, supplanting Darwin's theory of evolution. The theory of morphogenesis, the formation of forms that develop in time and space, is inextricably linked to mathematical information theory, physics and chemistry, as well as organisation and geometry. They become a coherent pattern arranged by mathematical concepts and the economics of technology and industry [M. Hensel et al. 2010].

Incidentally, the connection between biology and mathematics was initiated in the first decade of the 20th century, particularly in the works of Alfred N. Whitehead (1861–1947) and D'Arcy W. Thompson (1860–1948). Thompson, a zoologist and mathematician, viewed the matter of living forms as a diagram of the forces that act within them to sustain life [D.W.Thompson 1961]. His observations of homologies between skulls, pelvises, and plants of different species suggested a new method of analysis, namely the mathematisation of biology. Measurements of morphological features are specific to a given species, and sometimes to individuals belonging to a single species. Hence, these measurements vary, but there are layers that do not differ at all, and these constitute homologies.

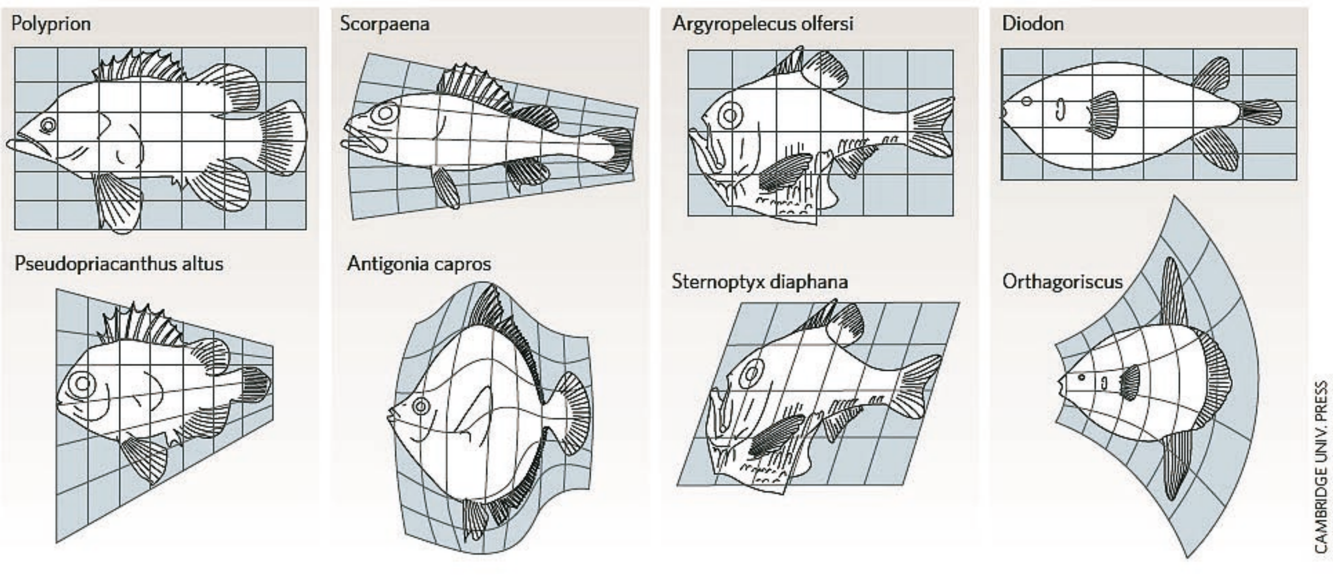


Fig. 3. Darcy W. Thompson, Homology of Forms of Related Organisms, 1910; source: *A ton for Thompson's* tome 2017

Homology, although it has two distinct meanings, has related meanings. In biology, it refers to organs or bodies that have the same evolutionary origin but different functions, and in animals, it also refers to behaviours resulting from inheritance from a common ancestor [W. Ullrich 1973]. In mathematics, on the other hand, it is the classification of geometric figures according to their properties. Hence, homology can be described by mathematical data, mapping points in a three-dimensional coordinate system in space, by dimensions, angles and radii of curvature. Comparisons of related forms made by D'Arcy Thompsom show recognisable deformations when moving from one form to another (Fig. 3). These forms are related when one can be deformed into another by transformations of the Cartesian coordinate system. Comparative analyses reveal that it does not matter what is omitted in each individual description of a form, nor does the accuracy of measurement matter; what matters is only that there is a morphogenetic tendency between forms [K. Januszkiewicz 2013, p. 44]. However, Alfred N. Whitehead (1861–1947), a mathematician and physicist, proved that the morphogenetic process is more important than matter, as it constitutes the living world, and Nature consists of interacting patterns of activity. Organisms are a collection of compounds that support each other in order to modify their own behaviour in anticipation of changes in activity patterns and everything around them. Anticipation and reaction drive the dynamics of life [M. Weinstock 2004, p. 13].

The combination of these hypotheses is important because it makes us realise that form emerges from process. It is a process that produces, develops and sustains the form or structure of a biological organism (and non-biological things), and which consists of a complex series of exchanges between the organism and its environment. Furthermore, the organism has the ability to maintain continuity and integration through the changing aspects of its behaviour. Forms are related by morphogenetic tendencies, which suggests that they are the same if not all of these characteristics are amenable to mathematical modelling [M.Weinstock 2004, pp. 14–15].

This research is particularly relevant for designers, especially when architecture and engineering engage in generative design processes in physical and computational environments in search of new conceptual patterns for a built environment that is sensitive to the effects of climate change.

2.4. Geometry and morphogenesis

The geometry of biological forms, just like computational forms, is not merely a description of a fully developed form. It is a set of rules defining boundaries and constraints that act locally as principles of organisation for self-organisation during morphogenesis. Geometric patterns and feedback loops are as important in morphogenesis models as they are in cybernetic models or other dynamic systems [M. Weinstock 2004, p. 14–15].

Research conducted by Alan N. Turing (1912–1954) on plant form development led to the formulation of a general theory of morphogenesis of cylindrical lattice structures. These structures are formed more locally than globally, node by node, and then modified by growth. To build a mathematical model of this process, one needs information about the global geometry, the characteristics of the cylinder, and a set of local rules for the lattice nodes. For many years, Turing was interested in the morphogenesis of daisies and fir cones with polygonal symmetrical structures such as those found in starfish, the modulation found in Fibonacci number sequences in the arrangement of leaves on plant stems, and the formation of patterns such as dots or stripes. His simple early models of morphogenesis demonstrate the breakdown of symmetry and homogeneity, or the emergence of a pattern from an originally homogeneous mixture of two substances. Mathematical equations describe these non-linear changes in the concentration of two chemicals (morphogens) over time, as well as how these chemicals react and how they diffuse [A.N. Turing 1952]. This leads to the hypothesis that the generation of geometric patterns begins with a smooth layer of cells in which information about buds, skin spots and branches appears during development. Chemical substances accumulate until the final density is reached, at which point morphogenesis takes effect to generate organs. The diffusion reaction model is still of interest to mathematical biology, where research focuses most on linking pattern dynamics to form. The model used by Alan M. Turing represented a single surface or flat plate of cells [S.A. Kauffman 1993, p. 566–577].

Currently, research on morphogenesis using computational models in molecular stereodynamics already covers processes involving curved plates [Ch.J. Marzec 1999]. Geometry is inherent in these process models, as it takes into account the 'units' that enter into dynamic relationships with each other and provides information about global geometry. Fred W. Cummings (born 1923) even proved that the interaction and diffusion of morphogens in cellular layers [L.G. Harrison & M. Kolář 1988] produce a Gaussian effect and determine the curvature of individual membranes or layers [F.W. Cummings 1989]. This means that changes in the curvature of the membrane in one place will cause opposite changes in curvature elsewhere. As a result of the development of computational models, this approach is currently being expanded to include the mathematics of curvilinear coordinate grids and uses fluid dynamics to simulate morphogenetic asymmetric organs and their branches [F.W. Cummings 1990]. However, studies of flat, folded cell plates reveal that they are the basis of morphogenesis and asexual reproduction. Computational models of morphogenetic processes can be successfully adapted in architectural research, and the self-organisation of material systems has been proven in physical form-finding processes [K. Januszkiewicz 2010, p. 126].

2.5. Pattern dynamics, diversity and integration

The concept of linking geometric patterns and forms during morphogenesis requires feedback, which is essential for maintaining forms in the living environment. In modelling the geometric pattern of form, feedback occurs in two loops: from form to pattern and from pattern to form. Structured formations with a biochemical pattern then cause morphogenetic "shifts" and, as a result, transformations in geometry. Such a change in geometry disrupts the pattern and a new pattern emerges, which initiates new morphogenetic shifts. This process continues until the distribution of morphogens and remains in equilibrium with the geometry of the evolving form [M. Weinstock 2004, p. 13].

Feedback loops, from pattern to form and from form to pattern, are a mathematical model of morphogenesis [A.V. Spirov 1993, p. 497], a model of a dynamic process from which form emerges. Incidentally, according to systems theory, concepts and principles of organisation in natural systems depend on the domains specific to each individual system, and contemporary research focuses on 'complex adaptive systems' that are self-regulating. What they have in common is the study of organisation, i.e. structure and function. Complexity theory formalises the mathematical construction of this process in systems from which complexity emerges. It focuses on the effects produced by the collective behaviour of many simple units that interact with each other, such as atoms, molecules or cells [W. Weaver 1948, p. 36].

This complexity is heterogeneous, as there are many different parts that have multiple connections, yet these parts behave differently, even though they are not independent. Complexity increases as the diversity and interdependence (connection) of parts increases. The process of increasing diversity is called differentiation, and the process of increasing the number or strength of connections is called integration. The evolutionary processes of differentiation and integration interact on many 'scales,' from the formation and structure of individual organisms to species and ecosystems [K. Januszkiewicz 2010, p. 126].

Systems theory and cybernetics today share a common conceptual basis, as evidenced by the frequent use of terms such as 'complexity science' and 'complex adaptive systems'. These terms also appear in extensive literature on thermodynamics, research on artificial intelligence, neural networks and dynamic systems. In mathematics, there is also a common approach to computational modelling and simulation. It is axiomatic in contemporary cybernetics that systems of increasing complexity are recognised, which in the natural evolution of systems show increasing complexity, from cells to multicellular organisms, from social systems to culture.

Research into the dynamics of geometric patterns with diverse components is important for architecture, especially for designs aimed at adapting to the effects of progressive climate change. This concerns not only the shape of objects, but also the dynamic behaviour of their material structure and its changing geometry in response to environmental factors.

2.6. Redundancy

In biological systems, redundancy is a fundamental evolutionary strategy. Hence, multicellular organisms have evolved from seemingly very efficient single-celled organisms. Cellular differentiation and multiple hierarchical arrangements of cells mean that the sum of cells becomes the basic component of a higher-level organisation with additional complexity and increased functionality. Redundancy in biological structures does not only mean that the system has more cells available for action in each tissue than a single task requires, but also that the hierarchical organisation of cells is arranged in such a way that the tissue has sufficient surplus adaptive capacity to change under environmental stress. Redundancy corresponds to the concept of irreducible complexity [M. Weinstock et al. 2004, pp. 40–45]. Biological forms are systems within systems, hierarchically arranged semi-autonomous organisations; each of them performs its own functions, but also has sufficient resources to participate in the reactions of the global organisation. To achieve this, each level of organisation requires differentiation and redundancy. This case of evolutionary benefits conferred by differentiation and redundancy is convincing. However, there are no detailed studies of these benefits for individual biological structures yet.

Evolutionary biology uses redundancy as an important strategy implemented at many levels in multiple and complex arrangements and material diversifications to achieve healthy and sustainable structures. Engineering, on the other hand, has a mechanistic view of material minimisation, simplicity, structural organisation and standardisation [M. Weinstock 2004, p. 15].

2.7. Self-organisation and behaviour

Biological self-organisation occurs under the influence of gravity and other stresses originating from the environment. Growth under the influence of gravitational forces is common and follows morphological geometry and cellular organisation.

The patterns exhibited by all natural systems and the frequency and occurrence of certain geometric patterns (in particular triangles, pentagons and spirals) in many different organisations and divergent scales are astonishing. These patterns are species-specific, so it can be said that biological self-organisation is fundamentally geometric, which also means that for the same set of materials, there is a common organizing principle (Fig. 4).

Fig. 4. Crassula succulentus (Latin: *succulentus – juicy*), geometric patterns of a plant that has adapted to store water in its leaves to survive periods of drought; source: own elaboration

The most important feature of biological self-organisation is that small, simple components combine with each other in three-dimensional patterns to form higher-level structures. These structures then combine into more complex structures that possess emergent properties and behaviours. An example of this is the behaviour of skin. When the skin becomes stretched, its resistance increases with increasing tension as the components of the skin align themselves with the direction of the tension, creating a tension known as linear stiffness. Form and behaviour enter into relationships that are not obvious. The form of an organism influences its behaviour in the environment, and individual behaviours will produce different effects in different environments or when other forms occur in the same environment. Behaviour is non-linear and context-specific [M.Weinstock 2004, p. 14].

Norbert Wiener (1894–1964), based on Whitehead's research, drew up the first mathematical descriptions of 'anticipation and response'. In the 1940s, Wiener developed the first mathematical description of the systematics of reactive behaviour in machines and animals [N. Weiner 1948; 1971, p. 261] providing the theoretical foundations for modern cybernetics. The subject of research was systems in which control (regulation, management) and information processes occur, and the method was mathematical modelling. This opened up a new path for solving many practical, and not just technical, problems. Cybernetics, by using mathematics to describe reactive behaviour, creates a general theory covering machines, organisms and phenomena that occur over time. It uses digital and numerical processes in which pieces of information interact with each other and the transmission of information is optimised. Feedback is understood as a type of 'control' device that regulates behaviour using information from the environment in relation to actual, set or optimal measurements [C.E. Shannon & W. Weaver 1963].

Ilya Prigogine (1917–2003), conducting studies on patterns of formation and self-organisation, expanded this field of research to include issues related to the second law of thermodynamics. This means that every system strives to achieve a state that can be realised in as many ways as possible under given conditions; it therefore strives to maximise entropy [I. Prigogine 1967]. He conducted and documented studies of the behaviour of theoretical biological and non-biological systems. This allowed him to claim that all biological organisms and many natural non-living systems are sustained by the flow of energy through the system [M. Weinstock 2004, p. 14]. The pattern of energy flow involves many small changes that are modified by feedback from the environment until equilibrium is maintained, but sometimes amplification occurs such that the system can either reorganise or collapse. A new order emerges from the chaos of this system at the point of its collapse. Such reorganisation creates more complex structures through higher energy flow through them, and this is a shift towards greater susceptibility to fluctuations and, later, to collapse or reorganisation [N. Paszkowska-Kaczmarek 2022].

Adaptive processes in natural systems have provided the initial concepts for evolutionary and genetic algorithms, which are helpful in designing artificial systems based on natural ones [J.H. Holland 1992]. Genetic algorithms are widely used today in control and optimization applications and in the modeling of pro-ecological systems. Incidentally, mathematical models and Boolean networks allow for the simulation of gene activity. They can produce tissue and organ differentiation in models. Stuart A. Kauffman argues that the self-organization produced by such networks is more complementary than Darwinian selection through adaptation to the environment. [S.A. Kauffman 1993, pp. 373–376, 444–454]. The action of periodic attractors in genetic lattices, proposed by Kauffman, now solves the problem of simulating dynamic gene processes in Cumming's morphogenesis model [M. Weinstock 2004, p. 16].

2.8. Collective behaviour

Collective behaviour arises from the repetition and interaction of simple rules. This is evident in the dynamics of social groups in many natural species. Flocks of birds and schools of fish produce what appears to be a coherent form or arrangement (formation) without a leader or centrally directed intelligence (Fig. 5). Insects such as bees and termites produce complex structural artefacts and highly organised functional specialisations without central planning or instruction.

Mathematical models derived from natural phenomena describe systems of individual 'factors' and units or cells with very simple internal processes, describing their simple interactions. Complex patterns and results emerge from the distribution of data and the connections between dynamic models. Stephen Wolfram's (born 1959) extensive studies on cellular automation [S. Wolfram 2002] provide a comprehensive overview of their characteristics and potential.

Mathematical concepts and techniques for generating collective behaviour, from simple local reactions, have sufficient potential to radically change environmental systems in architecture. The methods currently used in so-called smart buildings employ hybrid mechanical systems controlled by a central computer, which are costly and often unreliable in operation. In natural organisms and conceptual simulations, intelligent behaviours produce self-organising systems with the ability to dynamically distribute data obtained from the environment. Their application is expected to be in the design of climate-sensitive architecture.

Collective behaviors are mathematical models of self-organisation based on variable distribution and selection presented by Francis Heylighen (born 1960) [F. Heylighen 1989]. They are justified in complex systems, such as organisms and ecosystems, which develop from the interaction of elements that combine into various 'clusters'. Some contribute to the natural selection of the whole form, while others disintegrate and undergo further evolution. This process repeats itself at higher levels, and the newly formed whole at one level becomes a component of the system appearing at a higher level [H.A. Simon 1996]. Furthermore, na-

Fig. 5. Collective behavior of fish (mackerels) and birds (starlings); source: Hofmann 2012

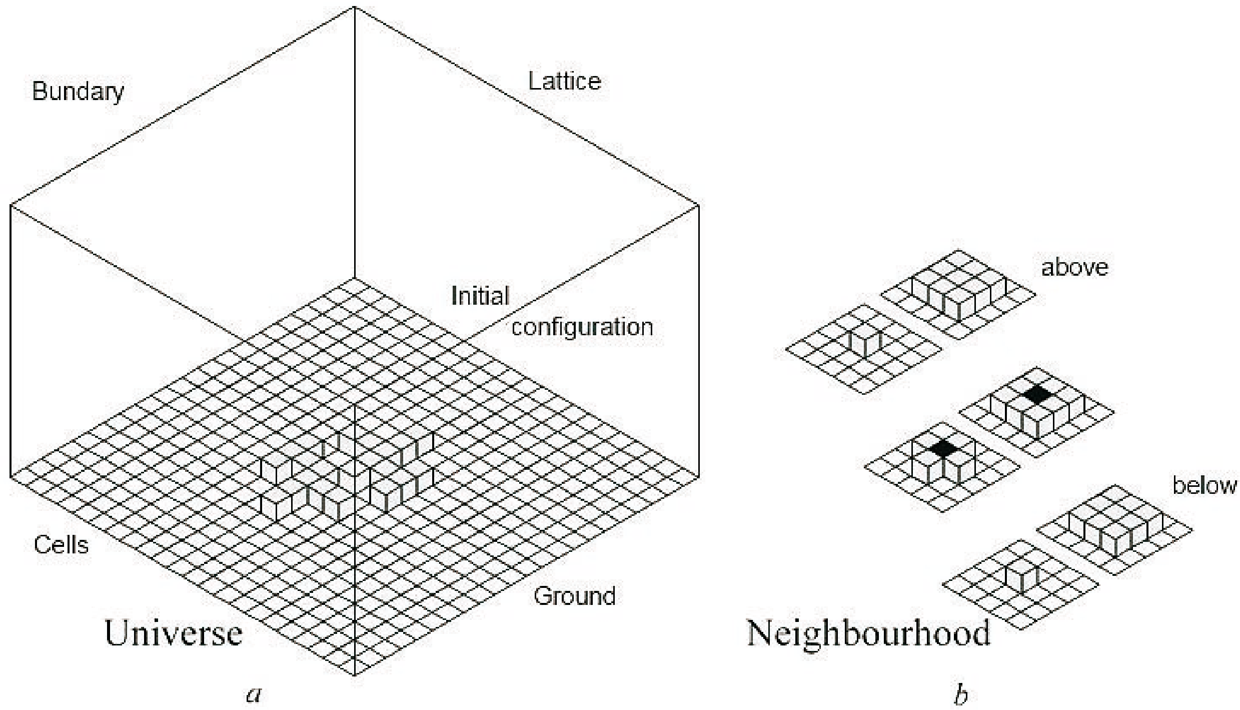


Fig. 6. CA cellular automata – operation diagram; source: Krawczyk 2002

tural evolution does not concern only a single system, but rather many systems and co-systems. The self-organisation of the ecosystem and the whole is just as important as internal morphogenetic self-organisation [K. Januszkiewicz 2013, p. 48].

Designers are waiting for new IT tools that will enable them to make wider use of processes and behaviours resulting from collective behaviour in relation to material and structural components. This concerns climate-sensitive architecture, in the design of which the processes and behaviours explained here could be implemented. Design principles could be abstracted from biological systems and adapted to building design. This requires a deeper involvement in understanding evolutionary development and analysing material organisation systems, which in turn requires understanding the behaviour of individual species. The morphogenetic tools available to designers only allow these tasks to be partially accomplished [N. Paszkowska-Kaczmarek 2022].

4. MORPHOGENETIC COMPUTER TOOLS IMITATING BIOLOGICAL PROCESSES IN THE DESIGN OF ARCHITECTURAL OBJECTS

The imitation of biological processes based on computational models involves design tools that change the traditional understanding of form creation. In computational design, form is no longer defined by a sequence of drawing or modelling procedures, but generated by rule-based parametric processes. The resulting externalisation of the relationship between algorithmic information processing and the creation of the resulting form allows a distinction to be made between process, information and form. Hence, any given shape can be understood as resulting from the interaction between the internal information of the system and external influences within the morphogenetic process.

4.1. Mobile vending machines CA

Cell growth in biological forms can now be imitated in cyberspace. Cellular automata (CA), previously called 'cellular spaces', are a class of automata invented by John von Neumann (1903–1957) and Stanisław Ulam (1909–1984), a Polish mathematician from the Lviv school working in the USA. It is a computational method that can simulate the growth process by describing a complex system through simple units that follow an uncomplicated rule. In the scientific works of Neuman and Ulam in 1951 (*The general and logical theory of automata*) and in 1961 (*Theory of self-reproducing automata*) [J. von Neumann 1951] These units were intended to be idealised models of systems found in biology, in particular describing the logic of self-reproduction [J. von Neumann 1961]. They have been successfully applied in calculations of phenomena from other scientific disciplines such as mathematics, physics, chemistry, sociology and materials engineering. Within these sciences, they are used, among other things, to simulate thermal conductivity, cryptography, random number generation, traffic analysis, neural network simulation, population dynamics analysis, and word recognition. Cellular automata gained greater popularity when Martin Gardner [M. Gardner 1970] described John Conway's 'Life', a game that generated two-dimensional patterns. In the early 1980s, Stephen Wolfram [S. Wolfram 1994] began research into the concept of representing physical phenomena, opening a debate on a new scientific discipline [S. Wolfram 2002].

CA cellular automata are discrete models of space and time. They typically involve interactions between cells in uniform square grids. Cells can take on a specific, finite number of cell states, which can change according to simple rules that each cell executes in relation to its surroundings [A. Ilachinski 2001].

This means that each cell is updated synchronously, according to local interaction based on the principle adopted for determining the state of neighbouring cells. While each individual change in cell state may seem insignificant to the human eye, the patterns generated across the entire CA cellular network are often complex and difficult to predict [Ch.M. Herr 2015]. A cellular automaton consists of a data structure and an algorithm that operates on it. This structure takes the form of an array of cells of a certain type. It can be an array of any number of dimensions – from a one-dimensional vector, through a two-dimensional matrix, to three-dimensional and higher-dimensional arrays. The most commonly used algorithms are well-known ones such as 'the game of life' and 'Langton's ant'.

The three-dimensional universe of CA cellular automata consists of an unlimited network of cells (Fig. 6a). Each cell has a specific state (from 0 to 1) occupied or empty, represented by a marker recording its location. The transition process begins with the initial state (occupied) of the cells and proceeds according to a set of rules for each successive generation. The rules determine who will survive, die, or be born in the next generation. The rules use the cell's neighbourhood to determine its future. The neighbourhood can be defined in many ways. Figure 6b shows two popular methods for determining which neighbouring cells to consider. The rule developed by Conway in 1970 is: 'check the neighbourhood of each occupied cell; survival occurs if there are two or three neighbours, death occurs if there are any other number of neighbours, and birth occurs in an empty cell if it is adjacent to only three neighbours.'

As each generation evolves over time, one of four cases may occur. Either the cells find a stable form and do not seem to change; or they become a so-called 'flicker' and alternate between two stable states; or all or a cluster of cells become a 'glider,' a group of cells that begins to orbit the universe forever, or all cells die, i.e., they extinguish themselves. Other rules can be proposed, with Conway's rule being the traditional starting point.

Architectural interpretation

Direct translation of the results of the CA mathematical model into architectural language is not entirely possible, as this model does not take into account either construction or usage realities.

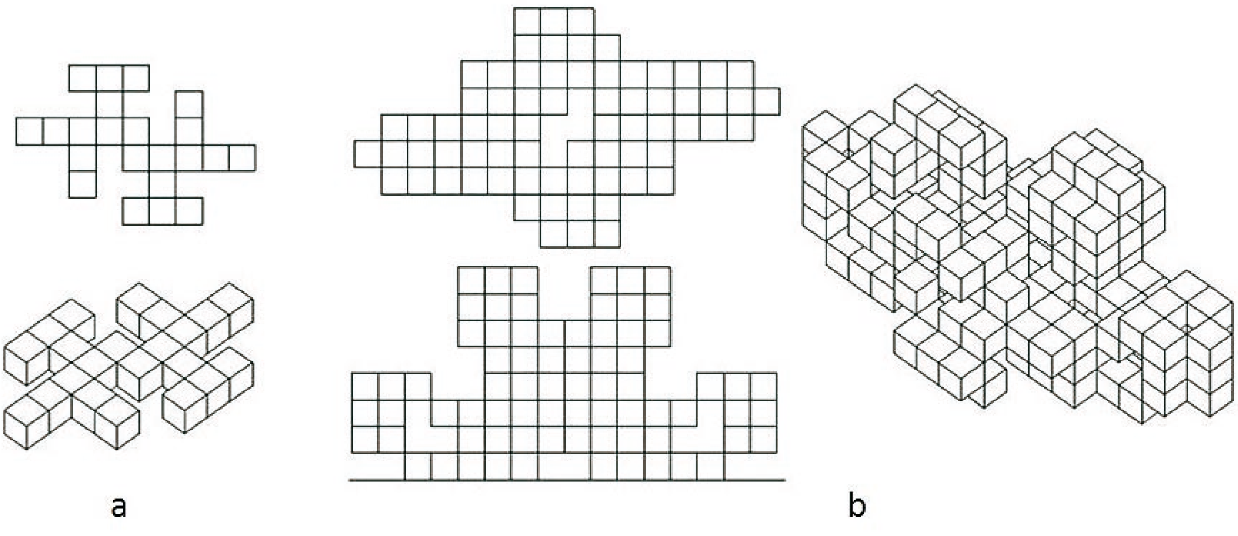


Fig. 7. Cell structure of the 8th generation with a designated area; source: Krawczyk 2002

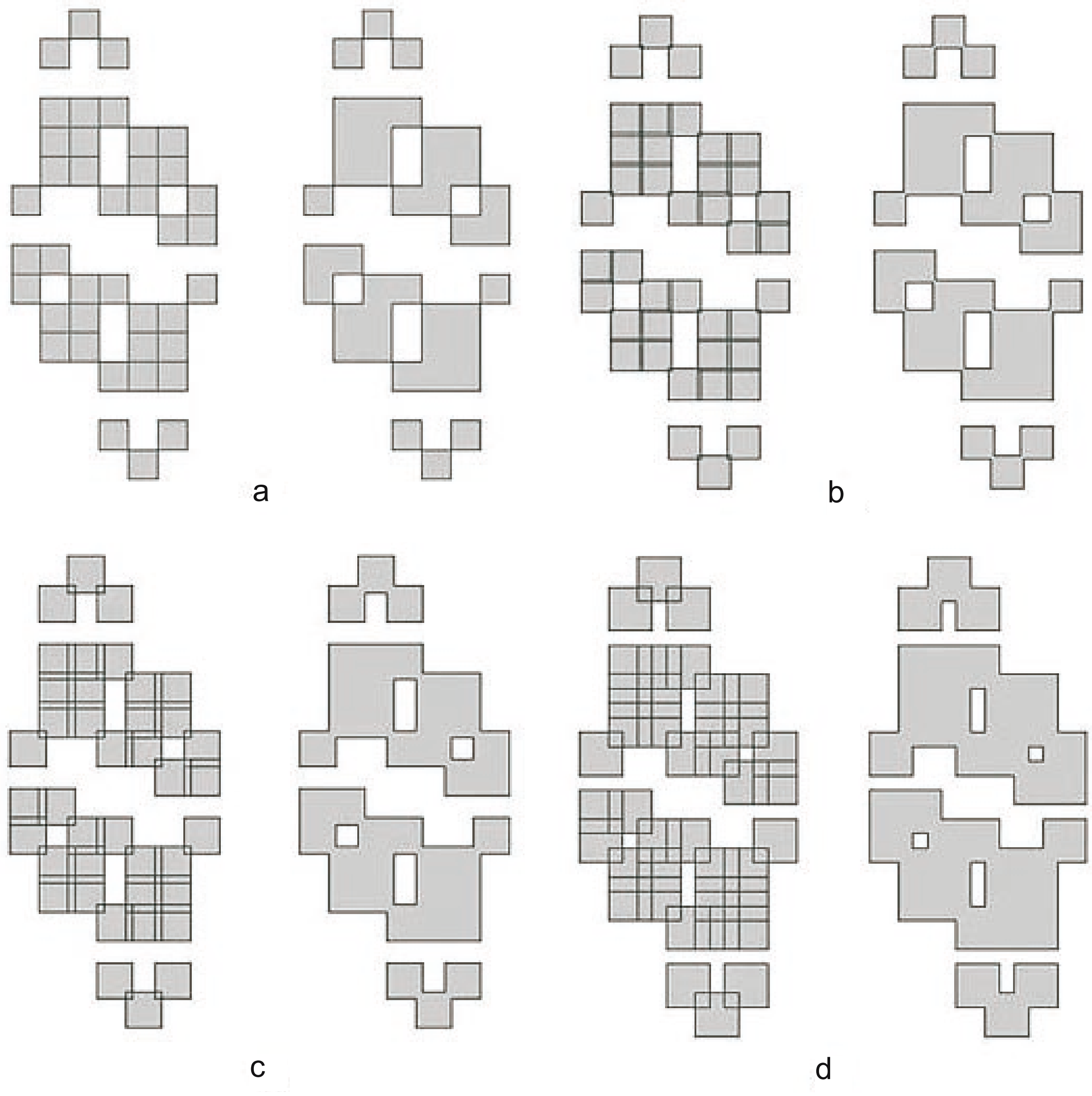


Fig. 8. Architectural interpretation – internal divisions; source: Krawczyk 2002

For example, when analysing the initial configuration and its processed results up to the eighth generation (Fig. 7b), one may ask whether these results can be interpreted in terms of building structures. In order to more easily translate the obtained cell configurations into the language of future buildings, there are two approaches: observing how the system configures itself or imposing certain restrictions on the system at the beginning of its operation. It is also possible to combine both approaches. In this case, a boundary representing a certain area is marked on the cell board and the directions of cell growth (e.g. vertically and sideways) are indicated, but not below. Then, the operation of the system can be observed and stopped at a satisfactory moment of component growth (Fig. 7).

Another problem is that some cells may not be horizontally connected to others, and others are not supported. Furthermore, the cells have no architectural scale and do not suggest any internal usable space. Fig. 8 shows the projections of individual cell layers and divisions suggesting solutions to these problems in terms of architecture and construction. The centre of gravity of each cell is important for further interpretation of the results.

Furthermore, cells can be adopted that could articulate the edge of the building in a different way

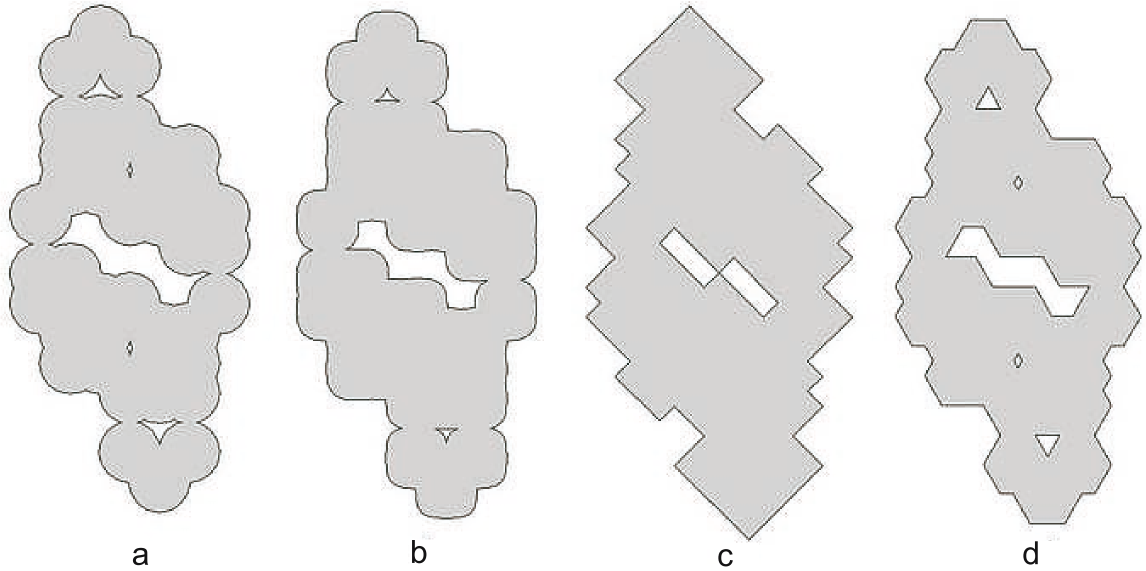


Fig. 9. Articulation of the building edges; source: Krawczyk 2002

than a square and that could take into account the orientation and additional surface area in the façades (e.g. window openings) (Fig. 9). These can be circles, ellipses, polygons, etc. The spaces of these units can be connected not only by creating large, adjacent areas, but also by defining a series of boundary points, an envelope that can be further interpreted or transformed (Fig. 10). Depending on the shape of the unit cell, they begin to develop into different curved edges

As mentioned earlier, the initial cell configuration also lacked vertical supports and struts. This problem can be solved by utilising the growth process, limiting the growth of the cell supporting the component from below or adding supports to the final configuration (Fig. 11). Two possible support strategies are shown above, one with posts at each corner of the cell and the other with supports placed in the centre of the cell.

Other approaches to the architectural interpretation of elementary cells are also used. For example, Robert Krawczyk proposes that at the beginning of the process, the size of the cell should be defined as minimum and maximum, and the actual size should be selected randomly (shifting each vertex). When it comes to interpreting cells as they are created, the 'growth stopped' option can be used in each generation. When a cell survives, it increases in size. This approach takes into account the actual growth process in nature and imitates it directly [R.J. Krawczyk 2002].

The example presented shows how a mathematical concept imitating the growth process can be transformed or interpreted into architectural elements. Working with CA systems is fraught with two types of difficulties: the unpredictability of the results of simple rules, and the difficulty of determining the rules that lead to the desired results. The aim of this analysis was to identify critical issues concerning both the operational model and the design strategies adopted. These relate to the decisions that the designer should make at the beginning and during the process of creating the configuration of components. The main points are as follows:

- what should the initial configuration of the cells be? Could Jean L. Durand's compendium of neoclassical design rules be helpful here;
- which generation to stop at and whether it is enough to just evaluate the designer's imagination;
- how to define neighbourhood and its permanent elements;
- when and what rules to adopt for growth;
- when and how to define cells, their shape as spatial units;
- what scale to adopt for the generated multicellular object;
- how to define support conditions, configure connection networks and limit the number or surface area of generated components.

It should be noted that cellular automata, viewed as a mathematical approach, differ from traditional deterministic methods in that the current results form the basis for the next set of results. This recursive substitution method is continued until a certain state is reached. Fractals and strange attractors are also created in a similar way. Many digital methods in architecture are parametrically controlled. An initial set of parameters is used to generate a single result. If an alternative is desired, the parameters must be modified and the

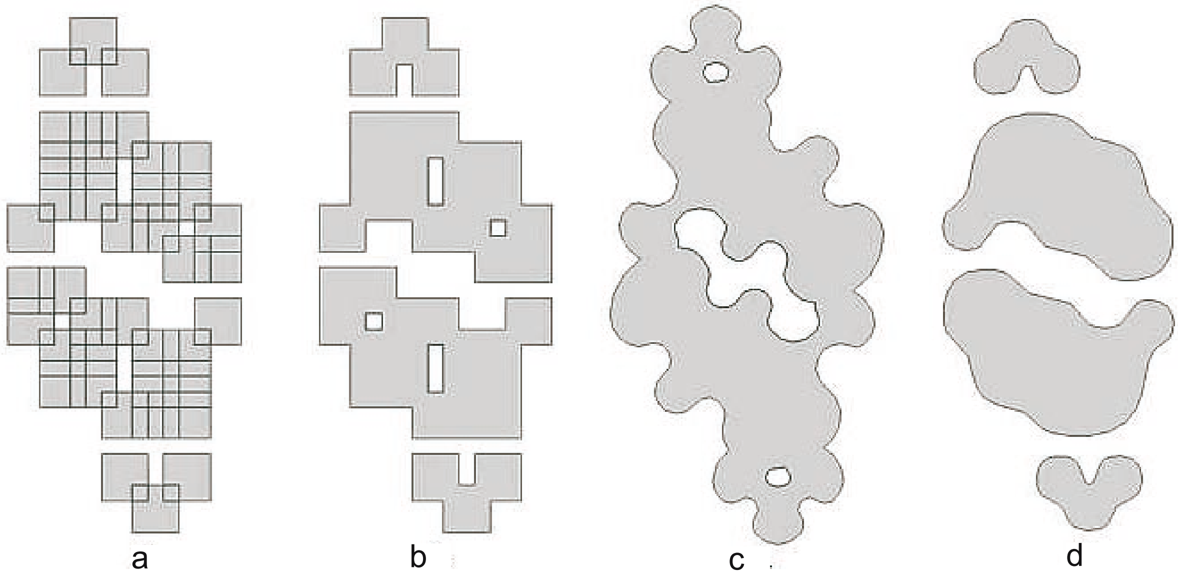


Fig. 10. Determining the support and the edge of covering; source: [R.J. Krawczyk 2002]

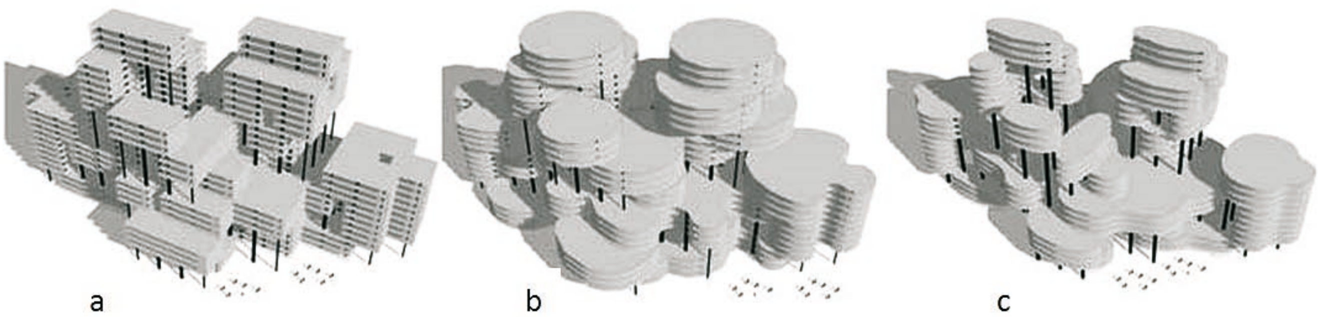


Fig. 11. CA cellular automata – variant assignment of architectural features; source: Krawczyk 2002

generation repeated from scratch. The difference between the two methods is that in parametric methods, the results are easily predictable, while in recursive methods, the result is usually unpredictable. This provides an interesting and rich platform from which to develop possible architectural patterns.

The spectrum of problems solved using cellular automata is broad thanks to the simplicity of their mechanics and the high flexibility of their assumptions. Architects most often use cellular automata because of the ease with which they can generate various geometric patterns, from ornamentation to the automatic volumetric generation of building structures. Ingeborg M. Rocker uses cellular automata (CA) to generate forms on a building scale, while Michael Batty uses them to design groups of objects on an urban scale. Mike Silver, on the other hand, presented his competition design for the San Jose State University Museum of Art and Design (2003), in which cellular automation was used to a greater extent than ever before [M. Silver 2006].

Following the introduction of cellular automata (CA) by Stanisław Ulam and John von Neumann in the late 1940s, many different types of cellular automata were developed, which became part of what Christopher Langton called 'artificial life' in 1986. The most complex examples are based on stochastic development, hence their structural properties are common to morphogenetic models, e.g. Alan Turing's (1952). This is the reason why some cellular automata are able to simulate the development of living beings, but also cities and artefacts.

Patterns often appear in nature, from the colouring of a leopard's fur to the shape that a fern takes as it grows. Stephen Wolfram proposed the idea that these patterns are generated in a manner similar to cellular automata. In this model, each pigment cell behaves according to the state of each of its immediate neighbours, similar to how a cellular automaton works. Another example of this is the way mollusc shells are generated. Their shells are extruded one layer of cells at a time (similar to fingernails), so complex colour patterns can act as one-dimensional cellular automata (Fig. 12). These patterns can be not only inspiring but also imitable on both straight and curved architectural surfaces.

Currently, cellular automata are widely used in parametric design to create objects at various scales that can be manufactured according to the principle of mass customisation (Fig. 13).

4.2. L-Systems

The L-system is what imitates plant growth in computer science, and it has been named the grammar of form. It is an abstract structure described by the language of form through sequences of simple objects called strings (equivalent to natural chromosomes). Plant modelling based on the L-system is described by two elements, i.e. two categories of formal grammar: analytical and generative. Analytical grammar determines whether a string belongs to the language described by that grammar. Generative grammar, on the other hand, is formalised by an algorithm that generates strings in a specific language and consists of a set

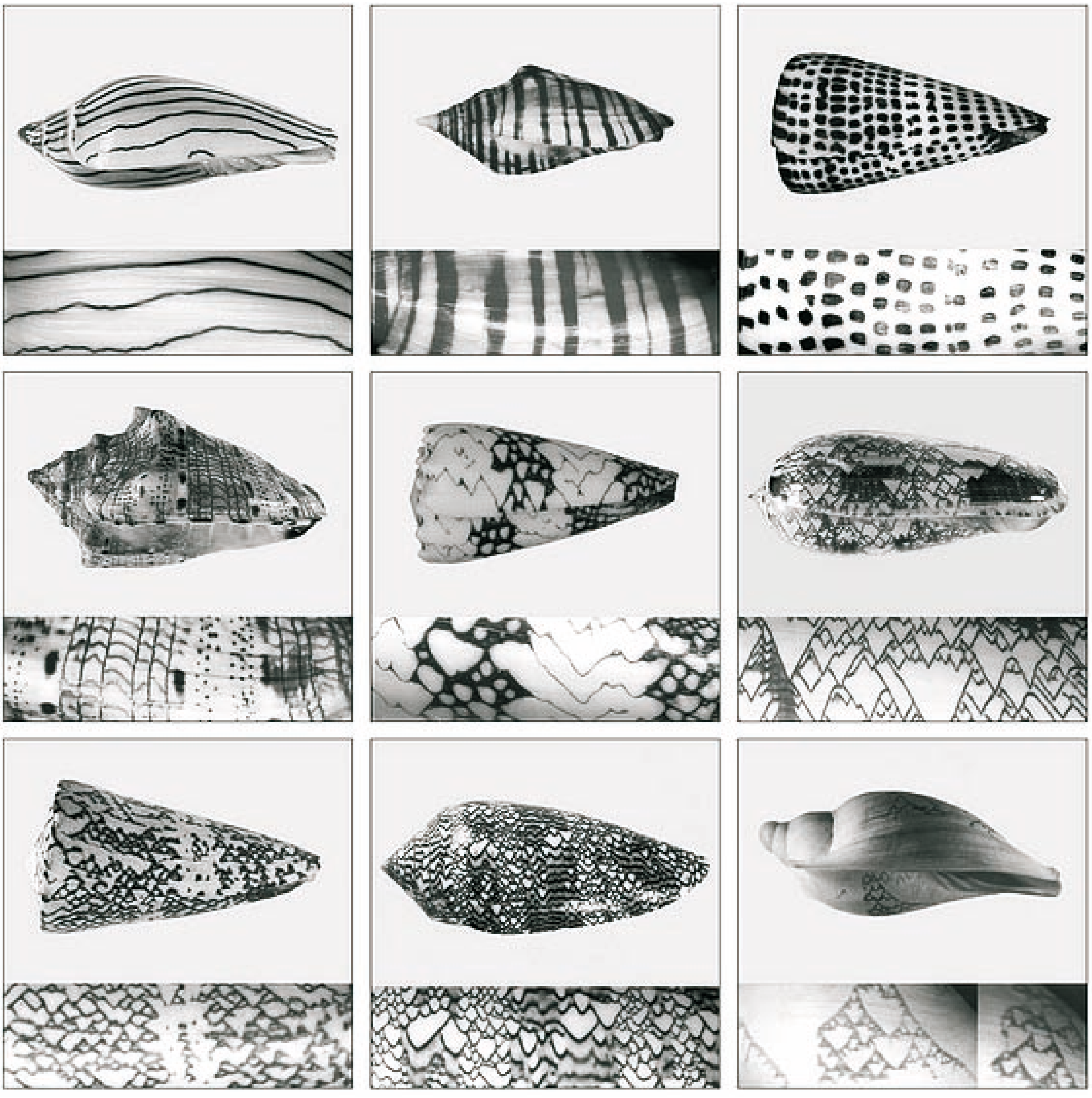


Fig. 12. Patterns on crustacean shells and their pigmentation that can be imitated using cellular automata; source: Wolfram 1984

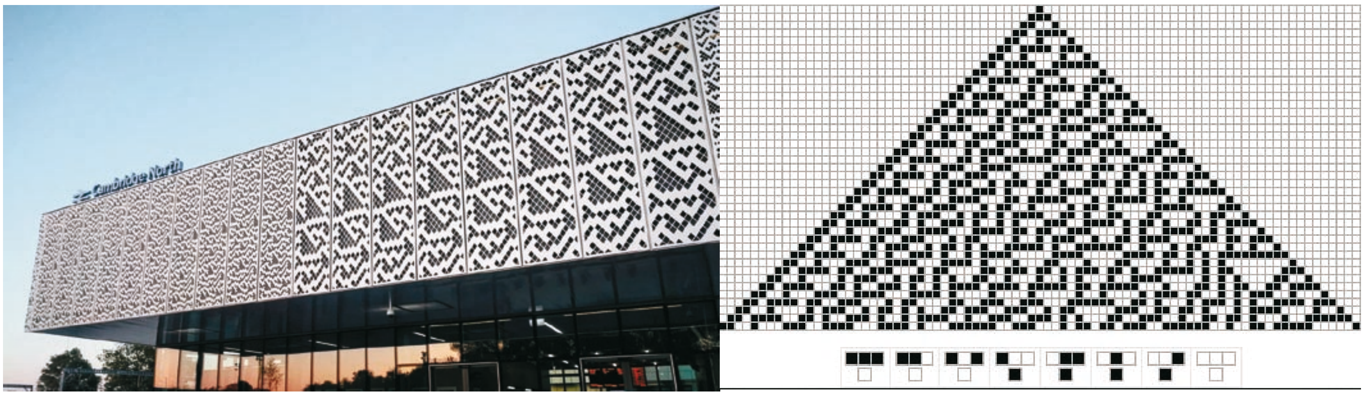


Fig.13. The facade of Cambridge North station, the pattern of which was generated by Automata cellular – the rule no. 30 was used, according to which obtained pattern is analogous to the patterns on the Conus textile shells; source: Wolfram 1984

of rewriting rules that transform strings, starting with a simple start symbol [K. Januszkiewicz 2012].

In 1968, Hungarian biologist Aristid Lindenmayer (1925–1989) developed a formal grammar later named the L-system. He conducted research on mathematical models describing the growth process of simple multicellular organisms. It is used to generate shapes with a fractal appearance, which have been used in particular to describe the appearance and stages of development of various plant species. Przemysław Prusinkiewicz also conducted research on L-systems together with Lindenmayer, which resulted in their joint publication on the subject [P. Prusinkiewicz & A. Lindenmayer 1990].

The rules governing L-systems are written as a sequence of characters, where each element has a specific function. Repeating these rules multiple times produces predictable, complex effects. The way in which shapes are drawn is determined by a given interpreter. One such interpreter is the ‘turtle’ developed by Prusinkiewicz. The initial state of the ‘turtle’ is determined by its position coordinates (x, y) and direction (α). It can then execute the commands written in the rules of a given system.

Using the L-system in computer graphics requires translating symbols into graphic structures. Depending on the model, different methods are used to transfer formal notation into graphics. One example is the so-called ‘turtle graphics’ (similar to the concept used in the Logo language). In this model, each symbol in the L-system is interpreted as a specific sequence of ‘turtle’ movements (Fig. 14).

The basic four commands are “F” – move forward drawing a line, “f” move forward without drawing a line, “+” and “-” turn left and right. With the help of the “turtle” you can draw not only plants – in a very simple way you can present the Koch curve and similar shapes.

Today, the L-system forms the basis for IT structures that simulate processes occurring not only in the natural world. This system has also found practical application in the generation of fractals. Special cases of the L-system are: the DOL (deterministic and context-free) system and the stochastic L-system, which is assigned a probability through stochastic grammars.

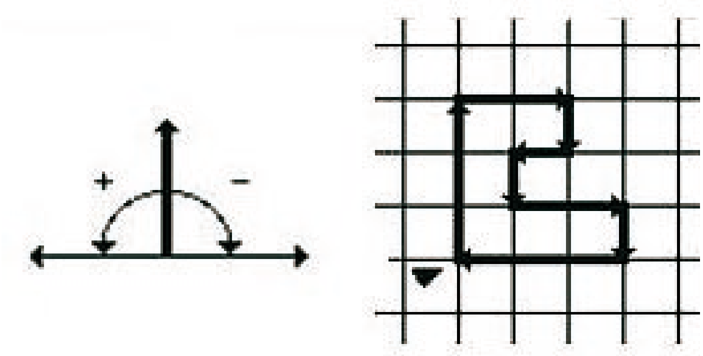


Fig. 14. Interpretation of the system rules by the “turtle” FFF-FF-F-F+F+FF-F-FFF and with tree structure, axiom: --F, rule: F: FF+[+F-F-F]-[-F+F+F], angle: 22.5 degrees; source: own elaboration

Architectural interpretation

Translating the results of L-systems directly into architectural language seems, at first glance, illogical – buildings do not grow like flowers or trees and serve completely different functions. However, with the development of information technology, this common view is changing. The L-system algorithm can be extended to facilitate the creation of architectural geometries and spatial denotations of functional use.

First, the graphical commands for the turtle were improved to control movement in 3D space. In such an environment, the geometry of solids can be generated by extruding it along the turtle’s path or by using segments of that path to create surfaces (e.g., loft). To control the drawing rules of L-systems with greater precision, parameters can be introduced into the string rewriting rules. At a basic level, this may involve specifying incremental changes to the turtle's state. The first experiments were conducted by Michael Hansmeyer [M. Hansmeyer 2003]. To enable a single L-system to generate multiple design permutations, Hansmeyer proposes stochastic substitution rules. Furthermore, substitution rules can be limited in scope so that they apply only to certain generations of the rewriting process. Further geometric freedom is achieved by introducing variable parameters directly into the turtle graphics language. Finally, the system is extended with environmental interaction, so that the turtle performs different connections (strings) depending on the local conditions and connections encountered [M. Hansmeyer 2003]. The logic of L-systems is particularly well suited to the production of modular systems. Since not every letter in the string has to correspond to a turtle instruction, it is possible to introduce letters that are simply replaced by groups of other letters. For example, in a system where (a) implies forward movement and (a c) means a right turn, a new letter (d) can be defined to create a square. The square, marked with the letter (d), can now be used as part of other modules, which in turn can become components of even larger modules (Fig. 15).

The Modularity project (Fig. 15) was generated by an L-system and is the result of research into whether nature's patterns and algorithms can open up new possibilities for architecture. It is the result of research conducted by Michael Hansmeyer on the logic of nature's developmental processes, which acts as a generator of architectural design, its application to the production of architectural forms and additional functions, such as the creation of organisational logic, space segmentation and the development of a structural system. The logic of L-systems can therefore be used not only to create the spatial organisation of components, but also to differentiate and articulate these components. By introducing parameters into the definition of components, they become flexible and adaptable to local conditions. Furthermore, three-dimensional Cartesian geometry space can also be based on 'agents'. This means that an agent moves in space with a position vector and three orthogonal axis vectors. Its instantaneous orientation is determined by a rewriting rule, which can, for example, direct it to a certain degree forwards, upwards or sideways.

The spatial denotation of functional features requires a branched structure in which the branches 'grow' in 3D space. It is not so much about manipulating L-system grammars as it is about a method that ensures complete freedom of choice of form. Paul

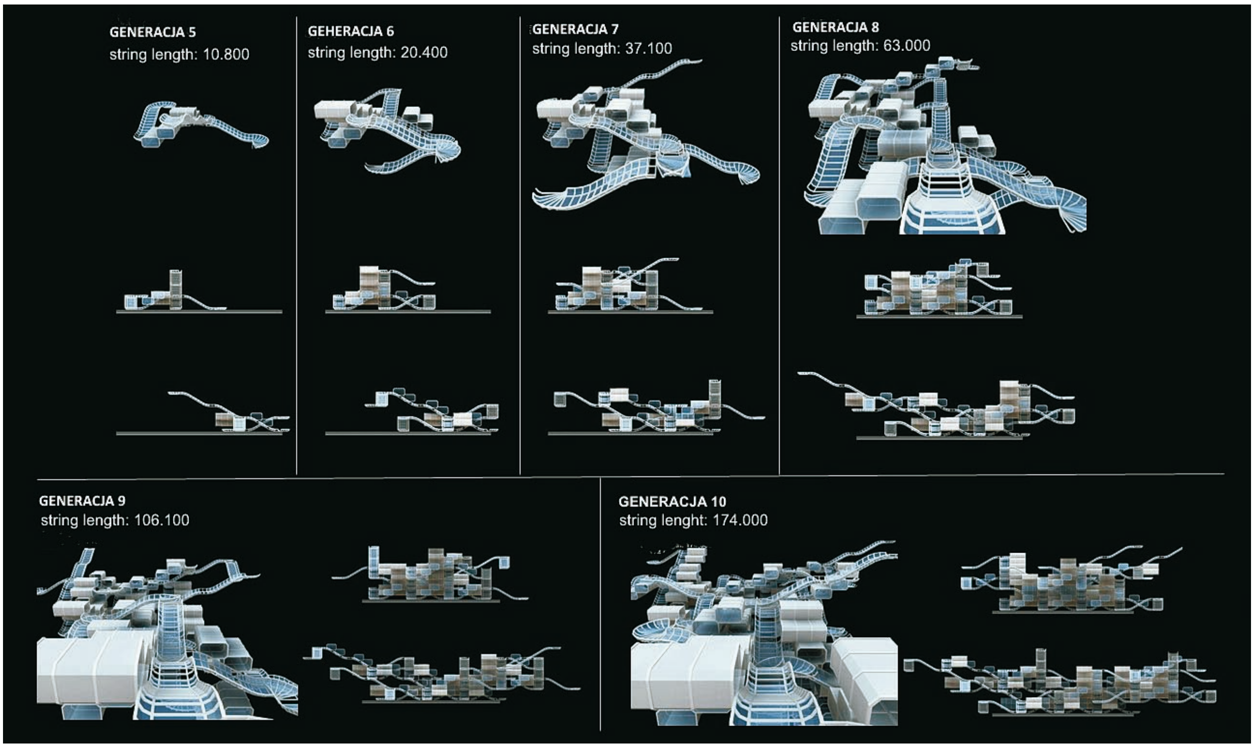


Fig. 15. Michale Hansmeyer, Modularity, 2003; L-system logic used for the spatial organization of components and their differentiation and articulation; source: Hansmeyer 2003

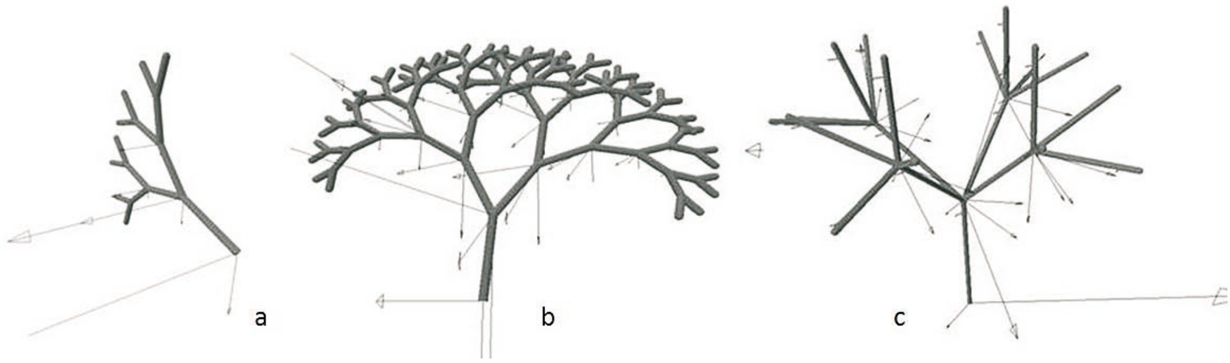


Fig. 16. Simulations by L-system of the distribution of forces and moments in real time in the application to the XML numerical program a) 3D binary tree after the fourth iteration, b) complete 3D binary tree model with rigid fixing, c) simulation taking into account the action of external forces (e.g. wind load); source: H. Noser et al. 2001

Coates proposed combining Lindenmayer systems with genetic programming [P. Coates et al. 1999]. Coates' modified L-system develops its branched structures in an isospace grid by filling the points of intersection with the grid with spheres. This shows how a function can guide form. The choice of an isospace grid (rather than a Cartesian grid) due to its lack of orthogonal deviation and homogeneity allows for complete freedom in the choice of form. The system searches for functional forms using genetic programming [U.M. O'Reilly & M. Hemberg 2007]. Coates' research has contributed to the development of systems that produce different designs [J.R. Koza 1992].

L-systems are also used and developed in structural engineering. New applications are being sought that would simulate the distribution of forces and moments under physical conditions and automatically perform calculations (Fig. 16).

The combination of L-system logic with the physical behaviour of building structures and computer graphics is intended to support the conceptual phase of engineering design. The XML numerical programme is based on parametric grammar similar to the L-system, which is why the user can first associate mathematical equations with symbols and then use them in the rules for defining objects. In subsequent iterations and interpretations of these rules, the object is visualised and the application automatically generates sets of files with mathematical equations describing the physical design of the object. These equations can also be solved automatically using already known IT techniques. This application was first developed in 2001 by computer scientists from the University of Zurich in collaboration with the Department of Statics and Dynamics of Aerostructures at the University of Stuttgart [H. Noser et al. 2001].

The extended Lindenmayer growth process model can also be used to generate complex surfaces when it interacts with two factors: geometric grain (a set of basic geometric data) and rewriting rules. These rules should determine how elements will change their shape, as well as the process in which geometry is repeatedly reinterpreted. In this way, surfaces can be obtained that consist of diverse components according to the seeded geometry. The 'Fibrous Surface' project (2005) is a synthesis of the above activities (Fig. 3.17). The project was carried out at the AA School of Architecture in London under the supervision of Michael Hensel and Achim Menges [A. Menges 2006].

In the example mentioned, the surface is represented by a structure data graph (a set of geometric data concerning edges, vertices and regions). During growth, all edges are constantly rewritten, and all parts of the surface change continuously until the ontogenetic drift configuration stabilises (ontogenetic drifts are developmental changes in form and function that are inherent to the growth process). Based on the surface on which the geometric seed was sown, further material data is introduced to generate elements for fabrication. In response to specific geometric features, such as global undulation and regional curvature, a variable distribution algorithm establishes a network of lines on the "surface" platform that determine the position of each fibre and the type of node. Next, digital components fill the system according to the command "construct a virtual solid model". The result is an organisation in which fibres only intersect when they are perpendicular to each other, which results from the input data regarding fabrication constraints. Otherwise, they pass under or over intersecting elements (similar to the structure of a bird's nest), thus creating a geometrically defined, self-clamping and stable structure. It is therefore possible to correlate geometric definition, structural behaviour and production logic, as in natural morphogenesis where formation and materialisation are inseparable. This

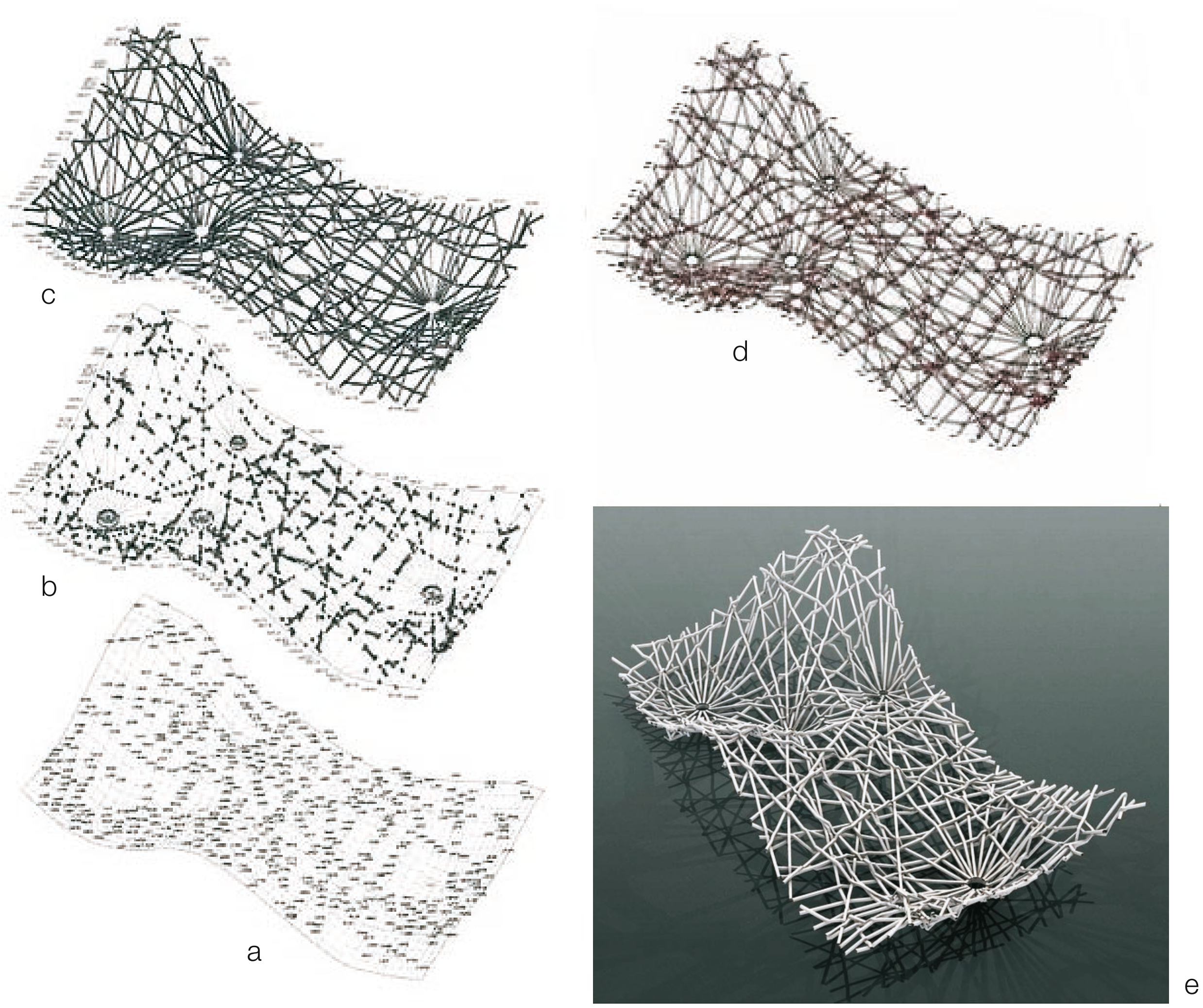


Fig. 17. Sylvia Felipe, Jordi Truco, Fibrous Surface, 2004. Digital Growth and Ontogenetic Drifts. Diagram: Surface geometry generated by a) digital growth process based on extended Lindenmayer systems (a) (bottom) provides geometric data for the algorithmic decomposition of parametric components (b) (middle), resulting in a complex network of self-locking straight rods (c) (top), ready for fabrication. d) digital definition of the fiber network obtained by synthesizing digital processes of component differentiation, propagation mapping, and digitally simulated growth, e) view of the fibrous self-locking surface structure. The tested prototype consists of about 90 components and 1000 connections; source: A. Menges 2006b

correlation is not only consistent within a single system (manufacturing), but is integral to the very process of generation driven by L-systems. This is particularly significant when one considers that the surface defining the morphogenetic contribution is constructed in a bottom-up process in which all parts respond to local interactions and the environment. Because these internal and external interactions are complex and the interpretation of L-systems is non-linear, the outcome of the growth process remains open. This approach to design enables architects to define specific material systems through a combined logic of formation and materialisation. It promotes the creation of specific shapes by developing the performative capabilities inherent in material arrangements and structures that are typically derived. Most importantly, it encourages a fundamental rethinking of current mechanistic approaches to sustainability and the associated functionalist understanding of efficiency.

Contemporary methods utilising Lindenmayer's achievements enable morphological processes and adaptogenesis to be carried out, replacing manual experiments. Adaptogenesis is a continuous process of creating new adaptations as a result of evolution. Adaptation is defined as the process of individuals adjusting to environmental conditions based on evolutionary modifications. Architects such as Karl S. Chu and Emergent Design Group, as well as OCEAN NORTH, use L-systems as a tool for generating forms and structures. L-systems are a promising tool for those interested in an alternative approach to environmental sustainability.

4.3. Evolutionary and genetic algorithms

Synthetic evolutionary algorithms imitate the mechanisms of evolution, the same ones that occur in biology. It is a set of methods and techniques that include not only genetic algorithms, but also genetic programming and evolutionary strategies. Evolution is now an indisputable scientific fact, documented by evidence from many fields of science. The essence of evolution is the combination of random (undirected) changes in genotype with strictly directed environmental pressure. It proceeds according to the following general principles:

- The genotype of a given individual undergoes modifications during reproduction. These changes may result either from minor, random mutations or from the mixing (crossbreeding) of the traits of the parent individuals;
- Changes in the genotype cause changes in the phenotype of offspring, which affects their degree of adaptation to the environment (assessed using the objective function);
- Changes in genotype are random. Changes that are beneficial to the individual occur just as often as those that are detrimental or neutral.
- Individuals are evaluated by comparing their adaptation to a given environment. Those that are better adapted have a greater chance of reproducing.
- Less well-adapted individuals succumb to competition for limited environmental resources and die out.
- The genotype of an individual is subject to changes (mutation, crossbreeding), while phenotypes are subject to selection [D.E. Goldberg 2009].

Biological evolution drives morphological diversity through genetic variability and results in high levels of adaptation, efficiency and effective resource management. Therefore, a synthetic evolutionary algorithm modelled on biological evolution is used for optimisation and modelling tasks. It boils down to finding the value of variable x, contained in a given set X, at which a given function of variable x takes the most favourable value. Function, called the objective function or quality indicator or criterion, measures the goal to be achieved [Z. Michalewicz 2003].

Classic optimisation algorithms typically use a deterministic procedure that approaches the optimal solution step by step (Fig. 18). Such a procedure usually begins the search for the optimal solution by starting from a selected solution, after which the direction of the search is determined based on local information. Next, a unidirectional search for the best solution is conducted. This best solution becomes the new solution, and the above procedure is repeated a specified number of times. Classical algorithms differ mainly in the way they

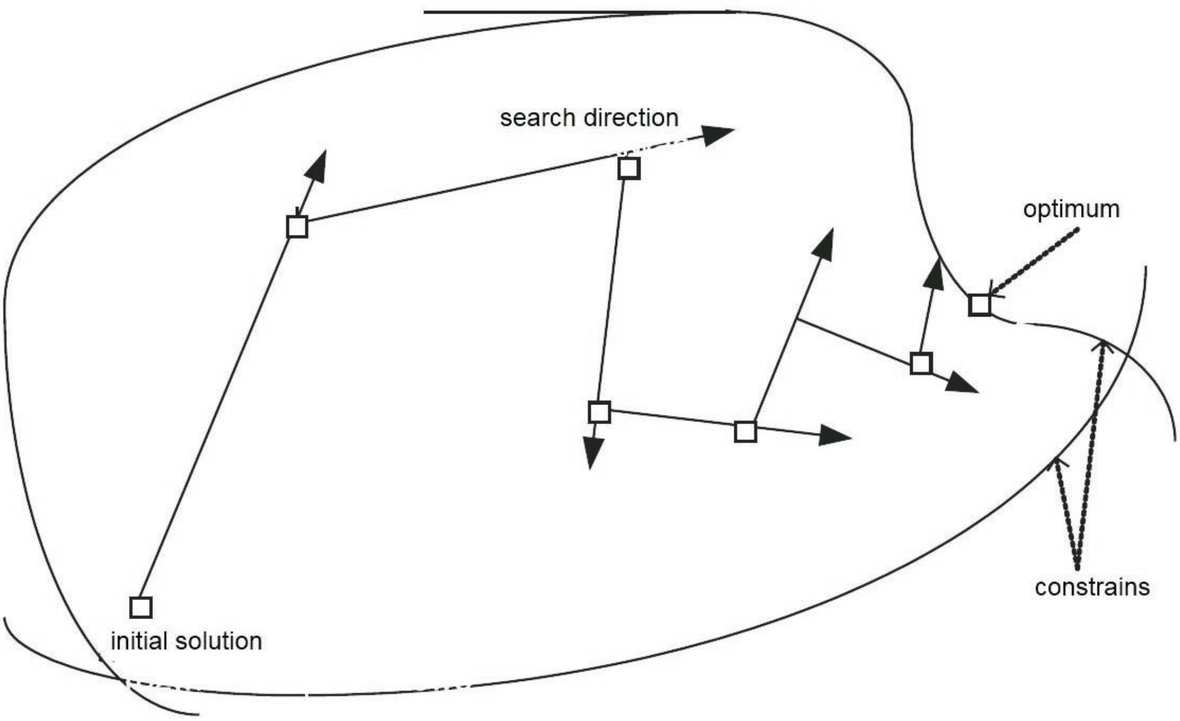


Fig. 18. Operation of the classic optimization algorithm; source: Figielska 2006

determine the direction of the search. The convergence of the algorithm to the optimal solution depends on the choice of the initial solution. Evolutionary algorithms can handle most of the difficulties encountered when using classical algorithms [D.E. Goldberg 2009].

Evolutionary algorithms have a unique ability to adapt easily and can be used to solve complex non-linear and multi-dimensional engineering problems. The quality of their performance does not depend on the problem; its structure or differentiability is irrelevant.

The basic features of evolutionary algorithms that distinguish them from other methods are:

- they do not process the task parameters directly, but rather their encoded form;
- they conduct searches, starting not from a single point, but from a certain population of them;
- they use only the target function, not its derivatives or other auxiliary information;
- they use probabilistic rather than deterministic selection rules [E. Figielska 2006].

The operation of the evolutionary algorithm can be described as follows: the algorithm begins the search process by creating a population of potential solutions called individuals, which are represented by chromosomes containing genetic information about these individuals. In each evolutionary step, called a generation, the chromosomes are decoded and evaluated according to a predetermined quality criterion called fitness (the fitness function can be, for example, the objective function), and then a selection is made to eliminate the individuals evaluated as the worst. Individuals with high fitness undergo mutation and recombination performed by a crossover operator. Selection alone does not introduce any new individuals into the population, i.e. it does not find new points in the search space, but such points are introduced by crossover and mutation. Thanks to crossover, the evolutionary process can move towards promising areas in the search space. Mutation prevents convergence to a local optimum. As a result of the crossover and mutation operators, new solutions are created, from which the next generation population is then built. The condition for terminating the algorithm may be, for example, a certain number of generations or the achievement of a satisfactory level of fitness [Z. Michalewicz 2003, pp. 81–92; E. Figielska 2006].

Genetic algorithms are the best-known class of evolutionary algorithms. The basic genetic algorithm was developed in 1975 by John H. Holland (1929–2015) [J.H. Holland 1975] in the FORTRAN programming language and developed thanks to his work in the 1960s and 1970s. Traditionally, chromosomes in genetic algorithms are binary strings of fixed length (chromosomes can also be encoded using strings of integers or real numbers). Chromosomes are evaluated in each generation. The population size is fixed. The parameters that must be specified are: population size, mutation probability, and the condition for terminating the algorithm. The fitness function must also be specified in advance. *Incidentally*, the idea of applying Darwin's principle of natural selection to automatic problem solving emerged in the 1950s, before the technology became widely available. In the 1960s, three different ways of imitating evolution appeared in three different places. In the United States, Fogel introduced evolutionary programming [M.J. Fogel et al. 1966], while Holland called his method a genetic algorithm. In Germany, Rechenberg and Schwefel [I. Rechenberg 1973; H.P. Schwefel 1981] proposed evolutionary strategies. For about 15 years, these methods developed independently. It was not until the early 1990s that they came to be regarded as different forms of a single technique – evolutionary algorithms.

The genetic algorithm works as follows: in a defined environment, an initial population of individuals is initiated (usually randomly) and evaluated. Each individual must be assigned a set of information constituting its genotype, which provides the basis for creating a phenotype. The genotype consists of chromosomes, where the phenotype and possibly some auxiliary information for the genetic algorithm is encoded, while the chromosome consists of genes. The phenotype is therefore a set of traits that are evaluated by an adaptation function that models the given environment. In other words, the genotype describes the solution to the problem, and the phenotype (fitness function) evaluates how good that solution is. The initial population is evaluated (selected) so that the best-adapted individuals participate in reproduction. Only genotypes undergo evolutionary processes and are associated by crossing with the genotypes of their parents and undergo mutation (small random changes are introduced). A new generation is created, which, after evaluation, will become the basis for the next step of the algorithm. The algorithm returns to step two if a sufficiently good solution is not found. Otherwise, the result is obtained [K. Januszkiewicz 2012b, p. 49] (Fig. 19).

Each solution has a set of characteristics encoded in its genome (this may be a set of numbers) that determines its phenotype. In the first step of the algorithm, a series of random genotypes is generated. Each phenotype is evaluated in terms of its fitness, allowing for the selection of genotypes that are crossed with each other in the next step of the algorithm. In this way, a new generation of genotypes is generated with a theoretically better average fitness for the entire population. Repeated testing of fitness, selection, crossing and reproduction of the population brings us closer and closer to finding the optimal solution. The method of crossing is important, as it should take into account the occurrence of so-called local optima, i.e. places in the solution space that contain a number of good (but not the best) solutions separated from the rest of the space by solutions with low efficiency. The algorithm often gets stuck in such a place due to the way it works. The quality of the results obtained using genetic algorithms depends on the size of the population, the time allocated to searching for a solution, the selection method, the crossover and mutation operators used, and the probability with which these operations are performed. Genetic programming is very similar to genetic algorithms. It also uses the principles of genetics and natural selection in the creation of digital programmes [J.R. Koza 1994]. The fundamental difference between genetic programming and genetic algorithms lies in the representation of the solution. While genetic algorithms create a sequence of numbers that represents the solution to a problem, in genetic programming, individuals are tree-structured programmes, and genetic operators are applied to the branches and nodes in these trees.

In addition to genetic algorithms, evolutionary programming is also used. It focuses mainly on optimisation problems with continuous parameters. The main area of application is the optimisation of real multidimensional functions. In evolutionary programming, each parent individual in the population generates offspring through mutation. The probability of mutation is generally uniformly distributed. After evaluating the offspring, a variant of stochastic tournament selection selects a certain number of the best individuals from the set of parents and offspring. The best individual is always stored, which ensures that if the optimum is reached, it cannot be lost. Evolutionary programming is an algorithm that uses only selection and mutation mechanisms (without crossover) [J.R. Koza 1994].

The optimisation process using evolutionary strategies is as follows: initially, individuals sample different locations in the state space while maintaining a relatively high value of the s parameters (thanks to which mutations favour broad exploration of the space). However, when individuals are close to the maximum (global or isolated local), their s parameters decrease rapidly, sometimes by many orders of magnitude. This behaviour of the parameters causes mutations to become increasingly subtle and to approximate the maximum value more and more accurately. It turns out that the phenomenon of self-adaptation is responsible for this 'intelligent' behaviour of the s parameters. Self-adaptation

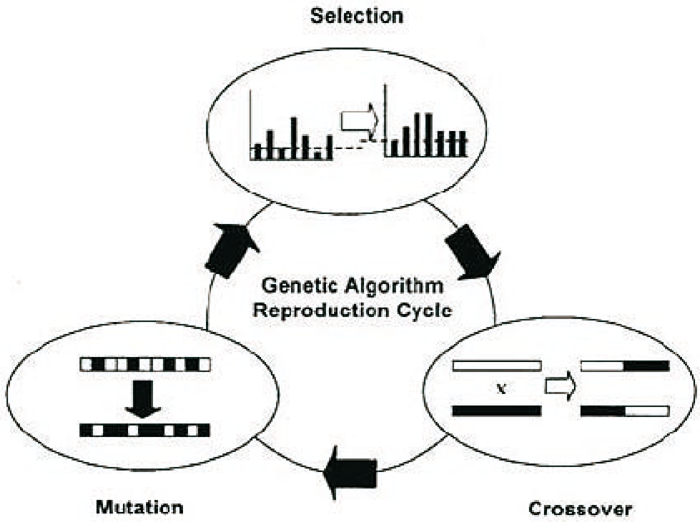


Fig. 19. Scheme of operation of the genetic algorithm; source: own elaboration

is the driving force behind evolutionary strategies. This term refers to a situation in which not only the parameters of the solution to a problem evolve, but also the parameters of the evolutionary process itself. Potentially, this allows evolutionary methods to work more efficiently [D.E.Goldberg 2009].

Evolutionary computational techniques were already known in fields where optimisation and searching for alternative solutions to the best solutions were involved. The combination of another generative algorithm with an evolutionary or genetic algorithm has made such programmes an attractive design tool for architects.

Architectural interpretation

Synthetic evolutionary algorithms that imitate biological evolution are developed mainly to solve multi-criteria problems. The objectives are then defined as fitness functions, and evolutionary mechanisms of selection, inheritance, reproduction and mutation are used as stochastic optimisation processes. These metaheuristic algorithms do not incorporate the latest research findings on micro- and macro-evolutionary mechanisms derived from genomics, phylogenomics and population genomics, which limits their degree of imitation. Nevertheless, the architectural design process, as in the case of natural evolution, is an open process exploring possible solutions, yet most design methodologies in architecture are based on a typological approach. The resulting limitations exclude a wide range of potentially more effective and better design options. The dynamics of biological evolution, on the other hand, suggest ways of continuously expanding

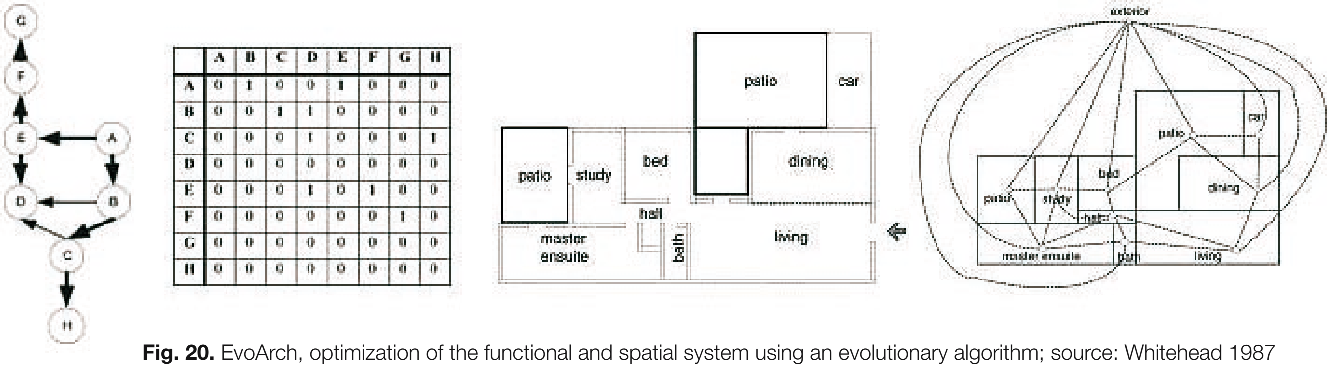


Fig. 20. EvoArch, optimization of the functional and spatial system using an evolutionary algorithm; source: Whitehead 1987

the design space towards new and unexplored possibilities that could potentially be included in a new set of typologies, while still meeting the constraints. In architecture, therefore, evolutionary processes are more important as exploratory processes than as optimisation tools. However, attempts are being made to use synthetic evolutionary algorithms as optimisation tools for design, but in combination with other generative tools.

For example, the problem of arranging usable space, which involves finding the best functional and spatial layout within certain constraints, can be formulated as a combinatorial optimisation problem and can be solved using an evolutionary algorithm. Then, spaces with a given function and adjacent to them can be represented in the form of graphs and adjacency matrices in connection with their coding scheme (Fig. 20). An adjacency matrix is a square matrix with a degree equal to the number of vertices in the graph. Each vertex of the graph is indexed by one of the consecutive numbers from 0 to N-1, where N determines the number of vertices of a given graph. In the case of simple graphs, the adjacency matrix is a zero-one matrix with zeros on the main diagonal. For undirected graphs, the adjacency matrix is by definition symmetric. The EvoArch programme encodes the topological configuration in adjacency matrices [S.Y. Wong et al. 2009], and reproduction operators operate on these matrices. The operators are designed to be unbiased, so that all nodes in the graph have an equal chance of being selected for replacement or mutation. To evaluate the usefulness of the results, EvoArch uses a matching function that takes into account preferences for adjacency between different functional spaces, budget, and other design constraints.

An evolutionary algorithm can also be helpful in the process of finding a form when the starting point is not a vision of the form, but an understanding of the multi-criteria requirements placed on the designed form and the tasks it performs. In this case, the requirements and possibilities can be defined as, for example, structural, environmental, acoustic or other criteria. The design may focus on one of these criteria, but it may also be based on a combination of different criteria in an effort to meet them. As a result, the final form, its geometry and materiality are a representation of the previously defined requirements and possibilities.

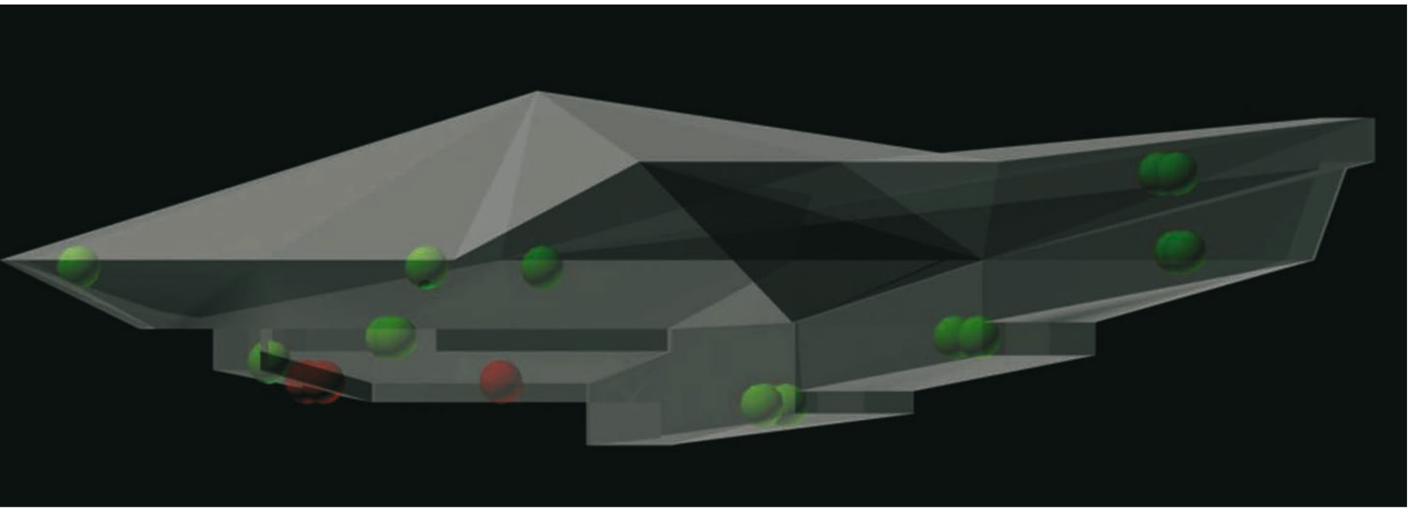


Fig. 21. Model of the Berlin Philharmonic concert hall and the configurations of the sound source (red) and receiver (green); source: Spaeth & Menges 2011

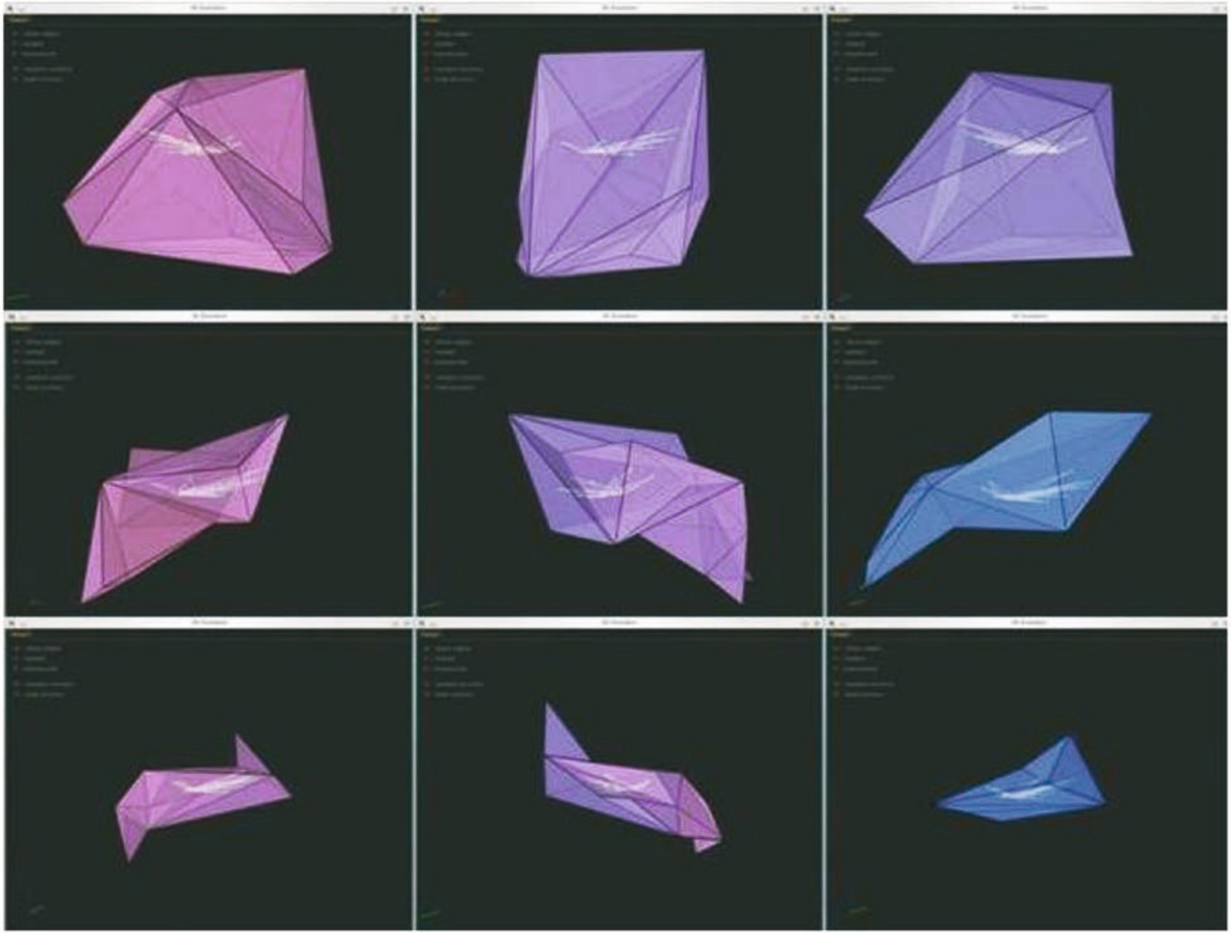


Fig. 22. Evolutionary algorithm in modelling the acoustic envelope of the Berlin Philharmonic concert hall; source: Spaeth & Menges 2011

The final design is the result of a bottom-up process, the essence of which lies not in 'creating form' but in 'finding form'. The evolutionary algorithm makes it possible to explore the area of potential design outcomes that meet the required parameters or criteria and to develop previously unforeseen geometric, structural and material solutions.

The acoustic envelope design at the Berlin Philharmonic Hall is an excellent example of this. The starting point for the evolutionary algorithm is to define the design task that the algorithm should solve. In this case, the design task could be described as finding an acoustic envelope that works on a receiver-source principle and will configure the sound in the space between the audience and the orchestra (Fig. 21). This is the initial and key design specification, defined by the designer and constituting the main design decision, apart from the specific acoustic requirements. In addition to these basic design decisions, contextual constraints resulting from the specific nature of the design task were also taken into account. [B.A. Spaeth & A.Menges 2011, p. 462].

It is evident that the audience, represented by the receivers, must have an uninterrupted view of the source (the orchestra) and that the audience should be located within this buffer zone. In addition to these constraints and requirements, the design task must be defined by morphological definitions. Clear morphological definitions guide the design process from a universal search process (driven solely by acoustic parameters) to a process that is also influenced by individual and intuitive design intentions. By defining and applying spatial morphological parameters as evaluation criteria in the design process, manual evaluation and selection were rejected due to disadvantages such as the limited speed and efficiency of evaluators or changes in evaluation criteria. To enable the analysis of acoustic capabilities, the properties related to acoustic behaviour were defined in the genetic code. However, the properties of the material that are relevant to acoustic properties (absorption and scattering) were described as values between 0 and 1. To maintain the homogeneity of the genetic code structure, absorption and scattering values were assigned to each point. Thus, the genetic

Fig. 23. Dominique Perrault, underground metro station covering, Piazza Garibaldi, Naples, 2007–2013; source: Bollinger 2008

code consists of groups, and each group contains the coordinates of a point, the absorption value and the dispersion value. Due to the uniformity of the genome, the material properties were assigned to points contained in the genome. However, due to acoustic analyses, these properties must be assigned to a surface (phenotype). The synthesis algorithm then transforms the point properties into surface properties. In this way, the surface (phenotype) is represented by three points of the genotype and can be generated (Fig. 22) [B.A. Spaeth & A. Menges 2011, p. 463].

The evaluation of generated phenotypes is a key aspect of the evolutionary algorithm. Individuals are selected for reproduction based on their fitness ranking. In most cases, the fitness value consists of several different evaluation criteria. In the evolutionary algorithm, these criteria are transformed into a single phenotypic fitness value for each individual, which can be written as * f (x). This is a weighted function of the criteria evaluation. The evaluation criteria were divided into two groups. The first group included acoustic criteria, and the second group included criteria related to the morphology of space. The simulations and tests carried out confirmed the validity of the design methodology adopted. The evolutionary algorithm used was composed of a geometry algorithm and an evaluation algorithm. Thanks to this, the geometry algorithm was able to create correct shapes and assign individual acoustic properties based on the initial configuration (source-receiver). The shapes created are therefore not arbitrary, but refer to the initial definition of the design task. It has been demonstrated that evolutionary algorithms have the potential to discover new design concepts for acoustic spaces.

It should be emphasised that the design effectiveness of evolutionary strategies depends on fitness ranking, as selection is the only control mechanism through which further project development can be directed. In nature, individual fitness is assessed at the phenotypic level in terms of the probability of reproduction. As in IT processes, each structure must be fully defined and modelled in order to be evaluated. Each evolved structure is based on the genetic information of the previous generation. Therefore, the definition of fitness criteria is important for the quality of the object and its construction, as it controls the direction of the evolutionary process [K. Bollinger et al. 2008, p. 25]. The genetic algorithms developed by Holland in the FROTRAN programming language were among the first optimisation IT tools to support calculations in structural design and are still in widespread use today. This is because computational processes enable the generation and evaluation of many possible structural articulations. In architecture, structural articulation often determines the aesthetic properties and impact of a building.

An example of this is the canopy (170 x 35 m) over the Piazza Garibaldi underground station in Naples. It consists of eight irregular rod segments which, like trees, grow out of the station platforms and branch out to reach the height of the surrounding trees (Fig. 23). Genetic algorithms helped to solve structural and optimisation problems. Entire populations of structures evolved and individuals were selected according to predefined criteria of architectural and structural suitability. The topologically designed structure could be described as a two-dimensional plane based on a system of self-

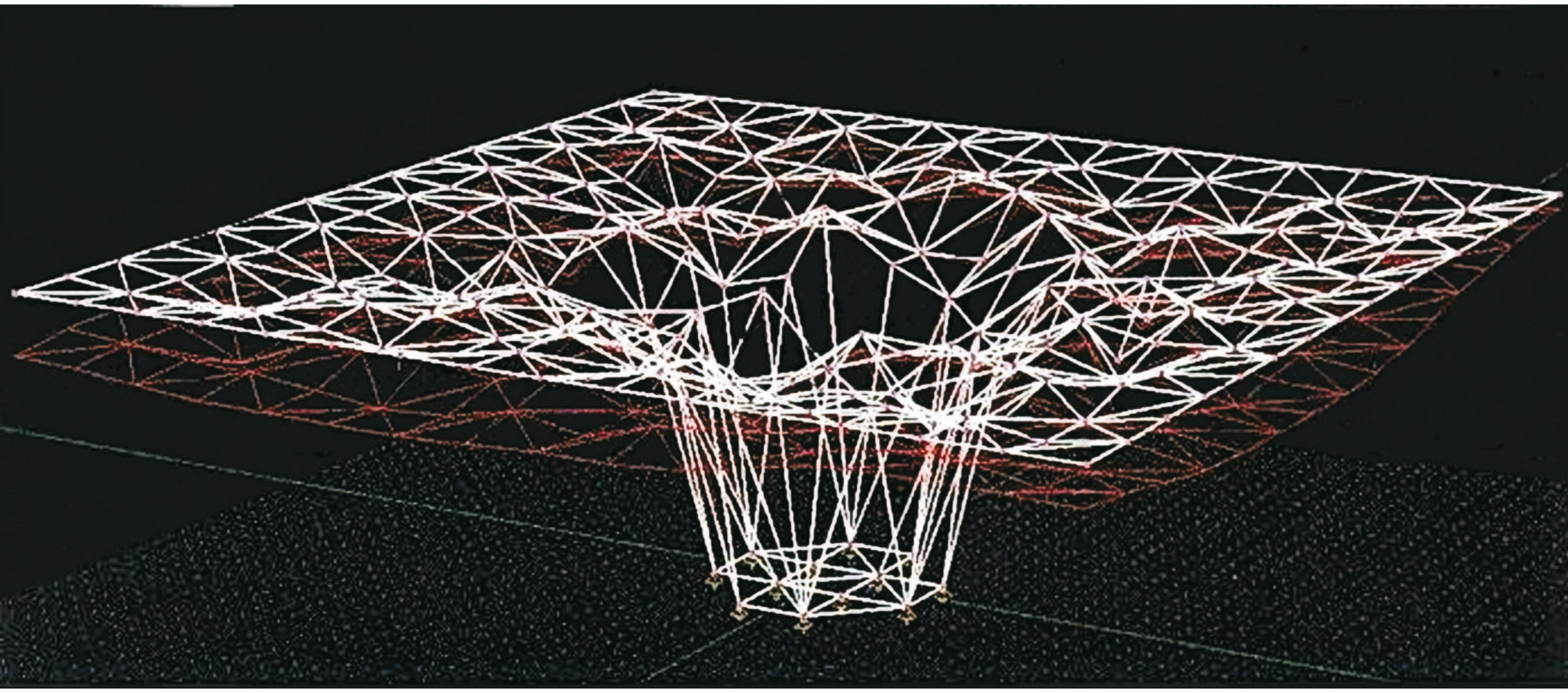


Fig. 24. Analysis of the load-bearing behavior of a segment of the roof bar structure after its generation in the evolutionary process; source: Bollinger et al. 2008

similar triangles in the third dimension. Each node was assigned a random coordinate within specific threshold value ranges. It was assumed that the bundle of rods reaching the level of the platforms acts as a supporting structure. In order to achieve cantilever load-bearing capacity and minimal displacement of the nodes, the upper plane was simply removed so that the triangulation could function as a spatial structure. The behaviour of the entire structure was simulated in RStab software (Fig. 24). The implementation design for the Piazza Garibaldi metro station roof in Naples was carried out by the Bollinger + Grohmann structural engineering office in Munich in collaboration with Fabian Scheurer from ETH Zurich, who specialises in the application of genetic algorithms in structural engineering.

By encoding the coordinates of all nodes in the genome and using a genetic algorithm that takes into account crossover and mutation, the efficiency of the structure was significantly improved over 200 generations of 40 individuals each. The criterion for efficiency was the displacement of nodes under their own weight. The genetic algorithm allowed the articulations of the structure to evolve in response to specific criteria without reverting to specific *a priori* typologies [K. Bollinger et al. 2008, p. 23].

Genetic algorithms, together with evolutionary computational techniques, are already being used in architectural design to solve complex functional, structural and formal problems (e.g. shape optimisation). They are helpful in solving well-defined construction problems described by structural, acoustic, mechanical, thermal and lighting parameters, etc. By acting on a population of achievable variants, they search for new and unexpected design proposals. The pressure of the effects of progressive climate change directs designers' attention towards architecture with the ability to adapt to changing environmental conditions. It is possible that evolutionary and genetic algorithms could search for adaptive possibilities and find the right solution, providing a method for effectively preserving kinetic architectural structures. Such structures, like plants or organisms, would need to have the efficiency that flexibility provides. However, in the traditional engineering approach to structural design, buildings achieve efficiency through rigidity. This is confirmed by analyses carried out after the collapse of the World Trade Centre towers, which showed that it is necessary to design tall buildings whose behaviour will result from the flexibility and elasticity of their structure [P. Bažant & Z. Zhou, p. 607]. In the case of this type of disaster, several forces combine and interact with each other, creating a non-linear acceleration of effects. The criteria for performance therefore need to be reconsidered, combining the need for flexibility with sufficient rigidity to ensure stability. In human artefacts and natural structures, there are models of surface structures that demonstrate the ability to bend without collapsing. Woven structures, such as baskets, can accept several local disturbances without global destruction. Baskets are highly redundant, as they have more material than is necessary to carry normal loads and have no rigid joints. All living structures have a high degree of redundancy, which enables them to adapt. The nodes of their structures differ significantly from engineering connections. Introducing these strategies into the design of tall buildings will radically change their performance and improve safety.

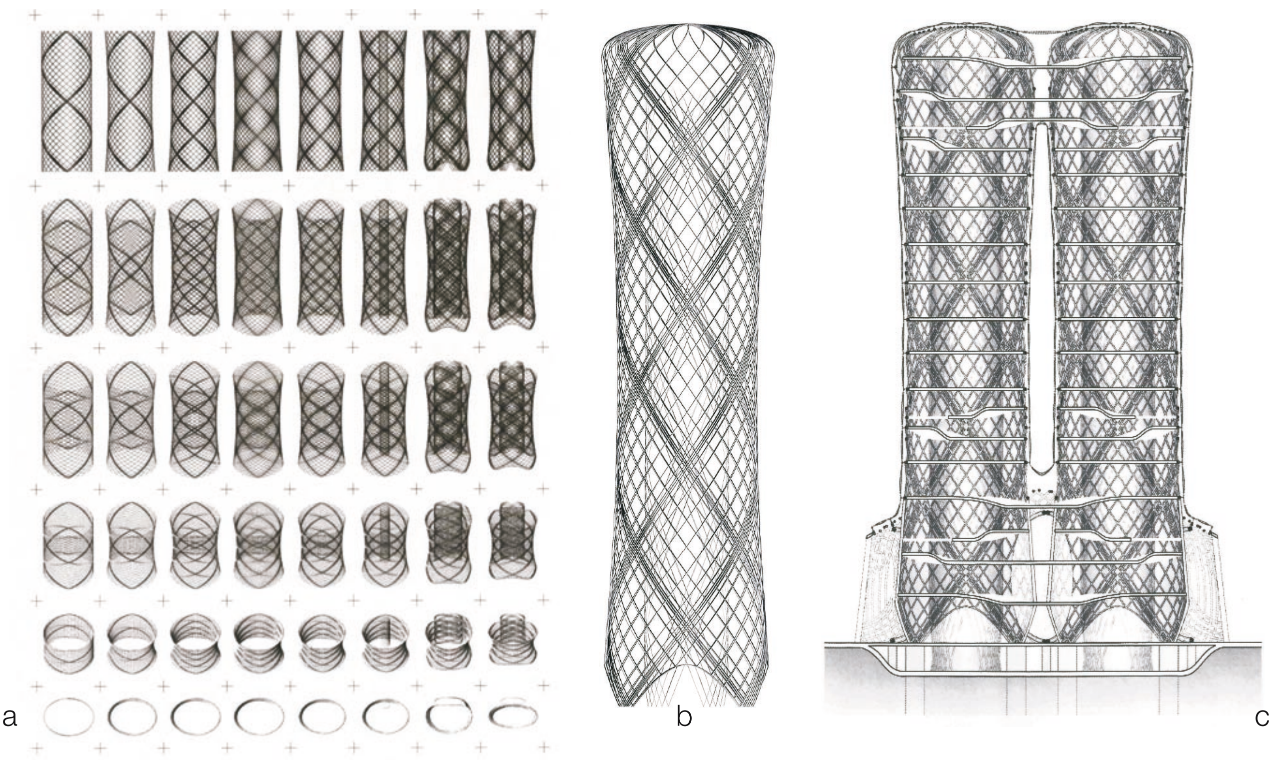


Fig. 25. Emergence and Design Group, reconceptualization of WTC skyscrapers, 2003 – evolutionary development of genotype, a) evolution of circle geometry into cylindrical helices, b) skyscraper form selected for phenotype development, c) cross-section; source: Weinstock et al. 2004

Thanks to biological evolution, there are structures in nature that have a cylindrical morphology and exhibit structural flexibility. These properties result from the structure of their skin, which has no ribs or columns. Natural evolution has provided several successful strategies for surface structures, including optimised shape morphology and the arrangement of components in complex hierarchical systems to provide a path for multiple force vectors resulting from loading. The Emergence and Design Group is investigating the possibility of developing such surface structures on an architectural scale, with structures that achieve their coherence by organising simple components into diverse bundles and weaves. The first results of this research were presented in projects inspired by the debate on the competition for the reconstruction of the WTC twin towers in New York.

The design of the new WTC towers began with consideration of the geometric aspect. Geometry is the essence of both natural and computational morphogenesis [M. Weinstock et al. 2004, p. 40]. It provides data on boundary constraints that inform the global configuration of the developed form, as well as local rules and principles of organisation and self-organisation during morphogenesis. Preliminary research on general patterns in natural systems suggested that a helix should be chosen for this evolutionary experiment. In the physical world, helical spirals occur in dynamic configurations at all scales. In living organisms, helical spirals are found in the arrangement of protein molecules in DNA and in the geometry of pine cones, sunflowers, and broccoli. Xylem vessels in plants are slender tubes that transport water and soluble substances from the roots to the stem and leaves. Spiral bands of lignin reinforce the xylem, and the spiral geometry allows the tubes to elongate and grow. In artificial systems, spiral steel helices could replace load-bearing structures, providing tall buildings with stability and flexibility.

A steel pipe with a cross-section of 150 mm was selected for work on the new support system, whose bundles will mimic xylem vessels in biological systems, and whose cross-section parameters will provide the basis for developing a genotype in an artificial evolutionary process. Imitating biological evolution requires that the ‘seed’ (input) be sown first, i.e. the data on the cross-section of the steel pipe, which will develop in the information space. This space is a mathematical ‘environment’ defined by the urban planning restrictions contained in the WTC competition conditions regarding the location and construction of high-rise buildings. A single component obtained on the basis of the pipe cross-section parameters rotated around the original centre allowed four sets of copies to be generated. This resulted in the first generation, i.e. a 60-metre double helix consisting of 10 pipe sections arranged in two bundles. These bundles were developed by generating an internal counter-rotating layer of pipes. To obtain the next generation, four sets were again selected, each with its outer and inner group of components, which developed in opposite rotations. In subsequent generations, the inner and outer layers of rods were arranged in quadrants between the bundles. At this stage of evolution, the structure consists of 80 elements arranged in two concentric continuous layers. After simulating the effects of gravitational forces on the global geometry, a population of diverse forms was obtained, from which one form with an extended base and top and slightly narrowed in the middle was selected (Fig. 25b).

In architecture, there are no genes responsible for the characteristics of a building. However, each building has a set of information that can be extracted. It is in the structure of the building that instructions (behaviour patterns) are stored. This allows the building to respond appropriately to environmental situations.

The development of the genotype continued after the removal of the geometric principle of parallelism of structural surfaces adopted for the inner and outer layers of the helices. This resulted in more complex geometric relationships between the surfaces, which evolved into curved surfaces with uneven distances between them, and even intersections between individuals appeared. The components thus moved to a higher level of structural organisation, in which microstructures intersect, wrap around, connect and disconnect. This complexity suggests that the phenotype can force new spatial organisation and structural possibilities. For example, floor slabs will be able to be three-dimensionally articulated and less symmetrical than in conventional tower structures, and may even be volumetric [M. Weinstock et al. 2004, pp. 43–44].

The development of the phenotype was triggered by exposing the geometry to virtual environmental forces. During this process, twin forms are created, their number increases, and aggregation occurs. These forms increase structural capacity through load sharing and distribution – not specialisation, but variability within a single geometry population. The shell ('skin') of the structure was developed based on digital research and finite element analysis of the mosaic geometry of the surface of the sugar apple (Annona reticulata) fruit (Fig. 26a). The skin of this fruit must maintain its structural integrity, resistant to the pressure of the swelling material inside during ripening. All panels have the same shape, but their size varies, and the

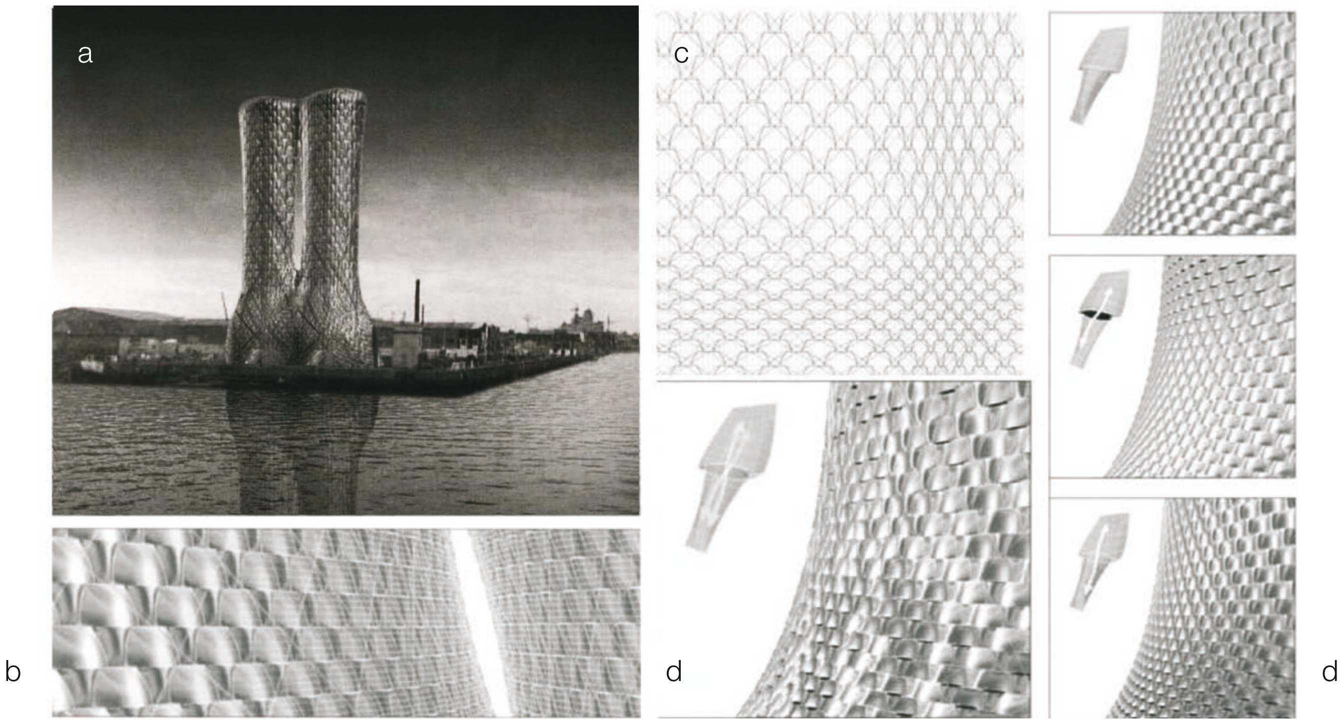


Fig. 26. Emergence and Design Group, reconceptualization of WT skyscrapers, 2003 – development of phenotype a) general view, b) pneumatic active kinetic facade, c) structural mesh of kinetic cushions, d) scheme of air exchange in the building; source: M.Weinstock et al. 2004

tessellation results in a surprisingly small number of changes required for complex double curvatures [Weinstock et al. 2004, pp. 43–44].

It has been assumed that the building envelope is an integral structural system for panels, which act as environmental regulators due to their efficient adaptive capabilities. The differentiation of panel geometry follows a logic similar to that of helices – they all have the same form and geometric logic, but their size varies through a limited number of parametric changes. A few parameter changes allow the panel shape to adapt to the changing curvature and variable shape of the twisted structure using a simple algorithm. The organisation of the structural interface, the connection between the helices and the panel regions, is local. This maintains consistency between the different geometric hierarchies of the components and has the ability to adapt to global changes in geometry.

Similar to the evolutionary phenotype, the architectural phenotype conveys all the physical characteristics of a building. The phenotype is not limited to morphological characteristics, but also includes physiological characteristics (e.g. heat transfer coefficient) and functional characteristics (e.g. comfort).

The building's façade has been designed as a three-layer kinetic membrane activated by pressure differences in the capillary system of pneumatic actuator cells, which are located between the inner, middle and outer membranes. Pressure differences in the capillary layers cause a change in geometry from convex to concave. Alternating changes in the geometry of the lower and upper halves of the panel regulate air ventilation and direct light transmission (Fig. 26b–c). The photovoltaic cells printed on the membranes will store and manage the energy needed to maintain this system locally. There is therefore no need for a central energy source, which increases the reliability and efficiency of the system and reduces production and maintenance costs [M. Weinstock et al. 2004, p. 44]. The building's façade has been designed as a three-layer kinetic membrane activated by pressure differences in the capillary system of pneumatic actuator cells, which are located between the inner, middle and outer membranes. Pressure differences in the capillary layers cause a change in geometry from convex to concave. Alternating changes in the geometry of the lower and upper halves of the panel regulate air ventilation and direct light transmission (Fig. 26b–c). The photovoltaic cells printed on the membranes will store and manage the energy needed to maintain this system locally. There is therefore no need for a central energy source, which increases the reliability and efficiency of the system and reduces production and maintenance costs.

The examples presented demonstrate that evolutionary algorithms and genetic algorithms have promising applications in architecture. However, their use requires the removal of limitations inherent in typology-based design methodologies, which exclude a wide range of potentially more effective and better design options. The dynamics of biological evolution suggest ways of continuously expanding the design space towards new and unexplored possibilities that could potentially be included in a new set of typologies that would still meet existing constraints. Thus, in architecture, evolutionary processes are more relevant as exploratory processes than as optimisation tools.

4.4. Swarm intelligence (SI) and agent-based modelling (ABM)

Agent systems, also known as swarm intelligence, is one of the techniques of computational intelligence[1] [J. Kennedy et al. 2001]. It fits into the broader concept of multi-agent systems [M. Wooldridge 2001]. Such systems, which are distributed by design, solve the task at hand by utilising the cooperation and communication of agents – individuals placed and operating in a certain environment, endowed with the ability to monitor their own condition and that of their surroundings, memory, communication and decision-making. Swarm intelligence, whose history dates back to the 1990s, is based on modelling the social behaviour of specific species of living organisms, with individuals of a given species serving as prototypes for agents. This technique was created by simulating the collective intelligent behaviour of groups of insects or animals, such as flocks of birds, colonies of ants, schools of fish, and swarms of bees [P. Coates 2000]. The modelling process is usually simplified, and the agent is characterised by a small number of strictly utilitarian and mathematically primitive properties and abilities.

Swarm intelligence systems exhibit characteristic mechanisms such as herd behaviour, self-organisation of populations and decentralisation of decision-making centres. These mechanisms give the group's decisions

[1] *Computational intelligence* is a rapidly developing branch of technical sciences. It is a subset of artificial intelligence, and the mathematical apparatus used belongs to the field of soft computing. See: J. Kennedy, R. Eberhart, Y. Shi, *Swarm Intelligence*, Morgan Kaufmann Publishers, 2001.

a global character, leaving the agent with localisation and simplicity of decision-making. Millonas' research has identified five characteristics specific to swarm intelligence systems:

- *proximity principle* – an agent's field of perception is strictly limited;
- *quality principle* – agents are guided by an assessment of adaptation to the environment;
- *principle of diverse response* – the agent's response is clearly directed, but not deterministic (element of randomness);
- *principle of stability* – the agent and the population are characterized by a certain inertia in action;
- *principle of adaptability* – the population reacts to the appearance of changes in the environment [M.M. Millonas 1994].

The main and most widespread metaheuristics of swarm intelligence are two algorithms called ACO: *ant colony optimization* [M. Dorigo & T. Stützle 2004] and PSO, *particle swarm optimization* [J. Kennedy et al. 2001].

Ant algorithms are based on earlier research into the mechanisms of organisation and communication of experiences in ant and termite colonies using stigmergy – leaving chemical traces (pheromones) in the environment. On this basis, a number of graph-based metaheuristics variants have been formulated, in which agents – virtual ants – based on primitive perceptual abilities, complete solutions to optimisation problems, perform a simple evaluation of them, and then update the degree of pheromone saturation of the components of the achieved solution. PSO techniques dating back to 1995 mimic selected aspects of the herd behaviour exhibited by certain animal species (e.g. schools of fish, flocks of birds, swarms of insects, etc.). They place a swarm of particle points in a multidimensional problem feature space, assign them random initial velocities, and then, using (a) particle inertia elements, (b) individual memory mechanisms, and (c) social swarm mechanisms, set the individuals in motion to search for the global optimum for the adopted objective function. These rules, together with the adopted neighbourhood relations within the community, make PSO well suited to Millonas' five principles [M.M. Millonas 1994].

In 1989, swarm intelligence (SI) was first introduced in robotic systems, where it describes emerging collective behaviours. Contemporary applications also concern the behaviours of human communities.

Among other swarm intelligence techniques that became widespread in later years, the following are worth mentioning: ABC, *artificial bee colony* [D. Karaboga & B. Basturk 2008] or BA, *bat algorithm* [X.S. Yang 2019], but also systems modelling selected bodily functions, e.g. AIS, *artificial immune systems* [J.D. Farmer 1986] or those based on group behaviour [L.B. Rosenberg 2015].

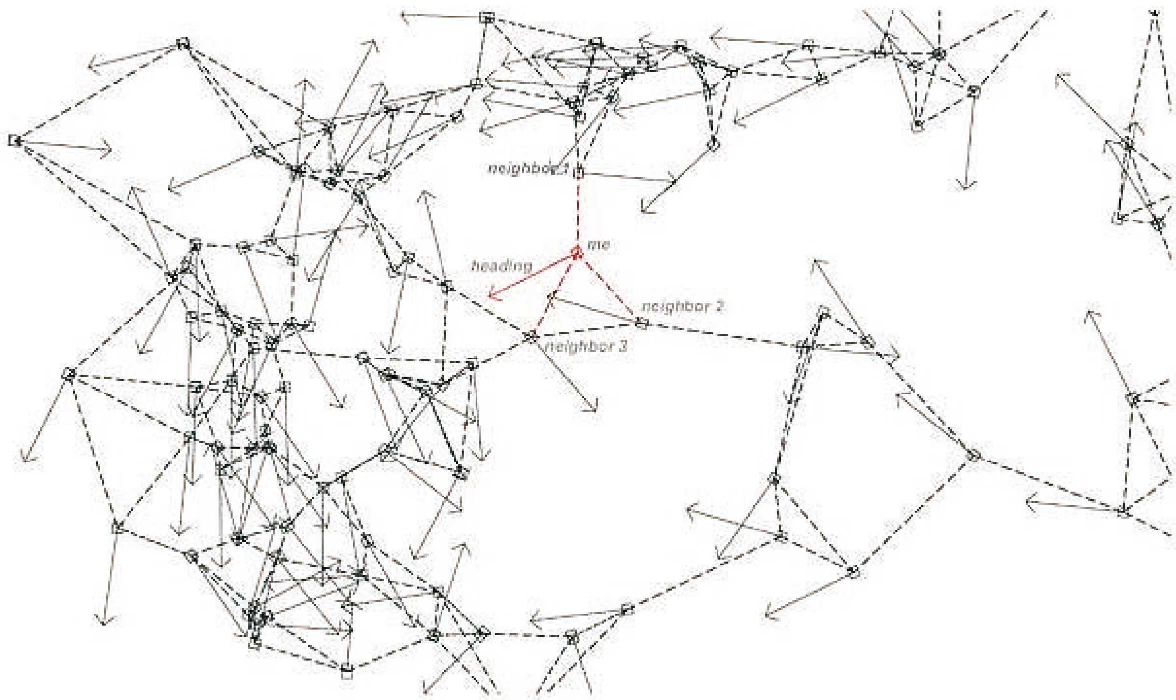


Fig. 27. Swarm diagram. The arrows represent the index of each agent, the dashed lines their nearest neighbors; source: Karaboga & Basturk 2008

Algorithms based on swarm intelligence are characterised by high flexibility in application in almost all fields of science and high effectiveness in solving complex problems for which classical methods do not provide an effective solution. Computational intelligence is used wherever automatic inference is needed and the amount of data (premises) makes it difficult to make decisions or formulate strictly deterministic relationships between phenomena and their consequences. Choosing the right algorithm is a key step in the problem-solving process. Algorithms based on the structure of computational evolution (CE) and swarm intelligence (SI) are often classified as metaheuristic algorithms and are widely used in engineering problems.

Algorithms based on swarm intelligence are characterised by high flexibility in application in almost all fields of science and high effectiveness in solving complex problems for which classical methods do not provide an effective solution. Computational intelligence is used wherever automatic inference is needed and the amount of data (premises) makes it difficult to make decisions or formulate strictly deterministic relationships between phenomena and their consequences. Choosing the right algorithm is a key step in the problem-solving process. Algorithms based on the structure of computational evolution (CE) and swarm intelligence (SI) are often classified as metaheuristic algorithms and are widely used in engineering problems.

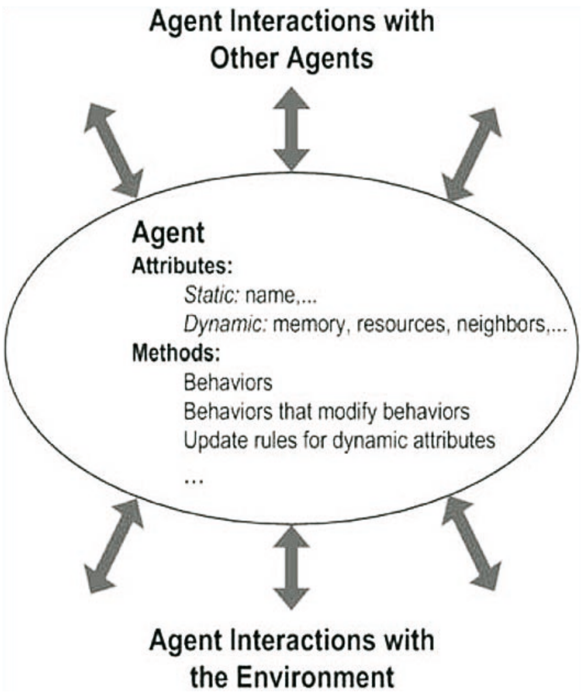


Fig. 28. Typical agent – diagram algorithm; source: Ligmann-Zielinska & Sun 2010

The process of creating Swarm Intelligence uses a population of individuals with a constant ability (size) to search across generations, and in each generation, the results of individual individuals are evaluated in order to adjust the search strategy for the next generation without any selection operations on individuals. This is the main difference between evolutionary computational methods (EC) and swarm intelligence methods [J. Kennedy et al. 2001]. Self-organisation strategies and independent work by each individual to solve problems are two important features of swarm intelligence methods. The self-organisation strategy leads to the creation of a system of units that respond individually to local stimulation and can work together to perform a global task; and an independent 'working group' leads to the avoidance of centralised supervision [J. Kennedy et al. 2001]. It is precisely these characteristics that enable the simulation of the collective behaviour of groups of insects or animals in nature [X.S. Yang & M. Karamanoglu 2013, p. 12].

Agent-based modelling and simulation (ABMS) is a relatively new approach to modelling complex systems consisting of interacting, autonomous "agents" [P. Coates & R. Thum 2000]. Agent models are digital representations of systems composed of elements and objects located in a shared environment. Agents interact with each other and their environment and strive to achieve their intended goals. They are able to make decisions about what actions to take in order to achieve their goals [A.Ligmann-Zielinska & L. Sun 2010]. They also have decision rules that drive their actions. Agents have behaviours described by simple rules and interactions with other agents, which in turn influence their behaviours. By modelling each agent individually, a full diversity of agents in terms of their attributes and behaviours is observed, as this results in the behaviour of the system as a whole.

Agents can represent animate and inanimate elements. They can also be grouped into larger units in the model and can be mobile. They can also have a simplified structure, in which case they are called weak agents, or be designed in such a way that during simulation they will gain experience and learn using artificial intelligence. In this case, they are referred to as strong agents in the model. When modelling systems from scratch – agent by agent and interaction by interaction – self-organisation can often be observed in such models. Patterns, structures and behaviours emerge that have not been explicitly programmed into the models, but arise as a result of agent interaction. Agent systems are a way of modelling complex systems. The agent that forms the basis of the system is an element endowed with characteristics that determine its behaviour. These characteristics can be both simple decision-making processes (e.g. if x is less than a given distance from y, then...) and more complex adaptive processes of artificial intelligence, adapting to a given context at a selected moment in the simulation. The nature of agents implies their active character, as opposed to the passive character of PSO particle models in the case of physical phenomenon simulations. The agent makes decisions using only the information it is able to 'observe', i.e. only local data (e.g. distance to the boundary) influences the agent. Thanks to this assumption, emergent phenomena can be observed, but this is only possible when operating with an entire 'swarm' of agents [P. Coates & R. Thum 2000].

Swarm intelligence (SI) as a branch of science is in its early stages of development, which is proceeding in a scattered manner. Among its most common applications are optimisation issues, data clustering, classification, and assistance in learning heuristic tools. Swarm intelligence algorithms appear sporadically in image processing as auxiliary procedures, e.g. in classification or decision support, often in combination with other computational intelligence techniques.

Architectural interpretation

The spectrum of applications for Swarm Intelligence (SI) systems is broad and covers many fields and disciplines. Although swarm behaviour has no direct reference to architecture and its design, its computer simulations, combined with other generative techniques, are used both to optimise and create architectural forms. They provide active support in solving complex design problems and in the search for new structural and spatial manifestations that may have practical applications.

Architects have at their disposal modelling based on swarm intelligence algorithms and agent-based simulation modelling. Both approaches can overlap in the pursuit of new, efficient architectural objects. From an architectural point of view, swarms define physical space with their presence, which can change dynamically. The complex choreography that develops through the movement of the swarm is an example of the emergence of collective behaviour. The resulting order is not imposed from above, but emerges from the bottom-up interaction of agents, leading to a range of generative strategies in architectural design.

Flocks of birds and schools of fish can be simulated using digital animation tools and then replicated using various types of elements. It is also possible to visually test what kind of space can be achieved. This has been possible since 1986, when Craig Reynolds developed the Boids programme, which simulates the behaviour of birds in flocks. (Fig. 29). As with most artificial life simulations, Boids is an example of emergent behaviour simulation; this means that the complexity of the birds' behaviour results from the interaction of individual agents (here Boids) following a set of simple rules. The rules applied in the world of Boids are three basic principles of swarm behaviour: separation, alignment and cohesion [C.W. Reynolds 1987, p. 26].

Each individual in a swarm population, treated as a point in space, determines the structure of that space, which can be formalised. By connecting these points, it is possible to test new, constantly changing relationships and interactions, influencing the structure of both the space and its components. Nature shows us that by creating closed-loop systems among large groups of independent agents, a high level of intelligence can emerge that exceeds the capabilities of individual participants.

The architectural space can also be treated as modular, as a cube controlled by eight points of a swarm. The form would change depending on the movement of these points. Simple translation, rotation and manipulation relative to the xyz axes can generate a complex transformation system. Active connections enable a simple module to construct different geometries [Y. Chen 2018, p. 63] (Fig. 31). Swarm modular can generate a basic framework for architectural form. Particles, based on simple rules, like birds or insects, will swarm around to generate form. This aspect pushes the boundaries of the designer's imagination towards unpredictable configurations of elements that

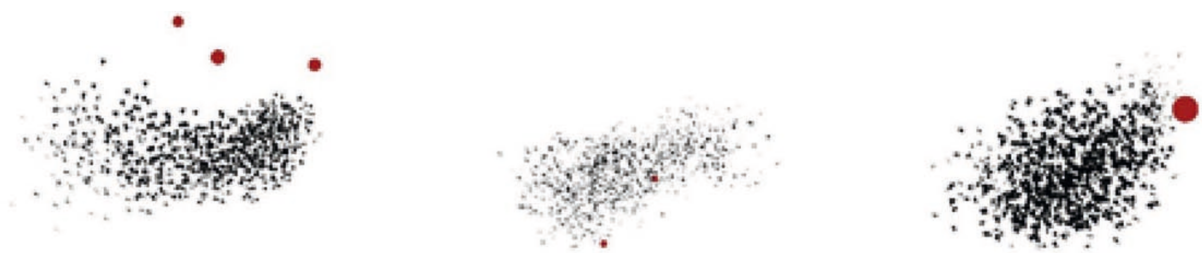


Fig. 29. 2D simulations of bird behavior made in Boilds program (red points are attractors); source: Chen 2018

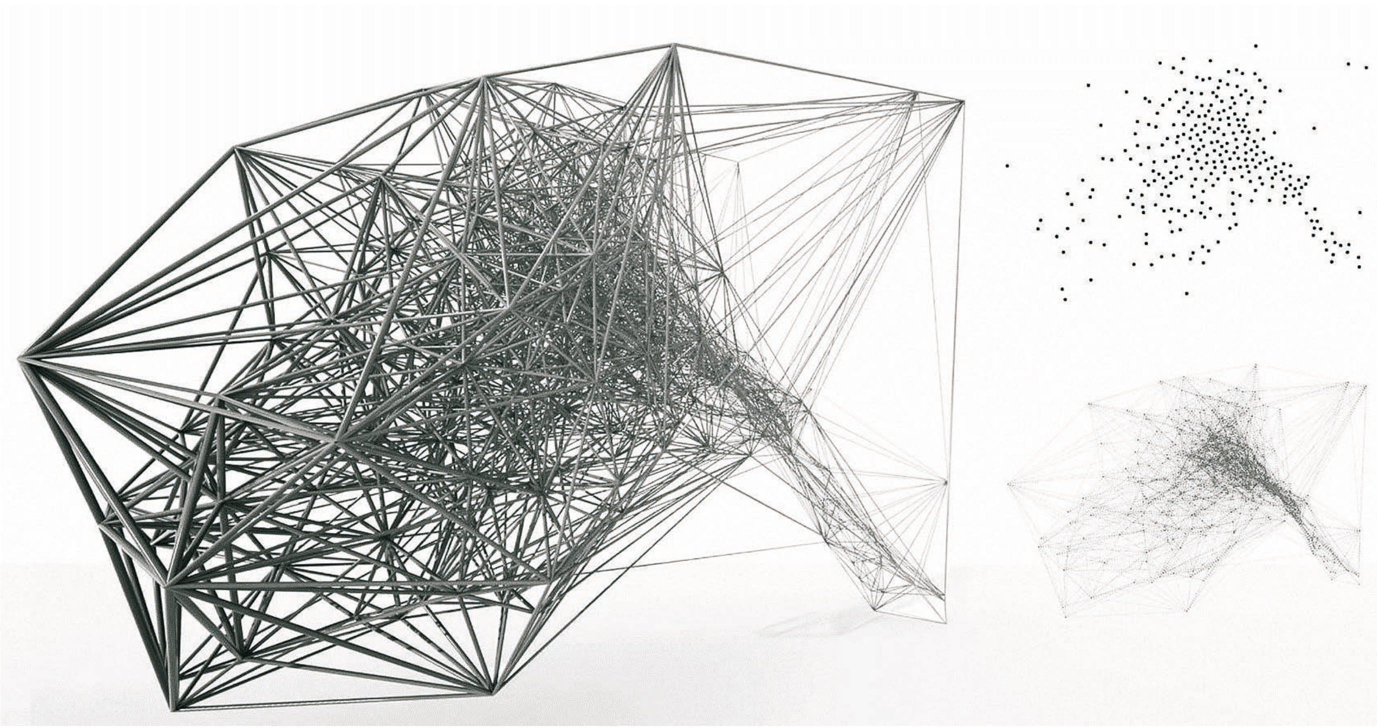


Fig. 30. Yuxing Chen, the structure of the swarm space created by imaging the inter-individual burns; source: Chen 2018

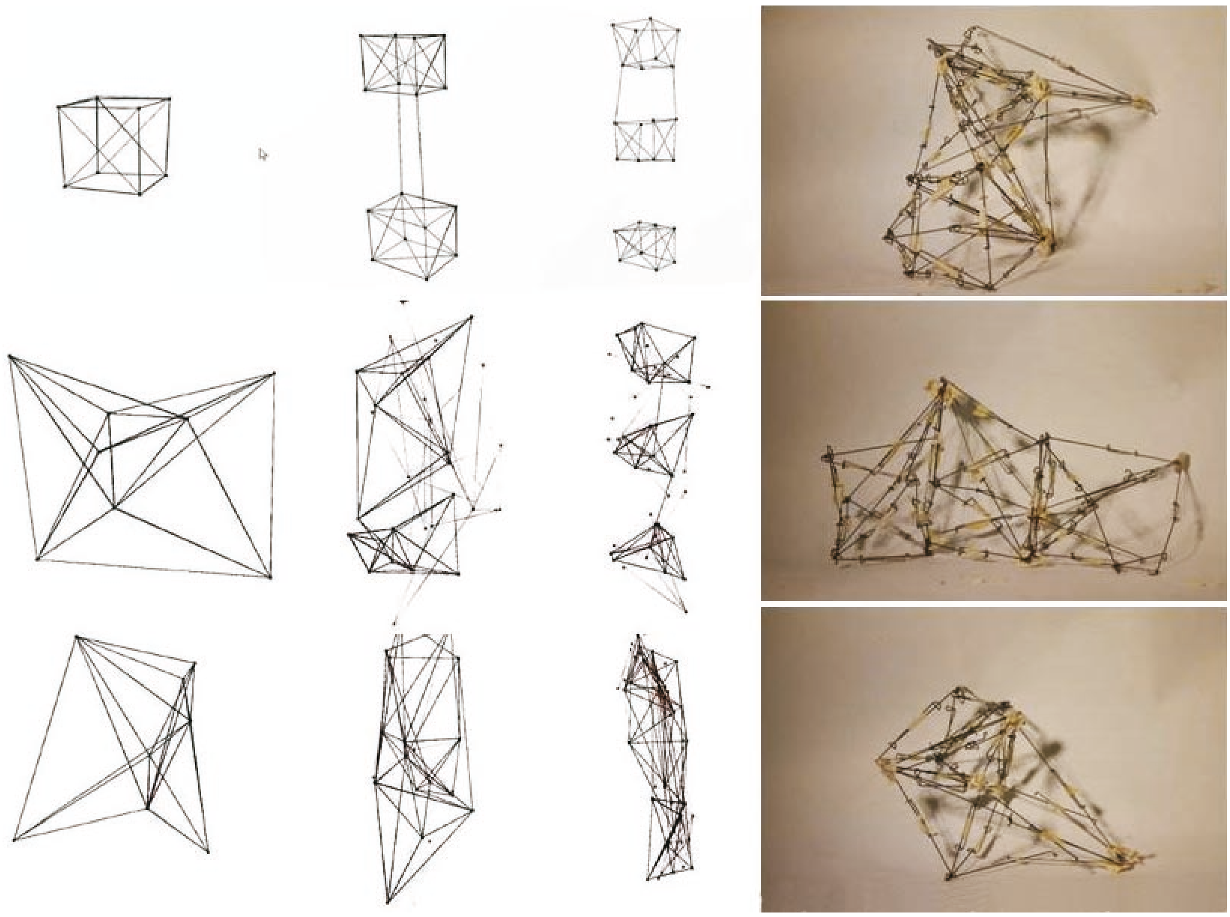


Fig. 31. Yuxing Chen, Modular Space Transformation by 8 Swarm Points; source: Chen 2018

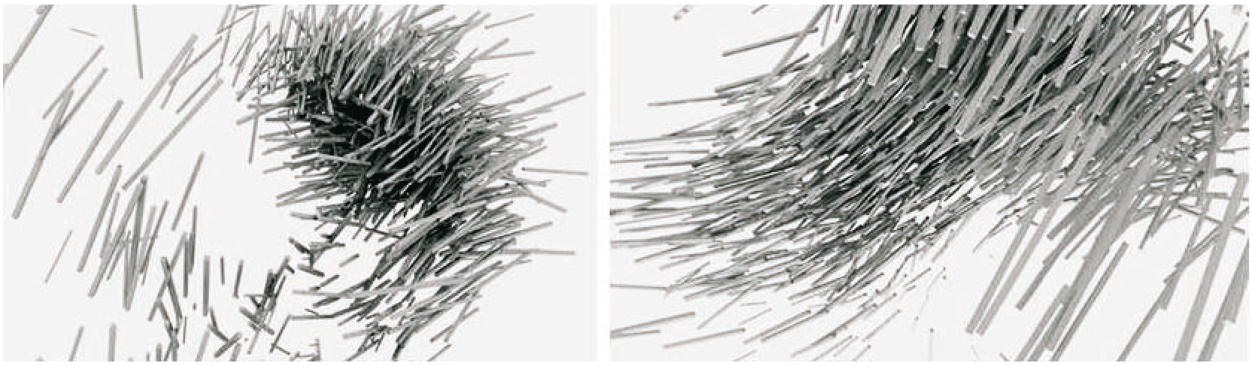


Fig. 32. Yuxing Chen, simulation of component behavior made in Boilds program; source: Chen 2018

may be useful for changing typologies in architecture and structural engineering.

If a swarm were perceived as the movement of particles, connections could be constructed through these particles or between their different paths. It would be a system similar to a structure. Then, instead of a static space defined by heavy building structures, the space created by the swarm could be constructed from natural materials, components of various sizes (Fig. 32).

The 'Uchronia' pavilion is a temporary structure built from wood waste accumulated at a Canadian sawmill. It is 15 metres high, 61 metres long and 30.5 metres wide. It was made from elements with a cross-section of 5 x 7.6 cm and a length of 2.43 to 3 metres. A total of 161 kilometres of wood and one million nails were used in its construction.

Swarm intelligence allows architects to understand the role of agents in the generative design process and the fundamental relationship between swarm behaviour, human activity, urban planning, tectonics and the nature of the universe. The aim is to study and utilise swarm intelligence to improve design efficiency on many scales.

In agent-based modelling (ABM), the environment in which agents operate is a passive element of the model. It can represent the natural environment, geographical space, and, importantly for architecture,

Fig. 33. Arne Quinze (Quize&Milan), "Uchronia" Pavilion, Burning Man Festival, Death Valley, USA, 2006; source: Chen 2018

architectural space or design space. In this case, the agent model is referred to as spatially explicit. Space, not only architectural space, is described in agent models using digital spatial data. Thus, the environment in which agents operate is the space in which agents take all actions. Depending on how the operating environment is defined, the agents' environment can be understood as a given distance from them, the neighbourhood of cells in a raster layer, or the number of snakes in a defined vector network.

The attributes and behaviours of agents can be freely changed, and the resulting consequences are analysed. This feature of agent models makes them useful for analysing problems on different spatial and temporal scales and at different levels of organisation [D.G. Brown 2006, pp. 7–13]. An agent model represents a selected system and explains how that system works. It represents the processes that occur in the environment covered by the design and allows experiments to be carried out repeatedly, using different parameters, without any damage to the system under study.

Both agents and the decisions they make usually have a spatial reference, hence the connection between ABM and architectural design seems to be a natural consequence of the methodological and methodological development of both techniques. There are many software packages and tools for creating agent models, varying in complexity, that allow for independent construction or editing of ready-made models. Depending on the user's level of advancement, knowledge and skills, programming a model can be somewhat challenging, especially in the initial stage, when many properties of the model elements need to be defined. The question remains open as to when agent-based modelling should be used in the design process, when it will be justified and useful.

An example is the method of optimising sunlight exposure in a tall building, which integrates NURBS geometry (CAD), genetic optimisation algorithms (GA) and a hierarchical agent system to control the building's geometry [K. Yi & H.Kim 2015] (Fig. 34). First, the generated random values were entered into a hierarchical system of agents (represented by four points and a central reference point) placed in a CAD model. The geometric layout of the building segments immediately changed according to the entered values, and the simulation tool indicated how many directions there were from which the sun's path would ensure the desired amount of sunlight exposure for the segments. The values obtained are then evaluated in relation to the minimum and maximum sunlight exposure during the year. Once the value of the objective function for the current position of the building segment has been found, an assessment is made as to whether this value is satisfactory. If it is not, then the geometry point agents (at the four corners) pass their values on for reproduction. The usefulness of the point agent (individually) is taken into account to decide on the method of reproduction. Typically, a point agent with better performance (better efficiency) will adapt better to the new generation than a point agent with lower performance. Part of the initial population is replaced by a new population thro-

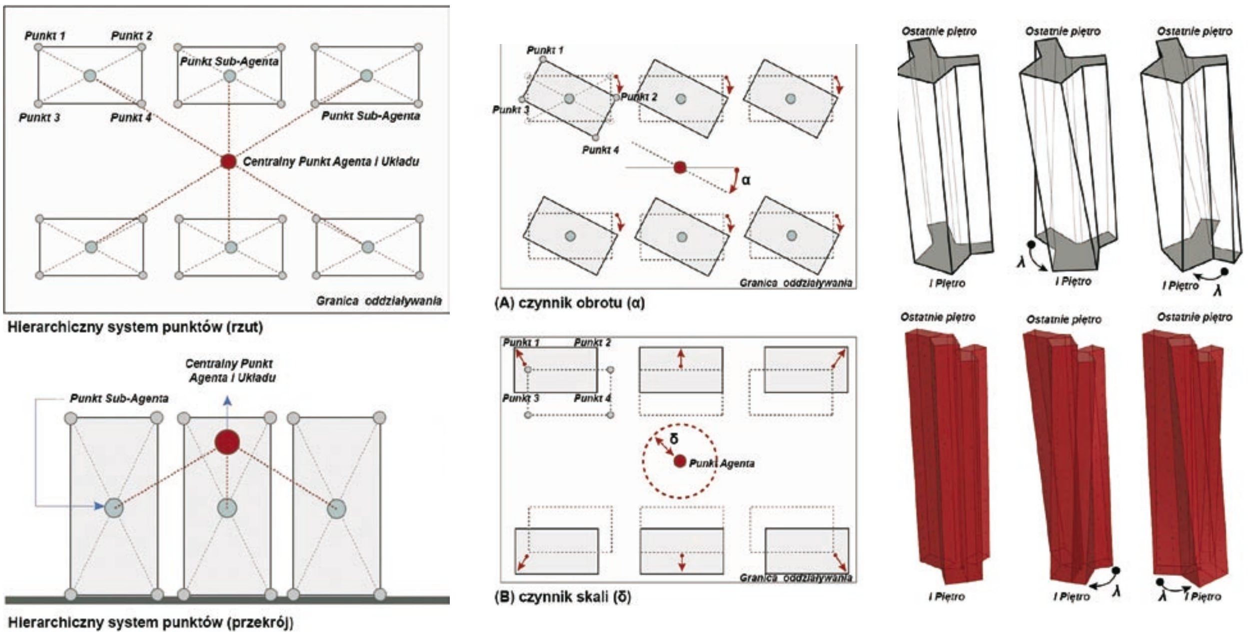


Fig. 34. Optimization of the segment layout of the object due to solar radiation using a hierarchical agent system; source: K. Yi & H. Kim 2015

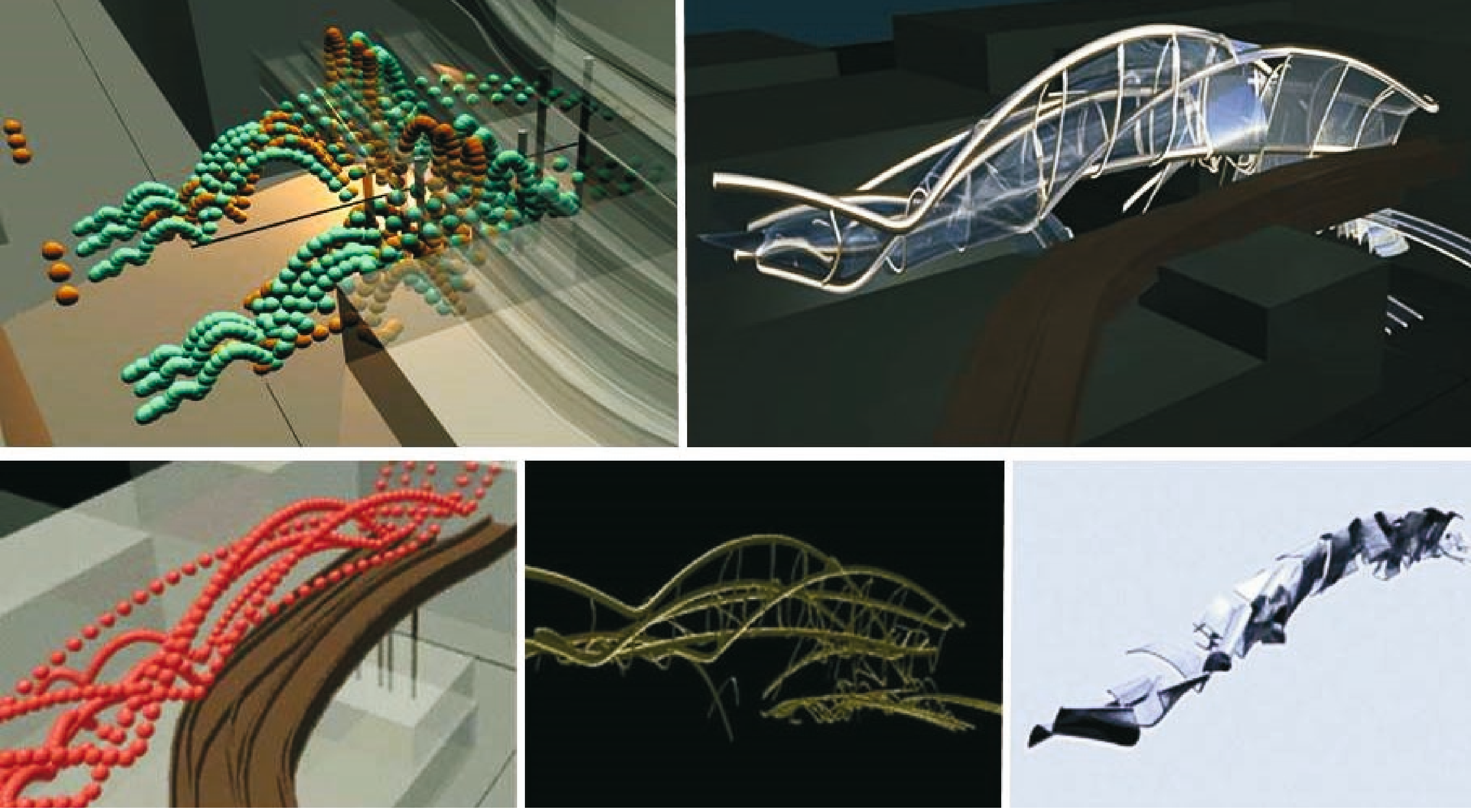


Fig. 35. Greg Lynn, Port Authority Triple Bridge Gateway,1994; source: Jencks 1995

ugh the crossing of surviving individuals and random mutations to prevent local optimisation. This process is called ‘reproduction’ of the initial population and leads to the evolution of a population that is better adapted to the environment than the previous generation. The reproduced offspring includes a new individual with the desired variable values (new agent point value), which is transferred to the building model in the CAD file. This updates the building layout model, and the simulation programme generates the next generations of values (through successive iterations), which are evaluated in terms of performance. This process will continue until the objective function value is met, i.e. values are found for the geometry that ensures optimal sunlight conditions for the building throughout the year.

Agent-based modelling has various areas of application in architecture. It is most often treated as a potential tool supporting the design process, improving the quality of functional, structural, operational and environmental solutions, as well as in the creation of innovative building systems based on the principles of self-organisation. In architectural applications, swarm intelligence offers high potential for efficiency in solving problems that are tailored to the task at hand.

The most widely used is the multi-agent design environment ‘protoSWARM’, which was developed for a long time without being given a specific name and was often referred to as a ‘swarm tool’ [T. Jaśkiewicz 2013, p. 207]. Its origins date back to the late 1980s, and its applications did not concern architecture. Nevertheless, Greg Lynn used elements of this environment in his competition design for the Port Authority Triple Bridge Gateway (1994) to simulate passenger traffic at one of New York’s transfer stations (Fig. 35).

Greg Lynn’s competition entry was the first architectural design in history to use a swarm of active particles to generate form, imaging the flow of station users and animation software. The traffic intensity values represented by these particles were rendered as spheres, while the directions of pedestrian flow were generated as ‘force lines’. Then, using Wavefront software, these spheres moved with their assigned speed index, illustrating the flow of human streams. This simulation allowed decisions to be made regarding the desired capacity and load of the designed bridge, which also influenced its shape (Fig. 35). Undoubtedly, Lynn’s experiments with particle flow contributed to the development and implementation in 2001 of the ‘protoSWARM’ design environment at UT Delft by a team led by Kas Oosterhuis.

The first version of ‘protoSWARM’ was developed in 2001–2002 for the Protospace Demo 1.2 project. Later, variants of this programme were used in several ONL projects[2]. The aim of protoSWARM was to provi-

[2] his refers in particular to projects such as: the Salzburg National Park Centre SNCP (2005), Manhal Oasis in Abu Dhabi (2006), Automotive City in Abu Dhabi (2007), Paracity, Muscle Fasade, Leafs portal and others.

de an open, extensible and easy-to-use virtual environment in which designers with diverse knowledge could programme autonomous virtual agents corresponding to the project components they were dealing with, and then develop and virtually implement complex, multi-agent systems as aggregations of these agents [T. Jaśkiewicz 2013, p. 207].

The competition design for the National Park Centre in Salzburg (2005) was initiated by a single virtual agent. This agent represented the collection of all functional spaces of the designed building. It also included all the restrictions and functional rules of the programme resulting from the competition conditions. This agent was then divided into specific functional clusters and ultimately into individual spaces. Over a hundred separate building spaces were created, each represented by an agent whose main parameters were recalculation of global parameters, such as the total area or the sum of the volumes of all cells, allowed for local changes to these values, while maintaining strict control over the overall vision of the design and its feasibility [T. Jaśkiewicz 2013, p. 207].

The development of the Automotive Complex project in Abu Dhabi (2007) followed the same design methodology as the National Park Centre in Salzburg, using the same virtual design environment, 'protoSWARM'. However, the size of the project was approximately 800 times larger. The functional programme and areas were not precisely defined by the client. Therefore, based on estimated calculations, the global floor area was assigned to the first agent, then divided into individual functions and assigned to subsequent programme agents (from 1,000 to 13,000 m2) in an ordered Fibonacci sequence. Parameters were also

Fig. 36. ONL (Oosterhuis, Lénárd), Automotive City in Abu Dhabi, 2007 – distribution of functional and spatial units in the protoSWARM virtual environment and the physical model; source: T. Jaśkiewicz 2013

area and location. Each agent was locally aware of its properties. It also knew which spaces it should be directly and indirectly connected to and had the ability to ask individual spaces about their parameters. The programmed behaviour in individual cells of the functional programme had simple rules and was adjusted during the design process. Each agent had to avoid collisions with other agents. Connections between cells were created top-down. Spatial attractors such as points and lines were also introduced into the system to repel or attract all or only selected cells to specific areas of the plan. Through these attractors, designers could impose additional, subjectively defined constraints. The assigned: type of function, floor area, number of floors, average floor height, orientations, proportions, as well as additional parameters defining the shape and connections with other elements of the programme. The aim of this project was to achieve high quality and efficiency as well as flexible spatial organisation. The designed building was to be a landmark for aircraft landing at Abu Dhabi International Airport. The organisational principles were once again introduced in the form of 'lines of force', where individual lines were treated as agents in the system. The power lines played more than just an aesthetic role here. They served as guidelines for the form of the building and defined internal roads for cars and pedestrians. They acted as curvilinear repellents for the programme agents. In this way, their orientation was always tangent to the lines of force, which made it possible to combine the functions of the building. Interaction with the agents was most often performed directly by the designers. The resulting arrangement of spatial units was covered with a wrinkled coating reminiscent of the shape of the desert sand [T. Jaśkiewicz 2013, p. 169–170] (Fig. 36).

The above examples show how it is possible to imitate a swarm using simple local actions and interactions between agents in a functional programme. Although the individual rules and behaviours were simple, the system as a whole showed a high degree of adaptability. After changing the parameters, the entire system would reconfigure itself, resulting in an unexpected spatial organisation. However, this organisation would always remain a logical result of predetermined rules and constraints, guaranteeing its proper functioning. If, however, the distribution of the functional programme resulting from its decentralised behaviour were completely emergent and unpredictable, the attractors and relationships between selected cells would be used to restore the system to a stable state, adding deterministic features and imposing constraints so that its operation remains within the selected development scenarios and overall design vision.

An important issue in the design of architectural objects is the distinction and relationship between internal (concave) and external (convex) space. In this respect, data-driven AI and ABM generation techniques can be seen as an innovative and synthetic way of mediating between them.

SI and ABM are based on a potentially unlimited number of movement processes that define the emergence of boundaries between the interior and exterior only during the simulation. Their synthetic nature is based on a basic algorithmic structure that defines neighbourhoods among all kinds of objects. In this case, space as such no longer needs to be organised or constituted by a geometric grid, but generates itself from multiple local interactions of point clouds or particle swarms. Individual units, architectural objects of various sizes, their interiors and exteriors, and the urban fabric of the landscapes in which they are located can be initially modelled on the same algorithmic principle of autonomous neighbourhood interaction according to simple rules. The resulting 'wild' architectures, as Kas Oosterhuis calls them, can be made visible and manipulable with the help of advanced computer graphics (CGI). As a result, AI and ABM generate a range of possible options for future states of buildings, traffic flows or urban spaces that may arise under changing environmental influences. This makes it possible to compare existing configurations with those possible in the future [S. Vehlken 2014, p. 12].

Urban design is moving away from mapping the movement of swarm agents in order to generate an optimal urban plan. The aim is to develop a flexible system that embodies self-organising collective urban intelligence [N. Leach 2009; L.B. Rosenberg 2015, p. 259]. There are many ways to model swarm intelligence within computational techniques. Manuel DeLanda presented an agent-based model of behaviour that can be developed to understand decision-making processes in a real city. These factors should be seen as specific, individual entities rather than abstract entities that embody the collective intelligence of the entire society [M. DeLanda 2011]. Nota bene, DeLanda's research is based on institutional organisations rather than urban planning. However, it envisages the development of a model of intelligent agents capable of making their own decisions and influencing others in their decision-making in order to generate urban fabric.

The term 'swarm urbanism' is often used in design circles. It refers to the 'swarm effect,' in which the urban grid is transformed parametrically using digital tools or Frei Otto's analogue form-finding technique. Such techniques, while producing interesting results, are limited by topological constraints (as in the case of morphodynamic networks) or geometric constraints (as in the case of Frei Otto's form-finding) and cannot produce qualitative shifts in form and space beyond these systems.

A behavioural understanding of urban topology and geometry is an unquestionable advantage of 'swarm urbanism'. This is made possible by an emerging system of collective intelligence, in which individual agents with built-in intelligence respond to each other, producing results with different justifications. Between 2010 and 2015, the Kokkugia team began experimenting with a multi-agent urban fabric design tool.

Kokkugii implemented this technique at the macro level for the Melbourne Docklands urban regeneration project, focusing on extending the Central Business District into the unused harbour area, and expanded it to the micro level at the level of actual building design, as they had previously done for the Taipei Art Centre. In their urban design projects with swarms, Kokkugia's concern is not to simulate actual populations (people or institutions) or their occupations, but to develop processes operating at much higher levels of abstraction, which involve instilling design intentions into a set of autonomous agents capable of self-organising emerging intelligent urban forms. The urban design methodology developed does not aim to find a single optimal solution, but rather a dynamically stable state that results from the instability of the relationships that create it [S. Vehlken 2014].

The application of swarm logic in urban planning allows us to move away from the concept of a master plan in favour of an algorithm as the main tool for urban design. This changes the concept of urban design from a sequential set of decisions (with a reduction in scale to a simultaneous process) in which micro or local decisions interact to create a complex urban system. Instead of designing an urban plan that meets a specific set of criteria, urban imperatives are programmed into a set of agents capable of self-organisation.

Architectural and urban design can benefit from the algorithmic logic of Swarm Intelligence (SI) and Agent-Based Modelling (ABM) and Simulation in the following ways:

- This type of software expands the possibilities for handling and optimising the complex interaction of various input variables for design and construction processes. It integrates the levels of individual particle movements (simulated people, traffic, wind, etc.) both at the mesoscale (individual objects) and at the global level of the urban fabric.
- Teams of agents (if properly tuned) will organise themselves into a series of potentially interesting or desirable forms in recurring cycles of multiple scenarios, thereby changing the understanding of design and implementation processes. From this perspective, architecture will be based primarily on movements. Furthermore, such a generation of forms develops in a way that would not be understandable without media-technological means of agent-based computer simulation.
- SI and ABM introduce a new kind of futurology to architecture. Through computer experiments in ABM software, many different scenarios can be tested and evaluated, offering insight into a variety of desirable futures.
- The design process incorporates both mimetic (zoo-technological) and posthumanistic elements. It combines traditional (human) cultural practices of architectural design with innovative media technologies.
- It is becoming possible to add an increasing number of elements to ABM, which allows for the smooth synthesis of many ideas or the communication of opinions from customers or future users during the ongoing design process.

SI and ABM help to configure environments that are increasingly faced with the task of organising highly advanced and interconnected systems, as well as modelling complex correlations. They can be used wherever there are ‘disturbed conditions’, wherever imprecisely defined problems arise, wherever system parameters are constantly changing, and wherever solution strategies become vaguely complex. Swarm intelligence offers an alternative way of designing ‘intelligent’ systems in which autonomy, emergence and distributed functioning replace control, pre-programming and centralisation [S. Vehlken 2014, p. 12]. Thanks to this access, AI is deeply penetrating various fields of science and culture. AI and ABM are used in economic simulations and models of financial markets, social behaviour, crowd evacuation and panic studies. They have become indispensable in epidemiology, logical system optimisation and transport planning. They are used in telecommunications and network technologies, as well as to improve image and pattern recognition. They are a component of some climate models and multi-rotor systems; they play a role in mathematical optimisation and, above all, in the design of not only architectural objects. [L.B. Rosenberg 2015, p. 612]. SI and ABM are new cultural techniques because they approach complex organisational problems using artificial populations of agents and their behaviour over time. In architecture, this poses new challenges for future generations.

5. MORPHOGENETIC COMPUTER TOOLS THAT HELP IMITATE THE GEOMETRY OF NATURE IN ARCHITECTURAL DESIGN

Generative tools that imitate the geometry of Nature are understood here as a set of techniques for transforming and modifying the initial shape, which lead to obtaining an increasingly complex shape or pattern or its description. These tools were not inspired by observations of Nature, but were developed as a result of exploration in the field of mathematics and geometry.

Geometry, like arithmetic, is one of the oldest sciences. Like other branches of mathematics, geometry has evolved from the study of shapes familiar from everyday life to the study of infinitely dimensional abstract mathematical spaces. The primary goal of both geometry and mathematics is to discover patterns and create them through activities specific to these disciplines. This has led to the development of generative techniques for producing geometric patterns that may or may not imitate or describe patterns and forms found in nature.

We consider techniques that are used in architectural design and serve both to generate form, shape and structure, but also to tessellate geometrically complex surfaces, or are a tool supporting the solving of problems related to optimisation and efficiency. These are primarily techniques such as: Shape Grammars, Voronoi Diagrams, fractal geometries.

5.1. Grammars of Form

Shape grammars originate from analytical geometry, where the use of formal notation made it possible to define all kinds of objects (including non-geometric ones) and their transformations.

Shape Grammars, first described by Georg Stiny and James Gips in 1971, were developed for the purposes of painting and sculpture. [G. Stiny & J. Gips 1972]. It is a set of transformations and modifications made to the initial shape, which lead to the creation of increasingly complex forms (Fig. 37). Transformations can be applied to one selected element or to all elements simultaneously (parallel process) – however, these two techniques can lead to divergent results. Shape grammar is the first generative system aimed at design. Five years later, Stiny's next text, *Two Exercises in Formal Composition*, became the basis for many applications of shape grammar in architecture [G. Stiny 1976].

The concept of introducing grammar into art and design refers to the analogy between 'visual language' and 'natural language'. The role of grammar in natural language is to independently facilitate the communication of meaning by providing structure and organisation. The universal principles of natural language were put forward by Noam Chomsky in 1957. His theory of transformational grammar was based on a system of internalised rules capable of generating an infinite number of grammatical sentences. In linguistics, for generative grammarians, grammar refers to the entire system of structural relations in language, understood as a set of rules for generating sentences. This idea of transformational generative grammar emerged with the use of computers to analyse natural language, focusing on formalistic approaches [M. Özkar & G. Stiny 2009].

Shape grammars are therefore an example of generative systems based on the linguistic model, hence their name. This makes it possible to use rules to determine the process of generating graphically expressed words of a language constructed from an alphabet composed of symbols. Shape grammars allow for the generative imitation of patterns found in Nature's creations.

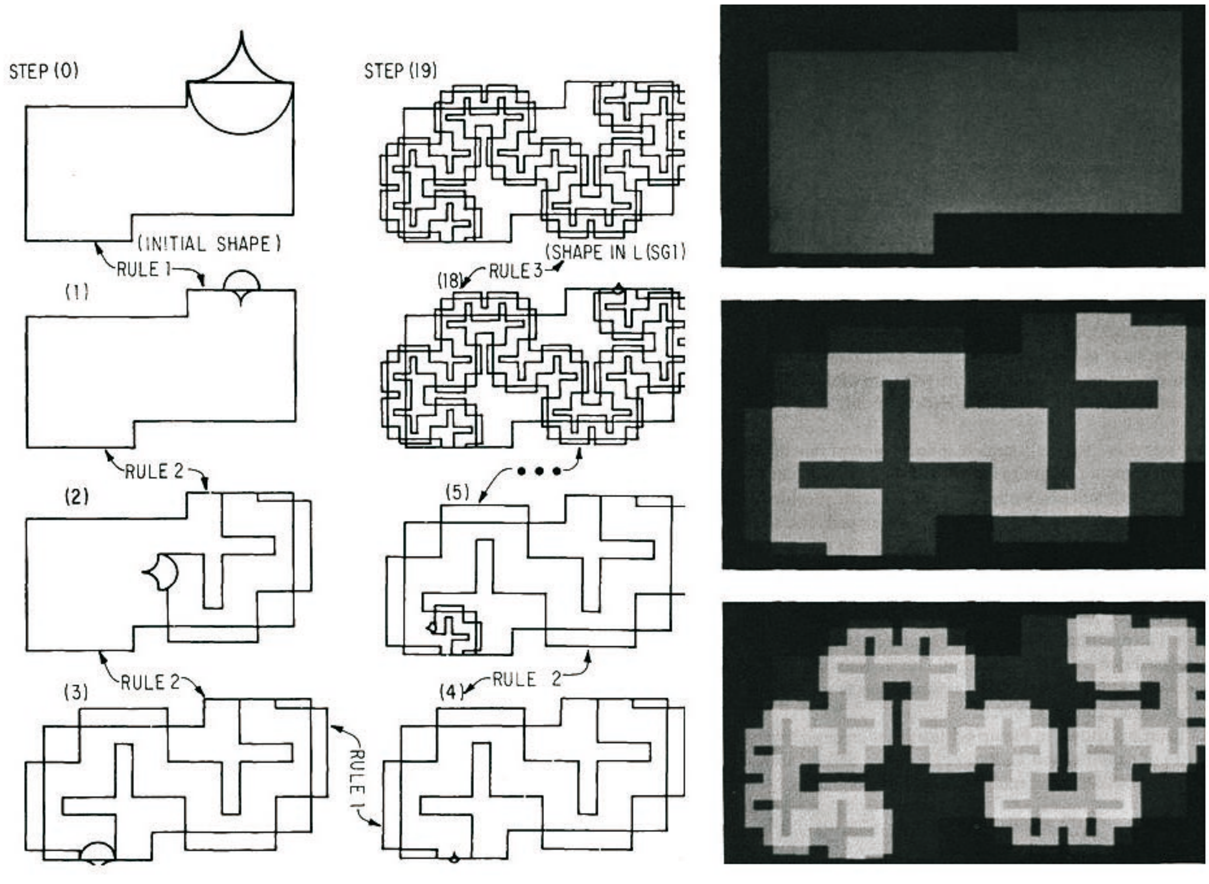


Fig. 37. George Stiny, generation of the shape using the L language (SG 1) and execution of this shape with acrylic paints on canvas in four colors, 1970; source: G. Stiny & J. Gips 1972

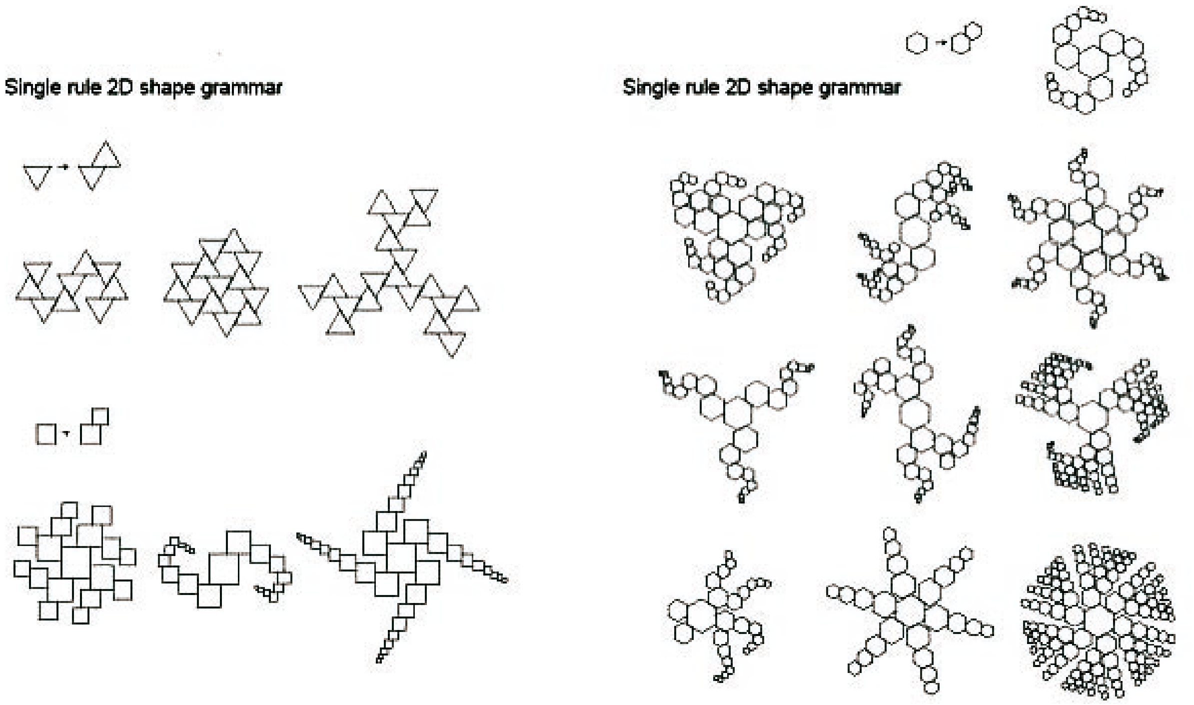


Fig. 38. Shape grammars. 2D representations of systems according to the adopted rules (Basic Grammars II); source: B. Tepavčević & V.Stojaković 2012

Shape grammars are a specific class of system based on artificial intelligence expert rules that generate geometric shapes. A shape grammar consists of shape rules and a generation engine that selects and processes the rules recursively, starting from the initial shape. The rules are used to determine how individual shapes are transformed and to describe the transformation process. Changes can be made to any element of the set that meets the conditions required by a given transformation (e.g. for obvious reasons, a circle cannot have a corner removed in the same way as a square). These rules are based on geometric transformations, i.e. translation, scaling, rotation, and reflection, which allow one shape to be part of another (Fig. 38).

A characteristic feature of Shape Grammar is that a finite set of rules and shapes can generate an infinite number of design solutions. Furthermore, it can serve as an analytical tool for decomposing complex shapes and as a synthesis tool, generating complex forms starting from a simple shape [K. Januszkiewicz 2012b, p. 48].

Architectural interpretation

The application of shape grammar in architectural theory and design has a history spanning over forty years. Shape grammar was adopted in architecture schools long before the development of conventional CAAD (computer-aided architectural design) drawing tools. Despite its popularity in academic circles, shape grammar has not found a place or application in computer-aided architectural design. Nevertheless, this tool has often been proposed as a support technique for architectural designers [A.McKay et al. 2012, pp. 143–144].

Shape grammar was treated as a type of formalism representing visual and even spatial thinking, and the term ‘shape grammar’ was applied to the grammar of visual design. In this sense, shape grammar represents a philosophy of looking at the world, not through learned or imposed decompositions, but through those that have practical significance at a given moment [B.Tepavčević & V. Stojaković 2012, p. 171]. It should be emphasised that the spatial aspect of shape grammar was crucial for its implementation in contemporary architectural theory and design.

Some architects found the analogy between traditional grammar (linguistics) on the one hand and the language of geometric transformation of architectural elements on the other hand appealing. Peter Eisenman was one of the first architects to explore the application of generative grammar in architecture inspired by Chomsky’s theory of linguistics. Generative grammar was the theoretical framework for a series of house designs created in the late 1960s and early 1970s. The grammar of shapes was also used in research into new aesthetics generated by computational algorithms. In 1969, Harvard University opened the first computer lab equipped with IBM hardware, which allowed shapes to be drawn on the monitor screen relative to the x and y axes using a light pen [D. Kurmann 2010, p. 10]. Soon,

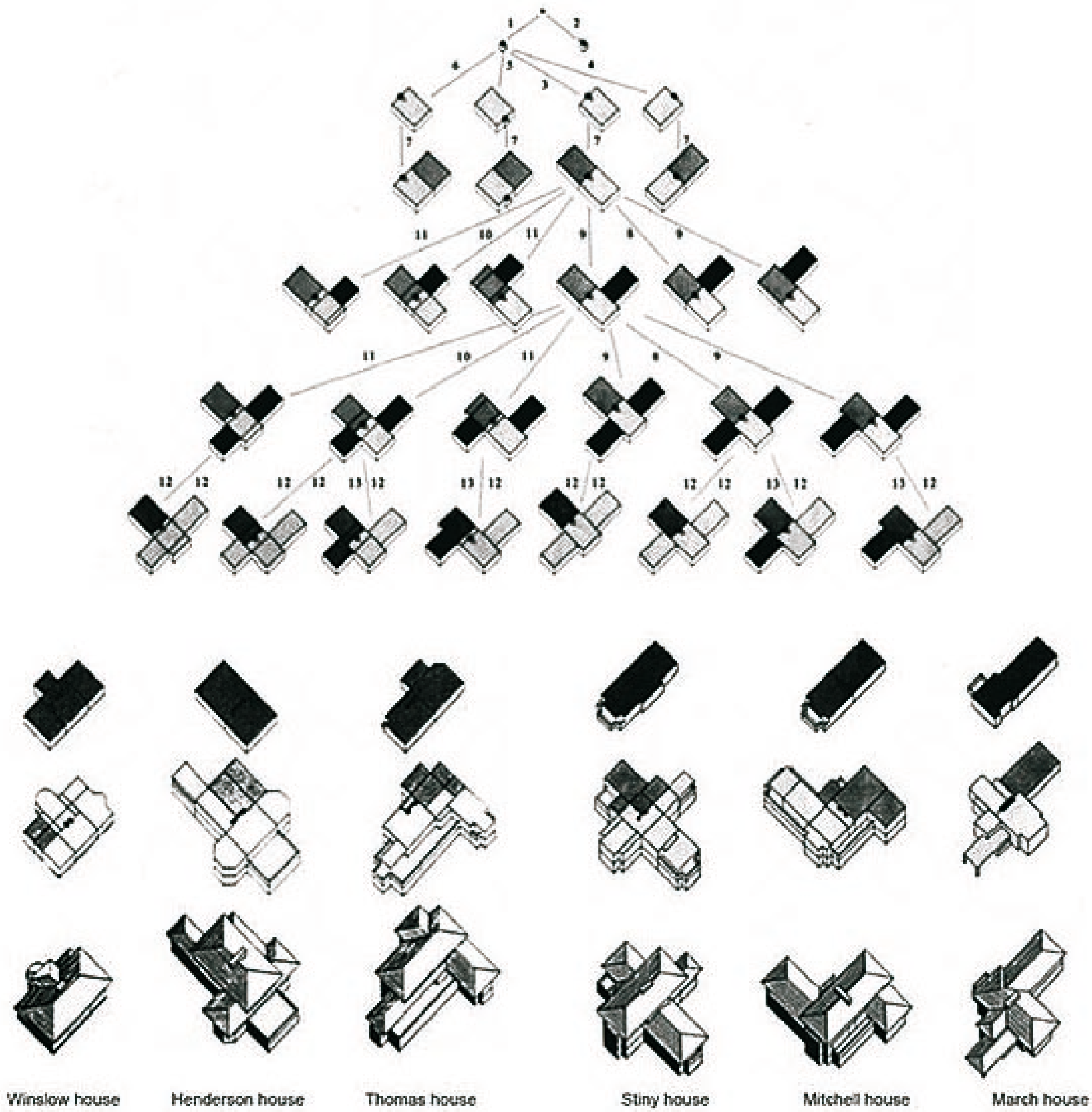


Fig. 39. Frank L. Wright's Grammar of Preii Houses (Koning and Eisenberg), 1981; source: D.R. Shelden 2002

the first version of the Top Down programme based on George Stiny's Shape Grammars was created.

Harvard University was the first in the United States to introduce the requirement to know the TopDown programme in architecture schools. The programme was written at UCLA by Robin Ligget and William Mitchell in 1987–1988 and was aimed at architects. It was inspired, in part, by Shape Grammars and, above all, by the then existing methodology of teaching design through programming. TopDown provided a visual and dynamic way of linking design changes with changes in the dimensions of compositional motifs [B. Tepavčević & V. Stojaković 2012, p. 173] (Fig. 39).

Until the last decade of the 20th century, shape grammars were developed as analytical tools. Thanks to their initial applications, they became an established paradigm in computational design theory.

The first analytical studies using Shape Grammars were conducted by Stiny and published in 1977, and concerned traditional forms in Chinese architecture [G. Stiny 1977]. This research established new analytical standards for examining architectural form. The following year, Stiny and Mitchell published a work entitled The Palladian Grammar, which initiated research into style in architecture using the grammar of form [G. Stiny & W.J. Mitchell 1978]. They proved that by defining rules for shape grammar, Palladio's system of proportions and 'architectural language' could be transformed into a modern 'generative form'.

In historical architecture research, Shape Grammar has proven to be a useful tool not only for modelling buildings with historical and geometric accuracy, but also for building qualitatively correct models that define complex relationships between architectural elements. In this sense, analytical grammar can serve as a platform for studying architectural typology at more complex levels, which cannot be carried out without the appropriate computing resources. With the development of information technology, Grammar of Form is being transformed from an analytical tool into a generative design tool.

In the early 1990s, Shape Grammars were used to teach architectural composition. Architecture students at MIT, Harvard, UCLA, and Yale used Shape Grammars to learn the design language of selected buildings and, based on this, make modifications to generate their own new design concepts. In subsequent years, Shape Grammars have been used for generative design in research projects at MIT. Birgul Colakoglu, Jose Duarte, and Lawrence Sass have made notable contributions to the use of Shape Grammars as a generative tool for designing houses [B. Tepavčević & V. Stojaković 2012, p. 174]. These tools were used, among others, by Alvaro Siza to create his own house models. Based on them, and especially on the models of houses in Vieira Malagueira and Duarte, he developed a successful version of a programme that allows the generation of many unique objects. Sass, on the other hand, introduced an innovative method of using the Shape Grammar procedure to generate house designs that can be made from plywood sheets. The Shape Grammar procedure is used here to divide the initial shape into flat components that are cut using CNC technology. This was an innovative approach to design and construction, combining CAD representation, fabrication and assembly [B. Tepavčević & V. Stojaković 2012, p. 174].

Shape grammar algorithms have been used in several projects by Gehry Partners. These applications aim to rationalise the division of curved surfaces in order to meet specific construction and fabrication requirements [D.R. Shelden 2002, p. 108].

The production strategy based on shape grammar logic was first used by Gehry & Partners in the Experience Music Project (EMP) in Seattle (1997–2000). Shape grammar allows curved free-form surfaces to be divided into flat, single-curvature deformed elements (e.g. sheet metal), whose size is determined based on the Gaussian curvature value [D.R. Shelden 2002, p. 114].

In the EMP project, curved shapes were first decomposed by a surface division algorithm. The basic rule for creating grammar divides a rectangle into four smaller rectangles. Depending on the local curvature of the surface, the algorithm can be repeated recursively in order to obtain smaller, rectangular flat areas of various sizes (Fig. 40). More efficient results are obtained

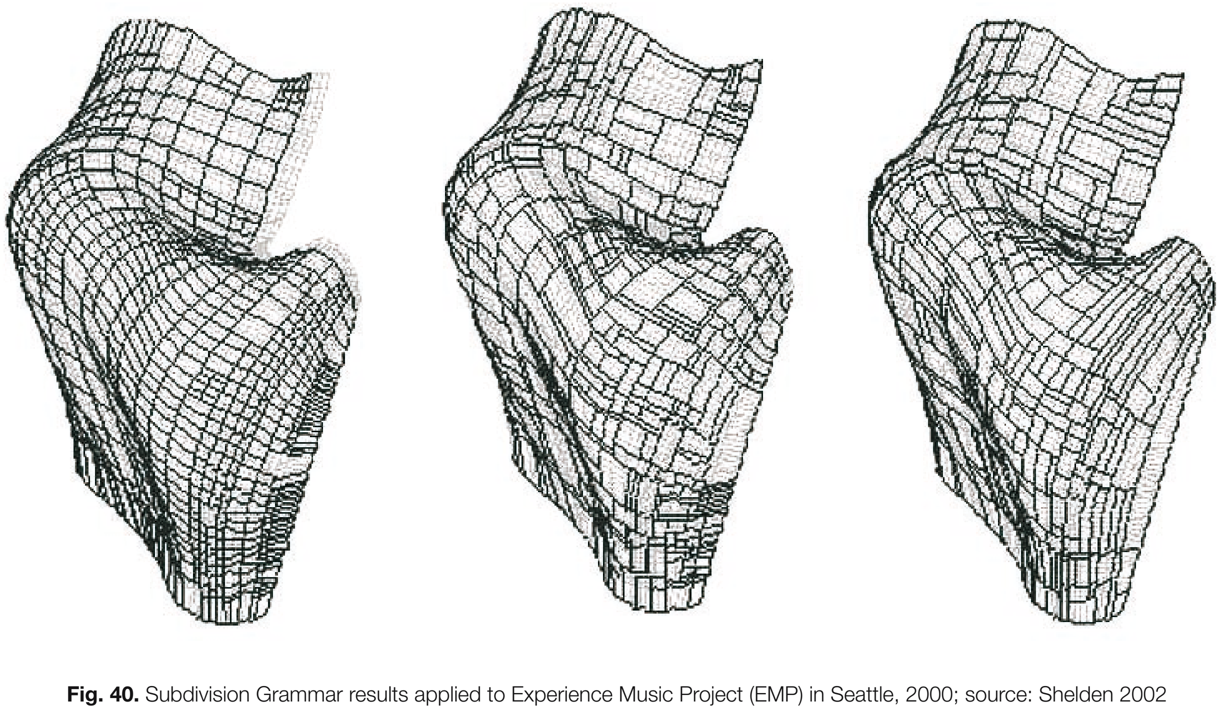


Fig. 40. Subdivision Grammar results applied to Experience Music Project (EMP) in Seattle, 2000; source: Shelden 2002

in subsequent phases of the algorithm as the optimal arrangement of components is sought. This is procedural modelling based on shape grammars.

Incidentally, the first commercial application for procedural 3D modelling based on an innovative grammar language was released in 2008 by Procedural Inc. under the name CityEngine. This application was developed at ETH Zurich by Pascal Müller as part of his doctoral thesis. CityEngine uses procedural modelling, which means that it automatically generates models using a set of rules that iteratively refine the design, creating more and more details. In this context, the design of a building begins with determining the volume of its façade, and the programme iteratively adds details that remain flexible for future changes. Such a building model has a hierarchical structure and contains important semantic information that can be reused when creating new designs [P. Müller et al. 2006]. This application has found wide use in urban architecture design.

The growing interest in 3D shape grammars has led to proposals for representing shape in a less ambiguous way. This has paved the way for many alternative formal grammars, including graph grammars. Graph grammar does not affect shape as such, but acts on graph representations (which often represent geometric shapes). A comprehensive description of how graph grammar works and a number of examples were presented by Amaresh Chakrabarty [A. Chakrabarti et al. 2011]. At the same time, shape grammars were combined with computational tools that imitated morphogenetic processes, which was an exceptionally innovative approach to optimisation in the field of design [G.S. Hornby & J.B. Pollack 2001]. A grammatical evolutionary algorithm was developed, enabling programming based on evolutionary algorithms. This tool was named Grammatical Evolution [M. O'Neill et al. 2009]. Until now, evolutionary algorithms had not been combined with grammatical representation (or design language), which filled an important gap. Evolutionary algorithms (EAs) have been successfully used to solve various design problems, and it has been shown that evolutionary techniques can be adapted (by changing the scale) to the complexity required for typical geometric patterns found in building designs (e.g., floor plan shape). Typically, direct coding, either through parameterisation of the search space or through component-based solution representation, was used to solve the problem. It was argued that a generative coding scheme, coding that specifies how the phenotype is constructed, can achieve greater scalability due to its self-similar and hierarchical structure. Furthermore, by reusing parts of the genotype to create the phenotype, generative coding is a more compact solution coding [M. O'Neill et al. 2009, p. 1037].

Currently, there are many versions of Shape Grammars available, as they allow human knowledge to be encoded naturally in a generative process. They prove invaluable not only in the design of architectural objects, but also in product design. For example, the Integrated Design Innovation Group at Carnegie Mellon University uses Shape Grammars to study the essence of product design, ranging from Harley Davidson motorcycles to passenger cars. Some versions of Shape Grammars are used today: generating systems such as cellular automaton rules for creating 2D shapes, context rules for creating 2D tiles, graph coding for animated 3D creatures, and cellular coding for artificial neural networks.

5.2. Voronoi diagrams

Voronoi diagrams, like Shape Grammars, were developed as a result of exploration in the fields of mathematics and geometry. Voronoi diagrams enable the decomposition of space and its division into regions. It is a type of tessellation that includes a class of patterns called Dirichlet tessellations. It is a special type of decomposition of metric space defined by distances to a specific set of discrete objects in space, e.g. by a discrete set of points. Although these diagrams had already been considered by René Descartes, they are named after the Russian mathematician Georgy Fedorovich Voronoi, who in 1907 defined and studied the *n*-dimensional cases of this tessellation.

Voronoi diagrams are created from a set of points distributed over a specific surface (2D diagrams) (Fig. 3.41). It involves dividing the studied area into parts (cells) in such a way that each point located inside a given cell of the diagram is closer to the node located in that cell than to any other node in the network [K.E. Brassel & D. Reif 1979]. Each point divides this area into *n* areas so that every point in any area is closer to a specific point from the set of *n* points than to the remaining *n* –1 points. All Voronoi areas are convex polygons. Every point in any area is generative. Delaunay triangulation is a double graph for the Voronoi diagram (Fig. 41), which means that the space (Rn + 1) into convex polygons (simplexes) such that two T simplexes have a common face or have no common parts (each bounded set Rn + 1 has a common part with only a finite number of T simplexes), e.g. the interior of a sphere inscribed in any T simplex does not contain any vertices of T simplex.

The idea of dividing space, similar to a Voronoi diagram, first appeared in the 17th century with René Descartes' (1596–1650) concept explaining the distribution of matter in the universe. He described this concept in the third volume of 'Principia Philoso-

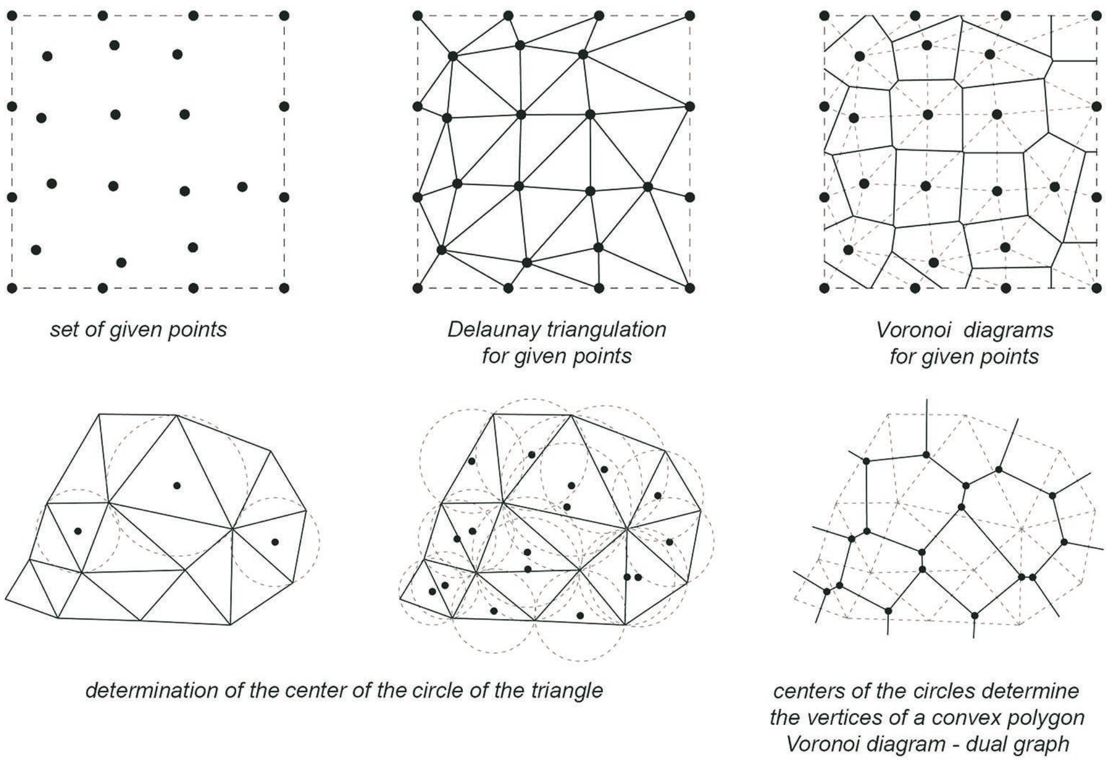


Fig. 41. Delaunay Triangulation and Voronoi Diagram, a) Creation of Voronoi Diagram on divisions of triangulated Delaunay mesh, b) Delaunay Triangulation and Voronoi Diagram for a given group of points; source: Rokicki & Gawell 2016

phie' published in 1644[3]. Johann Friedrich Carl Gauss (1777–1855) was also interested in the division of space and surface from a mathematical point of view. In 1801, Gauss published his work 'Disquisitiones aritmeticae' on number theory, in which he presented theories of quadratic forms and proved the law of reciprocity of quadratic residues using diagrams of the division of *n*-areas [K.E. Brassel & D. Reif 1979, p. 290]. Formally, however, this diagram was not introduced into literature until more than two centuries after Descartes. This occurred when Peter G. L. Dirichlet (1805–1850) published his book Theory of Numbers (1850) in 1838, and Gorgy Fiodorowicz Voronoi (1868–1908) presented his tessellation methods. Hence, Dirichlet tessellation and Voronoi diagrams.

Voronoi tessellation describes the self-organisation system of visible biological structures, including divisions on dragonfly wings, turtle shells, honeycombs and sea urchin skeletons (Fig. 42). The principles and methods of discretisation and self-organisation of living organisms inspire the search for optimal solutions for artificial structures.

Voronoi diagrams are used in almost all fields of science and engineering. They can be used to describe more than just biological structures. In aviation, they are used to identify the nearest airport in the event of detours. In mining, they can help estimate overall mineral resources based on exploratory drilling. In epidemiology, they can help identify the source of an infection. Voronoi diagrams are also widely used in computer graphics, geophysics, anthropology and urban planning. Voronoi diagrams are used to generate both two- and three-dimensional elements, as in the case of Benjamin Aranda and Chris Lasch in their project *Grotto* (2006).

Voronoi tessellations are considered one of the most fundamental data structures in computational geometry. They are used in modelling natural phenomena, to study their mathematical properties, especially geometric, combinatorial and stochastic ones, and their computational representation. These tessellations also offer various methods for grouping multidimensional data [F.Aurenhammer 1991, p. 347].

[3] see: R. DesCartes, *Principia philosophie,* Elzevir, Amsterdam 1644; also R. Descartes (translated by Izydora Dąmbska), *Principles of Philosophy*, Wydawnictwo Naukowe PWN, Warszawa 1960.

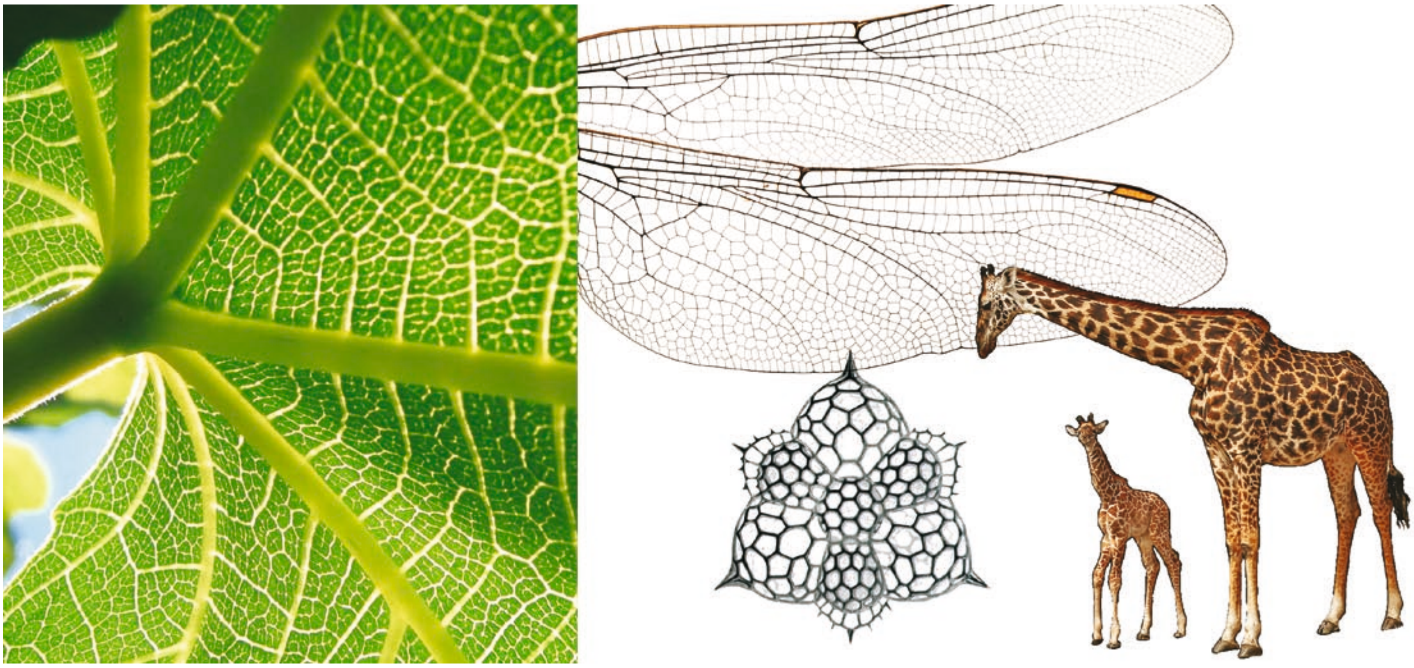


Fig. 42. Voronoi diagrams in the formations of Nature; source: N. Paszkowska-Kaczmarek 2022

Architectural interpretation

The spatial and biological aspects of Voronoi diagrams make these tessellations an attractive tool for architectural and urban design. Models of biological self-organisation found in nature serve here as patterns for shaping the built environment. The Voronoi topological structure, which fills space, simulates the natural exchange of information about objects in the architectural environment and divides a given space into a set of subregions according to the data of these objects.

Voronoi diagrams initiate the process of generating and evolving spatial forms that interact with all entities in the system – self-organisation. This leads to the evolution of the entire system, which constantly multiplies according to a spatial pattern defined on the basis of self-organisation. In the context of architecture, this would refer to co-adaptation, which finds space among the complex relationships and needs that exist in the disorganised subsystems of the environment [P. Coates & Ch.Derix 2005]. Furthermore, Voronoi diagrams are also used to analyse space occupancy. In this case, the centres of Voronoi cells become search agents which, during the search process, receive information about the surrounding cells. When the information from the surrounding cells matches their preferences, the agent occupies the space.

Zaha Hadid Architects used Voronoi diagrams to analyse data from the environment of the National Kaohsiung Performance Arts Centre site. The relationships between spatial elements such as existing buildings, plot boundaries, trees, monuments and access roads were analysed. These analyses made it possible to identify the main pedestrian flows and controlled access points, while clouds of active points determined the directions for the main visitor routes. In other words, Voronoi diagrams were also used here to control movement in space. The shapes of the cells in the resulting diagrams became a means of artistic expression, both for the surroundings and for the building itself [J. Park et al. 2008, p. 527] (Fig. 43).

Voronoi diagrams can also be helpful in explaining intangible social phenomena, and due to their geometric structure, they can actively participate in shaping space. In this case, the purpose of the diagrams is to establish a communication system and actively respond to changes and characteristics of the environment to which a given space belongs. The task of architecture is to create space for people and nature by gathering information about the environment surrounding that space in order to formulate requirements for architectural solutions.

Based on Voronoi diagrams, the Korean team Net. lab worked for sixteen months on a research project carried out by GNome. The research focused on how algorithms can be applied parametrically in the creation of cellular spaces in relation to specific variable-conditioned criteria expressing diverse social systems, scales and user needs (Fig. 44). A series of computational procedures based on the Voronoi algorithm were developed and incorporated into an application

Fig. 43. Zaha Hadid, Patrik Schumacher, National Kaohsiung Performance Arts Center, Kaohsiung, Taiwan 2007; source: Zaha Hadid Architects, Hadid 2007

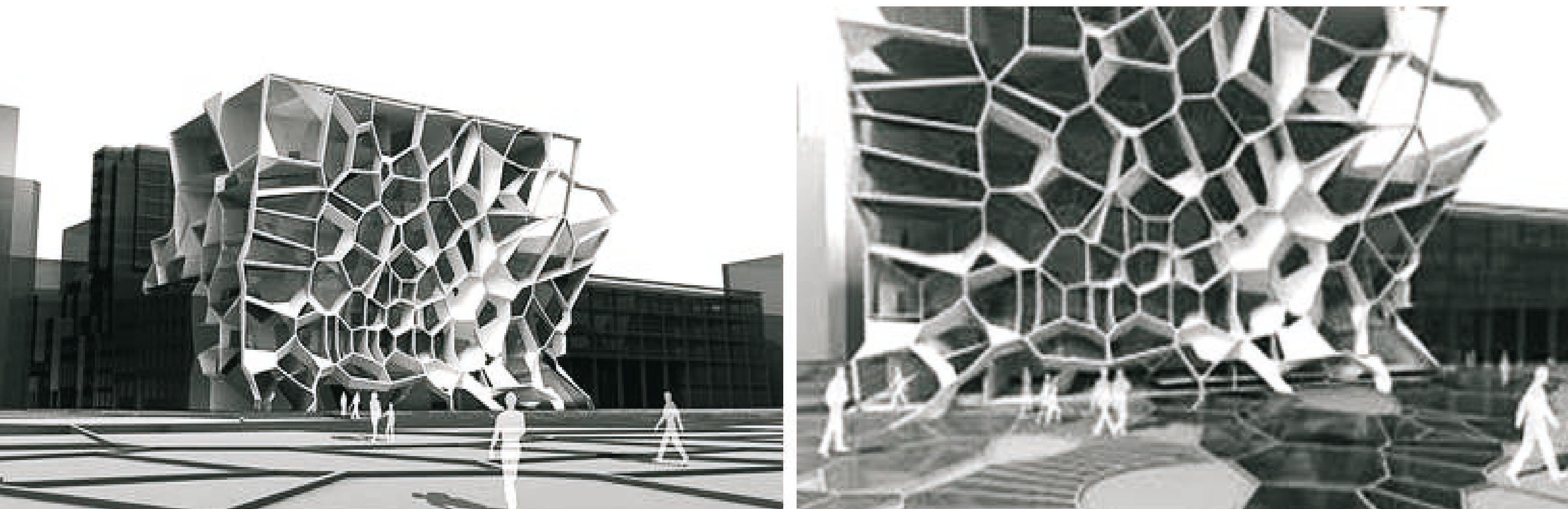
generated during the research. This application enables an iterative process of feedback, adjustment and optimisation of the design to given variable conditions. In addition, emphasis was placed on the integration of design, analysis and production processes through the application of Voronoi algorithms in real-world contexts [J. Park et al. 2008, p. 527].

Architects also use Voronoi diagrams not only to study spatial situations. The ease of generating geometric patterns based on these diagrams makes them an attractive aesthetic solution when it comes to changing, adapting and adjusting a given space [K.A. Kang & J.E. Yoon 2008, p. 156].

An example of such an application is the redevelopment project for Glorieta Juan Carlos I square in Spain, where ESC Design Studio designed an immersive environment using photovoltaic panels and artificial fog generation systems. The pattern on the square was generated using custom software that implements the Voronoi algorithm to divide the surface into pedestrian routes and allocate areas for other activities (Fig. 45).

The architectural interpretation of Voronoi diagrams is also visible in the Aldgate Aerial Park project in London. The Matsys' idea was to create additional public space above street level during the 2012 Sum-

Fig. 44. Net.Lab (South Korea), Interpretations of Voronoi Diagrams in Urban Public Spaces; source: Park et al. 2008



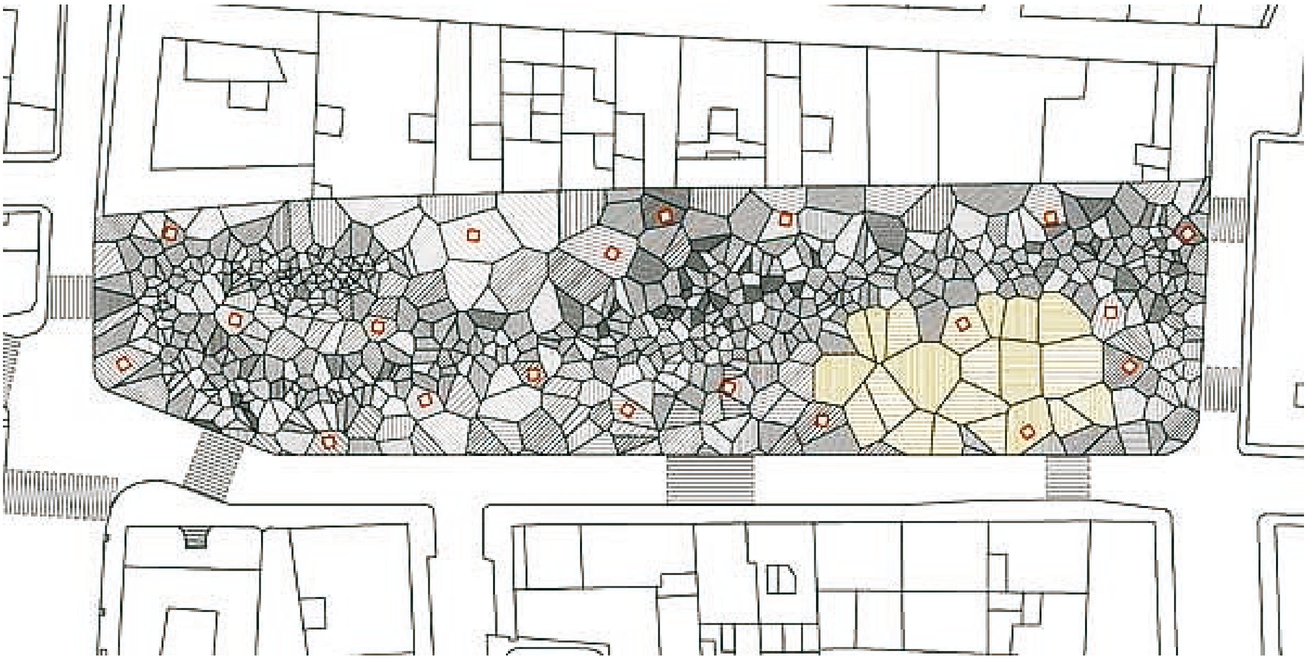
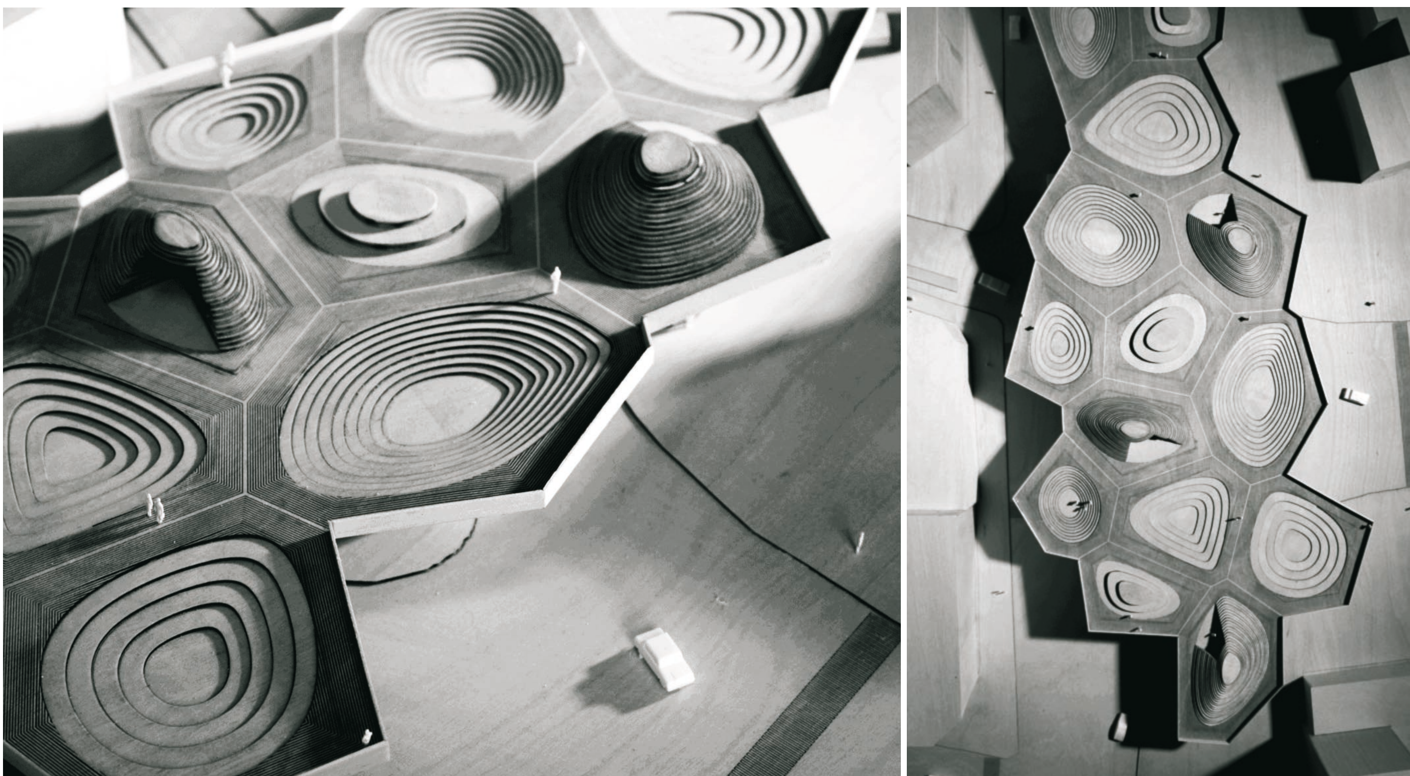


Fig. 45. ESC Design Studio, Glorieta Juan Carlos I, Mula, Spain 2009, Interpretations of Voronoi Diagrams in Urban Public Spaces; source: Nowak 2015

mer Olympics, where there would be separate gardens and places to rest. These were placed in individual cells of Voronoi diagrams (Fig. 46). In both cases, the interpretation of Voronoi diagrams brought the desired benefits, not only functional but also aesthetic [A. Nowak 2015, p. 31].

Voronoi diagrams, just like Delaunay triangulations, allow for the interpretation of a two-curvature free surface, i.e. its appropriate geometric approximation. Thanks to these tessellations, the essential features of the three-dimensional form contained in the digital model are preserved. However, how this surface will be

Fig. 46. The Matsys, Aldgate Aerial Park, London, 2012, Interpretations of Voronoi Diagrams in Urban Public Spaces; source: Nowak 2015



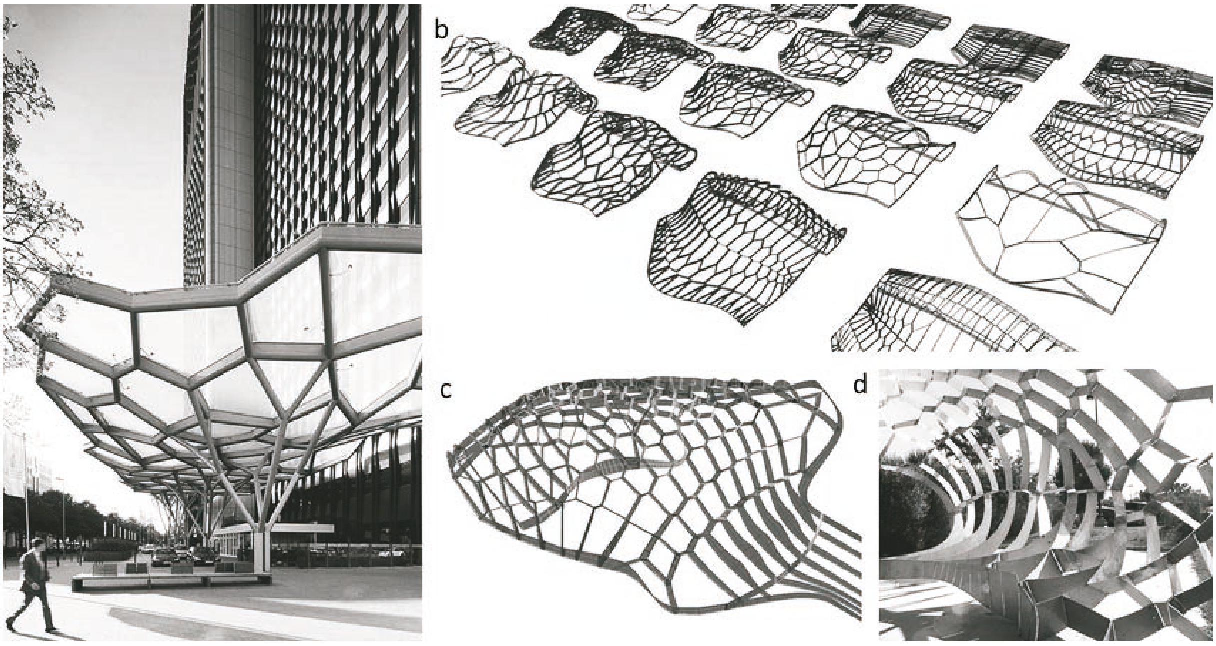


Fig. 47. Application of Voronoi diagram in discretization of structural surfaces; a) covering of WestendGate in Frankfurt, (project by Just Burgeff Architekten and a3lab, 2010); b) generation of Voronoi divisions – searching for rational solutions for the meditation pavilion in Houston, (project by Metalab Architecture & Fabrication with scientists from the College of Architecture in Houston); c) final digital model; d) finished view of the pavilion; source: Rokicki & Gawell 2016

produced depends on what will be defined as its supporting structure and what as its filling, how the tectonic elements constituting the physical entity of this surface will be distributed [K. Januszkiewicz 2010, p. 73].

The above examples of completed structures of various sizes and functions confirm the ability of Voronoi diagrams to geometrise and structure curved surfaces of free forms (Fig. 47). In digital design practice, various strategies are adopted for geometrising surfaces for manufacturing. Two-dimensional fabrication often involves contouring, triangulation or polygonal tessellation. Parallel, developable or unfoldable surfaces are also used. The aim is to derive two-dimensional flat components that are easy to manufacture using CNC from a geometrically complex surface. The challenge is therefore to select an appropriate geometric approximation that preserves the essential qualities of the three-dimensional form represented by the digital model [K. Januszkiewicz 2010, p. 73].

The shaping of spatial forms and the discretisation of the supporting structure of free surfaces based on Voronoi cells led to the development of three-dimensional diagrams.

3D Voronoi diagrams inspire the design of complex, multi-cell architectural objects (Fig. 48). Furthermore, they are an effective method of describing the distribution of cubic elements (cells) in space and the relationships between neighbouring points that define the enclosed space (Concave) [F. Aurenhammer 1991, p. 348]. Cells that share facets are recorded in the process of creating Voronoi diagrams. Thanks to this topological representation, it is possible to conveniently extract neighbouring spatial units.

Voronoi diagrams are now a widely available parametric tool for designing two- and three-dimensional objects. The wide availability of this tool encourages experimentation in the search for new structural and spatial solutions. The shaping of architectural and structural forms by Voronoi diagrams is one of the most important means not only of artistic expression of architectural forms, but also a synonym for their synergistic relationship with the works of Nature.

5.3. Fractals

Fractals, like Shape Grammars and Voronoi Diagrams, are the result of scientific experiments in the field of geometry and mathematics. The mathematical basis of fractals developed with the advent of high-performance computers. Thanks to this, it is now possible to create various types of graphical representations of fractal objects using generative tools (e.g. UltraFractal) that ensure recursion and iterative formation of geometric patterns.

A fractal (Latin: *fractus* – broken, fragmented) is a geometric object with self-similarity, whose dimension is not an integer and is difficult to describe in Euclidean geometry. Fractal curves consist of infinite elements that are infinitely small and therefore intangible (Fig. 49).

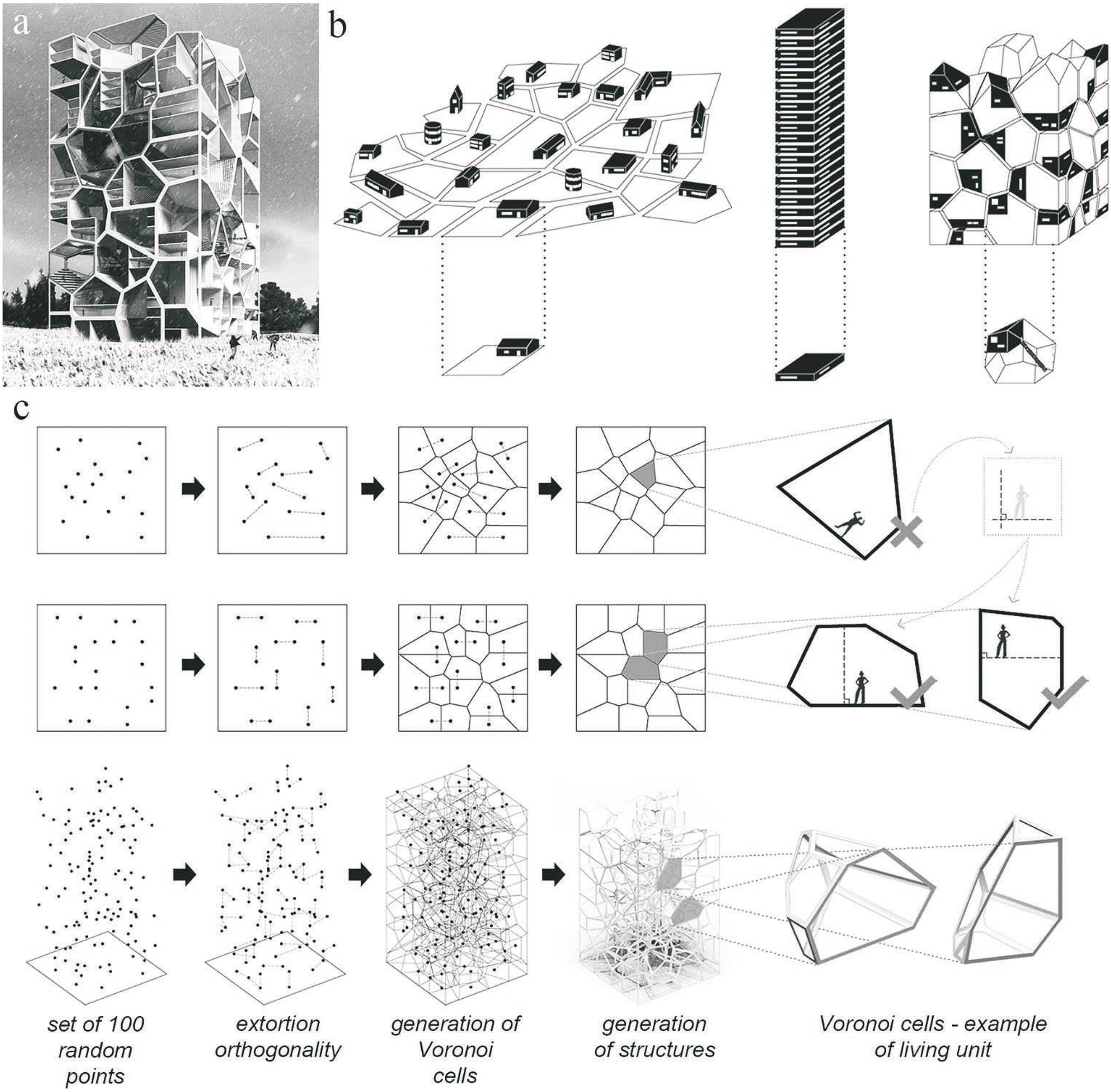


Fig. 48. Voronoi Diagram application in three-dimensional structural surface discretization; Vertical Village, Yushang Zhang, Rajiv Sewtahal, Riemer Postma and Qianqian Cai project, 2011; a) object visualization; b) block diagram; c) rules for generating Voronoi divisions into residential units; source: S. Roudavski 2009

These infinite elements are the reason why the length increases to infinity on an infinitely small scale, making it impossible to define a point on a fractal curve using coordinates or to accurately describe its position on the curve [J. Kudrewicz 2007, p. 20]. In mathematics, a fractal is a set whose fractal dimension (Hausdorff-Besicovitch) exceeds its topological dimension. The dimension referred to in the definition of a fractal is used to describe the degree of 'roughness' of a geometric object [J. Kudrewicz 2007, p. 53].

Fractals as mathematical objects entered the realm of science at the turn of the 19th and 20th centuries, although they were not called fractals at the time. They were treated as geometric curiosities, mathema-

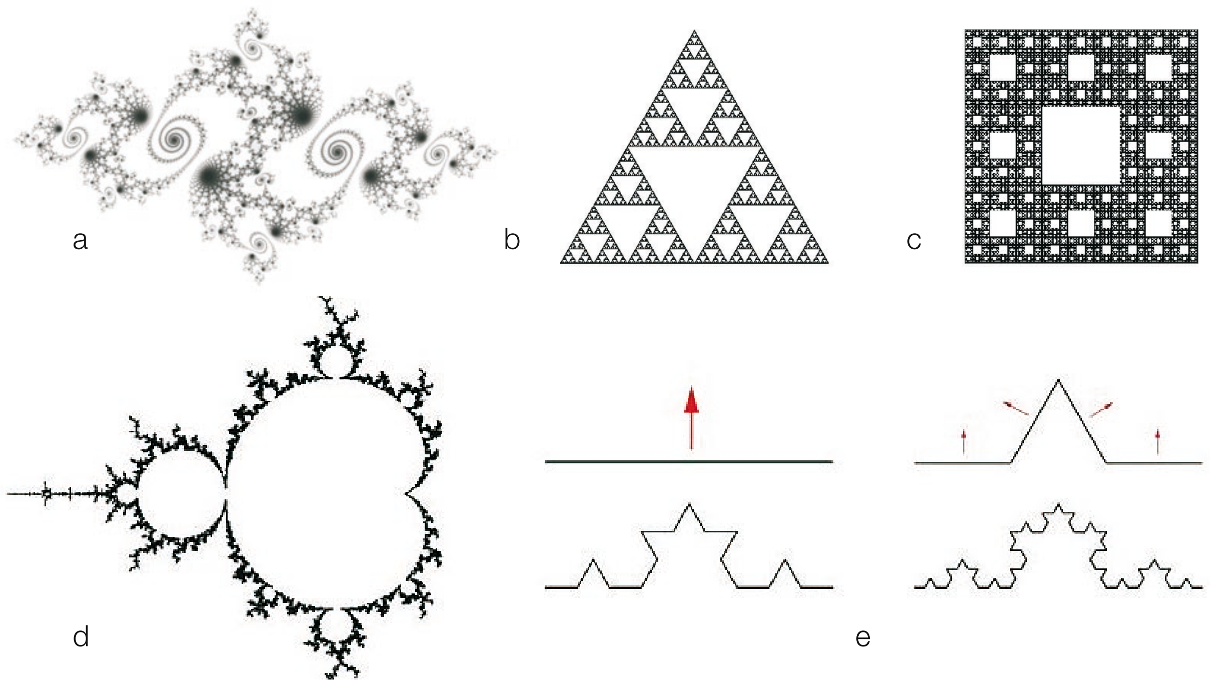


Fig. 49. Fractals on the plane, a) Julia set, b–c) Sierpiński triangle and carpet, d) boundary of the Mandelbrot set, e) formation of the Koch snowflake; source: own elaboration; R.J. Krawczyk 2002

tical oddities that contradicted Euclid's order. Fractal geometry was formulated and formalised in the late 1970s by Benoit Mandelbrot (born 1924), a mathematician of Polish origin. However, earlier Georg Cantor (1845–1918), Guiseppe Peano (1858–1932), David Hilbert (1870–1924), Helge van Koch (1862–1943), Wacław Sierpiński (1882–1969), Gaston Julia (1893–1978) and Felix Hausdorff (1868–1942) studied various mathematical aberrations, which today are considered precursors of fractal geometry.

Euclidean geometry does not describe real-world objects such as trees, clouds, mountains, etc. Ideal shapes such as lines, circles, squares, cubes, etc. are human-invented simplifications of Nature. Fractal curves, on the other hand, consist of infinite elements that are infinitely small and self-similar, and therefore can describe what Nature builds.

Fractal geometry is currently a rapidly developing field of knowledge. It is studied by specialists in various sciences: mathematicians, physicists, mechanics, as well as architects and artists. Fractals have also found application in communication, data processing and information storage. Many researchers claim that this is the geometry that Nature uses to build its creations. Fractal shapes can be found in clouds, coastlines, mountain ranges, snowflakes, trees and soap suds. Today, this geometry gives architects the opportunity to capture connections with nature and the cosmos, as well as to manifest a departure from the concepts of Newton and Laplace.

Architectural interpretation

Architects perceive fractal geometry as an integral part or marker of chaos theory and complexity science [M.J. Ostwald 2001; M.J. Ostwald 1998]. However, there are not many similarities between the definitions of architects and mathematicians regarding 'fractal architecture'. Architects have ignored mathematicians' views on the built environment, and mathematicians have ceased to analyse the work of architects who have appropriated and use fractal geometry. They are mainly concerned with formal effects in the search for new aesthetic qualities. Nevertheless, fractal geometry allows for the mathematical study of forms that represent an endless sequence of self-similar shapes, details meandering from large to small scales. It has descriptive power, allowing us to capture, explain and understand the complex diversity of Nature's creations.

Fractal geometry is a unique design tool that allows one to reach the core of architectural composition, enabling the expression of a complex understanding of Nature. [C. Bovill 1996, pp. 103–113]. Especially when it comes to the use of rhythm, fractal concepts can be applied, e.g. a relatively simple formal technique such as the progression of forms (from larger to smaller). Scaling is a fundamental property of all fractals. Hierarchy and size distribution apply to all complex systems. Both those we can see and those invisible to the human eye contain many small components, only a few medium-sized ones and not many large ones. This distribution results from the universal power law, according to which the number of parts in a system is inversely proportional to their size. This universal law is observed by most natural systems (e.g. DNA, lungs, blood vessels, nerves, etc.), as well as by complex artificial systems (e.g. the global power grid, etc.) [N.A. Salingaros 1999, p. 83].

Fractal geometry shows that different scales with complex structures, as seen under high magnification, combine into a single complex system. This is not only a visual geometric measure, but also affects the stability of the complex system. It has been observed that stable systems follow a universal distribution and size distribution. However, there are many other criteria that a structure must meet to guarantee its stability. The correct distribution of components is a necessary but not sufficient condition for the stability of the system. When some weight is missing, the system is less structurally stable. Stability is then threatened by a violation of the hierarchy and universal distribution of sizes. The same happens in situations where too many scales are randomly distributed. The sequence of scales must comply with the universal distribution: larger scales contain fewer units of that size, while smaller scales contain more and more smaller units. These general properties of complex geometric systems refute the stylistic assumptions of the modern movement, which after the Second World War led to the design and construction of large-scale urban and architectural forms. No modifications will make such forms suitable for human biology [N.A. Salingaros 2008, pp. 10–11].

An interpretation of the principles of fractal geometry and the universal principle of hierarchy and distribution can be seen in the design of the Biocentre at Goethe University in Frankfurt (Fig. 50). Peter Eisenman skilfully combined fractal geometry with Euclidean geometry, obtaining a wide spectrum of components of various scales and sizes.

In 1990, Aaron Betsky described Eisenman's Biocentre as a conventional geometric system corrupted by fractal geometry, which he called a virus and a parasite [A. Betsky 1990, p. 184]. Under pressure from criticism, there is a retreat from attempts to in-

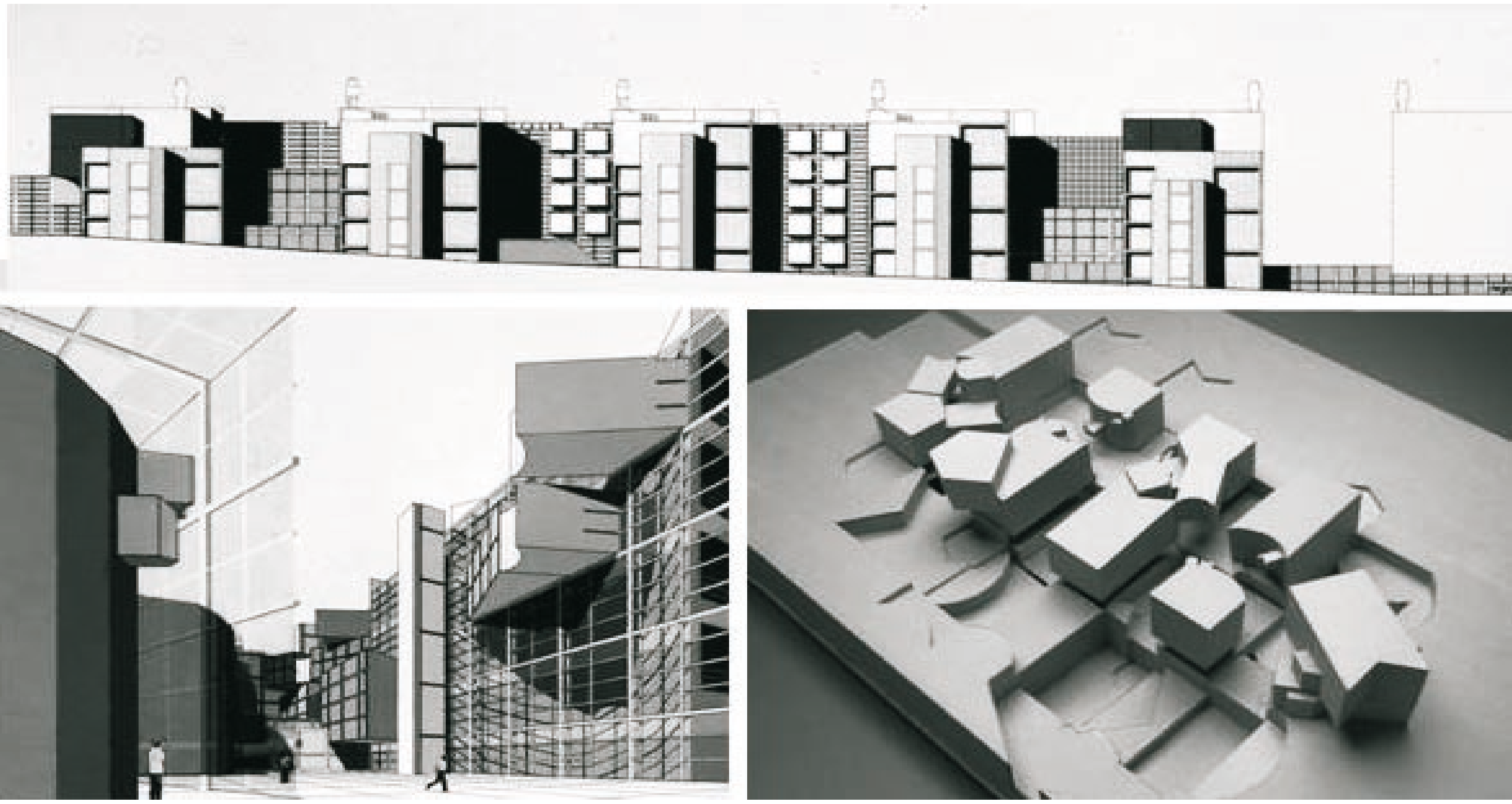


Fig. 50. Peter Eisenman, Biocentrum, J.W. Goethe University, Frankfurt, 1987; source: Betsky 1990

terpret chaos theory and fractal geometry and from the search for a connection between architecture and complexity theory. However, the early 1990s saw rapid development in 3D design tools and human-machine interaction. It is now possible to transform solids and simulate movement. New tools are taking design into the third dimension, changing creative attitudes towards the understanding of complexity, continuity and linearity in architecture.

New IT tools are taking design into the third dimension, changing creative attitudes towards the understanding of complexity, continuity and linearity in architecture. Using these tools, Greg Lynn formulated a concept for the Cardiff Bay Opera House based on a fractal he generated himself (Fig. 51). This fractal served as the basis for further interpretation in configuring the functional programme and spatial requirements of the building. The skilful contouring of curvilinear forms allowed the architectural surfaces for individual components of the form to be stretched. By exposing the solids in different scales and sizes, Lynn brought out their hierarchy of importance and balanced distribution. This was a completely new approach to architectural design, which went beyond the ideological and formal framework based on Euclidean geometry.

Similarly, the architectural firm Ushida Findlay, using new design tools, developed a series of innovative designs in the 1990s using the golden ratio and fractal geometry (often in combination). Their S-Project presents fractal geometry in a particularly attractive way (Fig. 52). It concerns the redevelopment of a Tokyo neighbourhood at the intersection of multiple transport arteries and a railway line. The S-Project examines the concept of “the city as home”, an idea that has gained prominence with the realisation that natural systems have similar patterns on many scales.

Charles Jencks’ controversial book, published in 1995, reopened the debate on the influence of com-

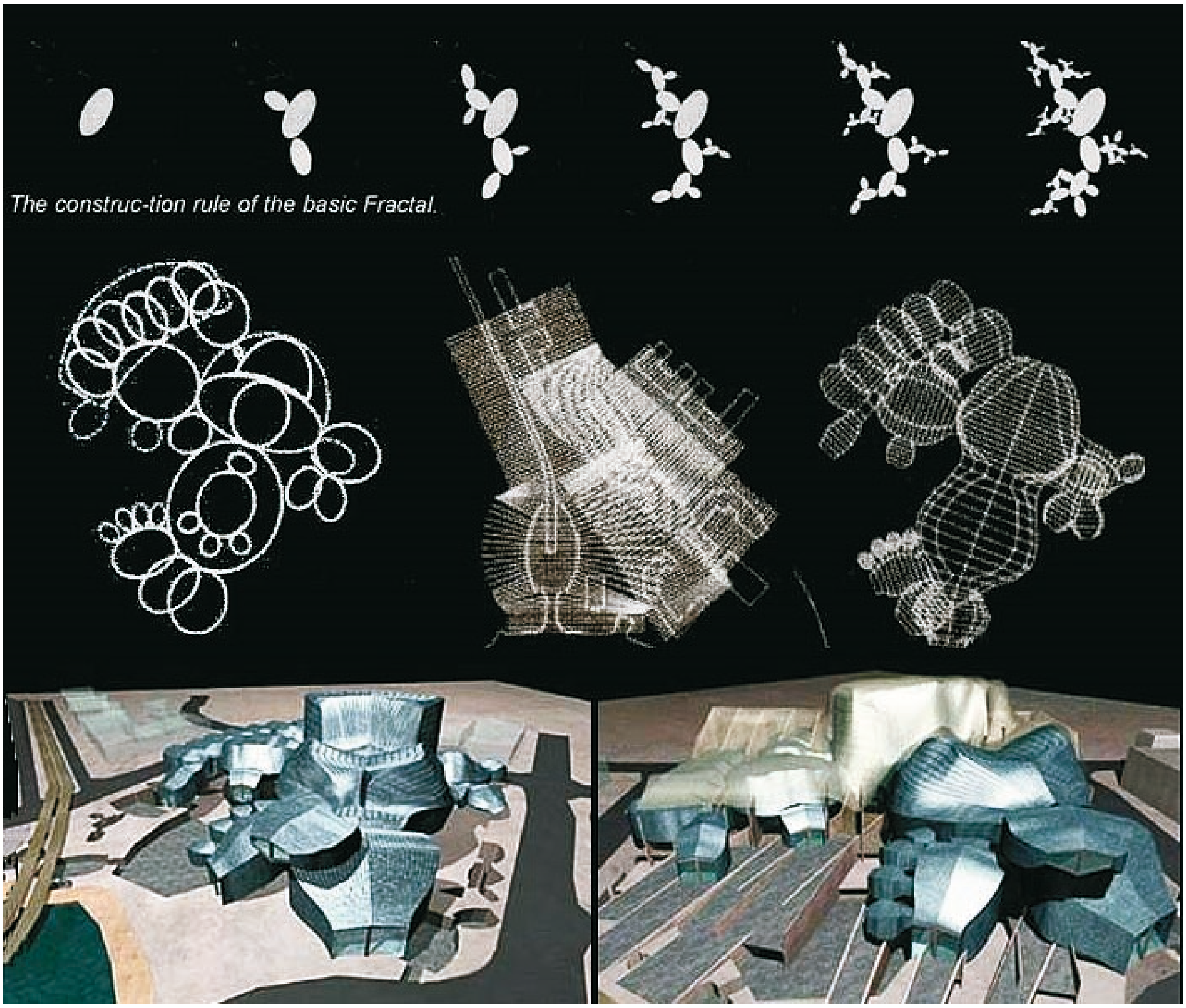


Fig. 51. Greg Lynn, Cardiff Bay Opera House, 1994 – competition entry; source: Jencks 1995

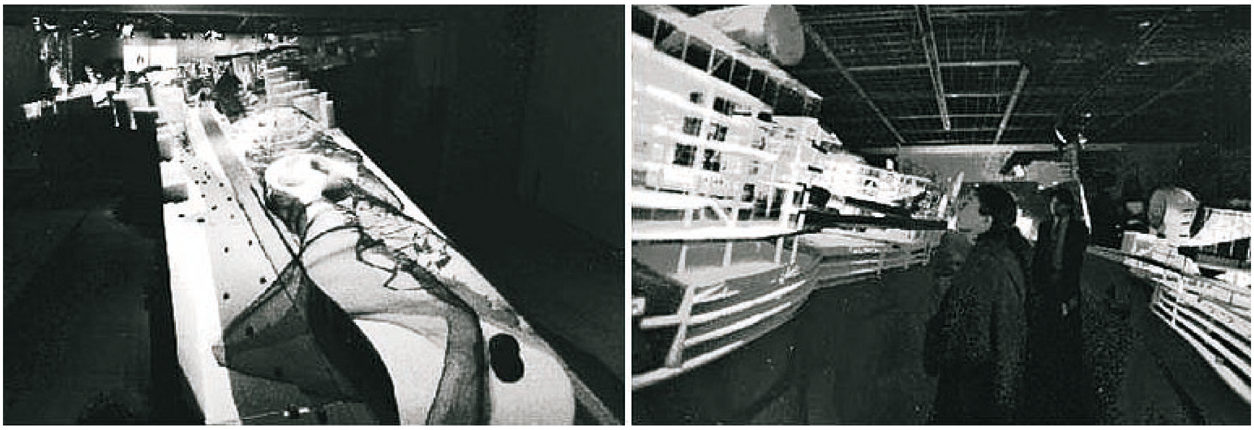


Fig. 52. Ushida + Findlay, S-Project "Parallel Landscapes", Tokyo 1996 – concept design (model); source: Ullrich 1973

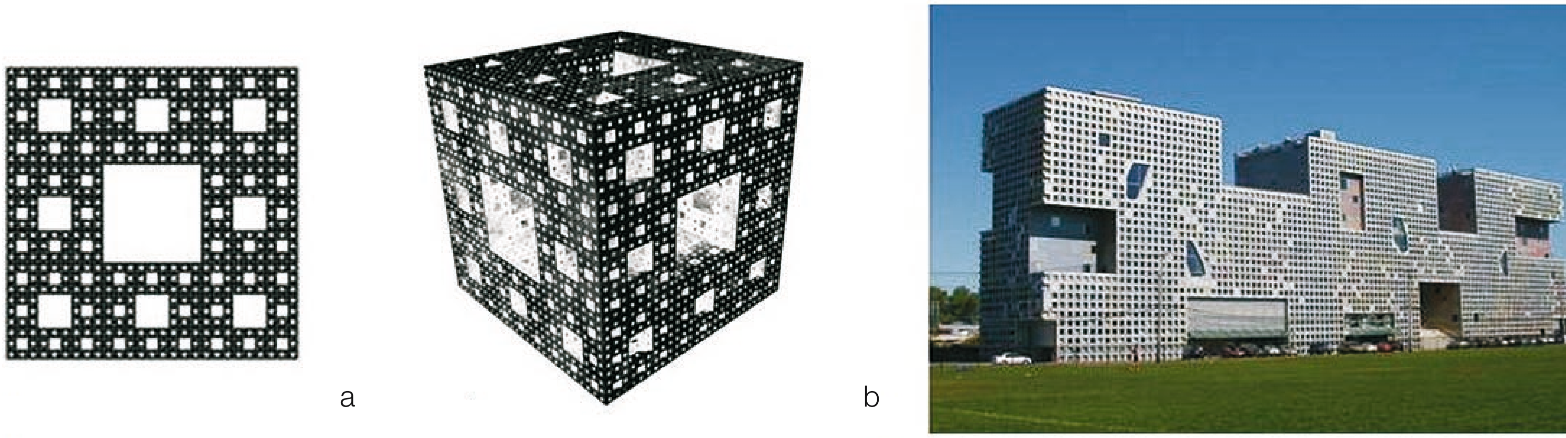


Fig. 53. Steven Holl, Simmons Hall, MIT, Cambridge, USA, 1999–2002; fractal models: a) Sierpinski carpet b) Menger sponge are an inspiration for the form; source: J.R. Koza 1992; Wikipedia Commons

plexity science on architecture and culture [Ch. Jencks 1995], and it was supplemented by Carl Bovill's research findings on fractal geometry in architecture, published in 1996, based on the mathematics of complexity theory [C. Bovill 1996]. Bovill argued that humans have an innate need to analyse the structures of natural creations and then look for similar features in man-made artefacts, which he supports with analyses of building facades from different historical periods.

Since Mandelbrot defined the concept of a fractal, fractal geometry and fractal dimension have come to be seen as indispensable tools, not only for imitating Nature's creations, but also as instruments for the analytical study of designed and existing objects. These are tools that can be used to give humanistic characteristics to the built environment and restore the relationship between humans and nature.

Interest in the study of fractal geometry was sparked by Mandelbrot's book Fractals: Form, Chance and Dimension (1977). Mandelbrot defined a fractal as 'any curve or surface that is scale-independent' [R. Pawelec 1999, p. 363; B. Mandelbrot 1977, p. 148]. This property is referred to as self-similarity and means that any part of the curve, if enlarged to scale, would look identical to the entire curve. The transition from one scale to another can be represented as iterations of the scaling process. This gives rise to the need to understand and comprehend terms such as fractal geometry and fractal dimension.

Fractal geometry defines a specific set of objects exhibiting a high degree of self-similarity. Peitgen and Richter (1986) explain, as Ostwald quotes, '(...) fractal geometry is defined by a repetitive or iterative feedback structure that produces a type of geometric phenomenon called scaling or characteristic irregularity.' [M.J. Oswald 2013, p. 648]. The concept of self-similarity lies in the concept of scale, because 'scaling is a property whereby a figure, examined at increasingly finer scales, is perceived as self-similar; or that at different scales, the object in question tends to resemble itself.' [M.J. Oswald 2013, p. 648]. Both definitions emphasise the shape of fractals as geometric phenomena, i.e. iterative feedback structures with details at every scale, such as clouds, trees, inflorescences, etc. So it is mainly about geometric shape.

Fractal geometry is analysed using visual methods. This involves observing an architectural object in order to identify the self-similarity of its elements, which are in fact fractal patterns or representations of mathematical fractal forms. Visual analysis helps to determine whether fractal patterns are present (e.g., Sierpinski carpet, Menger sponge, etc.) in projections, elevations, and other significant parts of the object (Fig. 53). Noticing such fractal properties helps in studying the object itself. If fractal patterns are found in certain repetitive elements, such as cascading, arched and striped forms, they can be compared to known fractal shapes (e.g., Julia sets, Koch snowflakes, etc.). Visual studies use simple techniques applied in architectural composition analyses. The projections of the object, its cross-sections and usually its façade are considered, as they most fully reflect the architecture of the building. However, in historical objects, subsequent changes may significantly alter its appearance, but the object retains its load-bearing and architectural structure. Visual inspection of a building can be carried out by architects and viewers with artistic sensitivity who are able to recognise the main artistic features of an architectural object.

The *fractal dimension* is a topological measure of how much space an object fills [M.J. Oswald 2013, p. 648]. Hence, fractal dimension can be understood as a property of fractal geometry (irregular objects, their shape) that fills space. Mandelbrot proposed seven main types of methods for calculating surface roughness [M.J. Oswald 2013, p. 649]. One of these is the box counting method, which is most commonly used to describe the roughness of building facades and projections.

The fractal dimension of an architectural form is determined based on the box counting method proposed by Carl Bovill in 1996. This method is based on the fact that the fractal dimension visually expresses the degree of 'roughness' and 'irregularity' of the structure, which determines the degree of complexity of the object. It is applied as follows: a grid of a specific size (S1) and number of cells containing image elements is superimposed on the image, which can then be calculated (N for s1). Next, the grid size is reduced (S2) and the number of cells is counted again (N for s2). The fractal dimension between the two scales is then calculated by the relationship between the difference in the number of occupied fields and the inverse of the difference in mesh sizes. This calculation can be expressed using mathematical equations, where S is the grid size and N is the number of cells that overlap with the image details [C. Bovill 1996, pp. 41–43].

In 1994, Batty and Longley applied the box counting method to architectural and urban analysis [M. Batty & P. Longley 1994]. However, Carl Bovill must be considered the first to have thoroughly researched architectural objects using this method in 1996. He analysed the fractal properties of the projections and elevations of several canonical buildings [C. Bovill 1996]

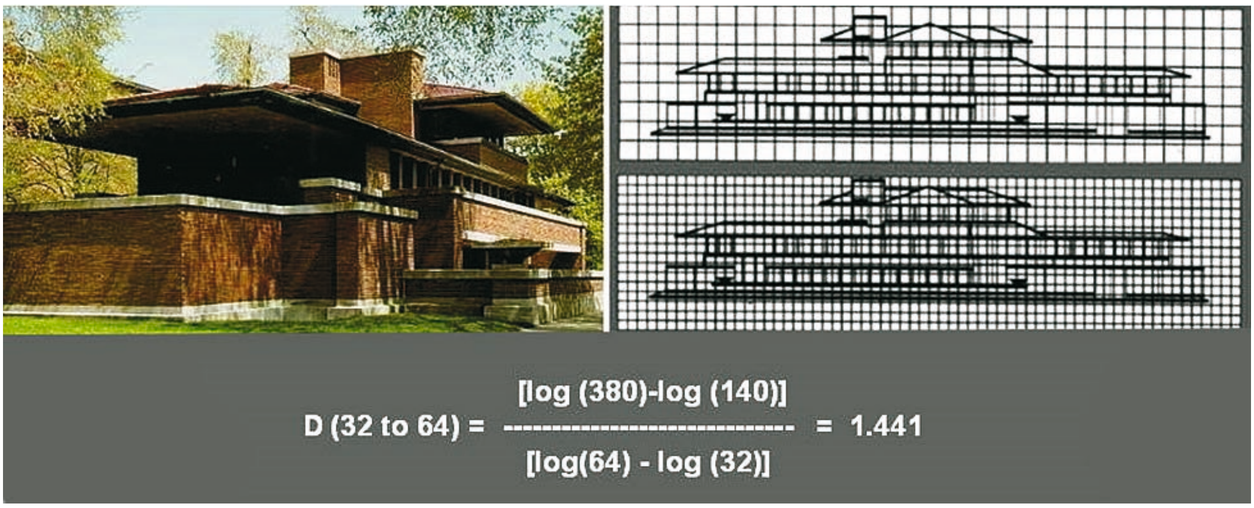


Fig. 54. Carl Bovill, box-counting analysis of fractal properties of Frank L. Wright's Robie House, 1996; source: Bovill 1996

(Fig. 54). Today, the fractal dimension of an architectural form can be easily determined using the standard FracLac programme for images (J), which enables fractal analysis of projections, cross-sections and elevations [A.L. Karperien 2007]. Using this programme, it is easy to specify the fractal dimension of a complex image that cannot be described by other traditional methods. The programme counts the number of fields or cells that have been touched or covered by the contour of the analysed object in the grid. The second step is to reduce the size of the grid and count the boxes again, and the process is repeated at multiple grid scales. The first grid (number of boxes counted) and the second grid (number of boxes counted) are compared. By plotting a log-log graph for each grid size, the slope of the resulting line is measured, which is a measure of box counting (D). When the process is repeated a sufficient number of times, the data is plotted and the average slope of the resulting line is the estimated fractal [A.L. Karperien 2007].

In biomimetic design, fractal geometry imposes structural and spatial articulation, forces the pursuit of self-similarity and scaling of components, as well as their hierarchy and distribution, and manages the means of artistic expression of the designed object. The fractal dimension, on the other hand, draws attention to maintaining consistency between the visual and non-visual projections of the object in relation to the whole form and its parts and surroundings. Analyses of the fractal dimension allow us to explain why the impact of architectural works is what it is, what its beauty consists of – a canonical issue in architectural theory [W. Tatarkiewicz 1973, p. 6]. They prove that perceived harmony or disharmony results, for example, from a lack of consistency between projections, elevations and cross-sections. Fractal dimension analyses are particularly important when it comes to historic buildings undergoing reconstruction or restoration. Furthermore, fractal analyses allow us to identify common structural patterns that underlie the organisation of nature, art and human perception.

CONCLUSIONS

For several decades, mathematics, physics, biology and computer science have been intertwined with architecture in an attempt to create a common basis for design in a way that allows natural and artificial environments to interact with each other in the best possible way.

The instrumentalisation of natural processes, from which form emerges, has opened up new design possibilities on the path to achieving this goal. Currently, design tools, known as morphogenetic, allow designers to approximate the principles and solutions found in Nature's designs.

Although the goals of plants, insects and animals differ from those of humans, these creations share characteristics that are of interest to humans: they are the most efficient in terms of materials, function and energy, and they are best adapted to changing living conditions. Above all, evolutionary development and exchange between biological form and the environment encourage the use of these processes as tools for designing eco-efficient artificial architectural forms, especially in an era of advancing global climate change.

The understanding of ecology developed in the second half of the 20th century should change the way we approach the built environment. Complex architectural forms that respond to climate change require new tools and new design strategies. This can be achieved by combining information and material processes. Organisms have many stable states that combine changing spatial requirements with appropriate formal and structural articulation.

In order to survive, biological structures evolve in an effort to build highly complex systems designed to provide optimal solutions for any given requirements. Every organism and form of life emerges from a process of evolutionary self-organisation. The development of IT design tools and the instrumentalisation of the principles of morphogenesis are therefore justified in becoming an attractive model to follow in architectural and urban design – all the more so because the aesthetics of natural forms have always been accepted by the public.

In the 21st century, the instrumentalisation of natural processes from which form emerges has opened up new design possibilities that are changing the existing understanding of the concept of mimesis in architecture and art. It is no longer about reflecting nature and its principles or imitating the shapes of Nature's creations, as Aristotle wanted. Instead, it is important to embed biological processes and functionalities similar to those of organisms in buildings for the mutual benefit of humans and the biological environment.

The development and popularisation of morphogenetic design tools integrated with CAD systems is a promising alternative for future architecture focused on climate change and sustainable development.

LITERATURE

1. ***A ton for Thompson's tome*** **(2017)**, "Nature Physics", Vol. 13, 315.
2. **Aurenhammer F. (1991)**, *Voronoi diagrams – a survey of a fundamental geometric data structure*,

“ACM Computing Surveys (CSUR)”, Vol. 23, No. 3, pp. 345–405.

3. **Bäck T., Fogel D.B. (1997)**, *Handbook of Evolutionary Computation*, Oxford University Press.
4. **Batty M., Longley P. (1994)**, *Fractal-Cities: A Geometry of Form and Function*, (U.S. ed.), Academic Press Limited, London.
5. **Bažant Z.P., Zhou Y. (2001)**, *Why did World Trade Center collapse – Simple Analysis*, “International Journal of Structural Stability and Dynamics”, Vol. 1, No. 4, pp. 603–615.
6. **Betsky A. (1990)**, *Violated Perfection: Architecture and the Fragmentation of the Modern*, Rizzoli, New York.
7. **Bollinger K., Grohmann M., Tessmann O. (2008)**, *Form, Force, Performance. Multi-Parametric Structural Design*, “Architectural Design”, Vol. 78, No. 2, pp. 20–25.
8. **Bovill C. (1996)**, *Fractal geometry in Architecture and Design*, Design Science Collection, Springer Verlag GMBH.
9. **Brassel K.E., Reif D. (1970)**, *A Procedure to Generate Thiessen Polygons*, “Geographical Analysis”, Vol. 11, No. 3, pp. 289–303.
10. **Brown D.G. (2006)**, *Agent-Based Models* [in:] H. Geist (Ed.), *Our Earth’s Changing Land: An Encyclopedia of Land-Use and Land-Cover Change*, Greenwood Press, Westport, pp. 7–13.
11. **Chakrabarti A., Shea K., Stone R., Cagan J., Campbell M., Vargas-Hernandez N., Wood K.L. (2011)**, *Computer-Based Design Synthesis Research: An Overview*, “Journal of Computing and Information Science in Engineering”, Vol. 11, No. 2, 021003.
12. **Chen Y. (2018)**, *Swarm Intelligence in Architectural Design* [master disertation], University of California, Berkeley, https://www.yuxingc.com/flocking-incubator [accessed: 17.06.2024].
13. **Coates P., Broughton T., Jackson H. (1999)**, *Exploring Three-Dimensional Design Worlds Using Lindenmayer Systems and Genetic Programming* [in:] **P.J.** Bentley (Ed.), *Evolutionary Design by Computers*, Morgan Kaufmann Publishers, San Francisco, pp. 323–341.
14. **Coates P., Derix Ch., Krakhofer I.S.P., Karanouh A.-M. (2005)**, *Generating Architectural Spatial Configurations. Two Approaches Using Voronoi Tessellations and Particle Systems* [conference paper], 8th Generative Art Conference (GA2005) 15–17 December, Milan, https://www.researchgate.net/publication/47529516_Generating_architectural_spatial_configurations_Two_approaches_using_Voronoi_tessellations_and_particle_systems [accessed:].
15. **Coates P., Thum R. (2000)**, *Agent Based Modelling*, University of Greenwich, London.
16. **Cummings F.W. (1989)**, *On Surface Geometry Coupled to Morphogen*, “Journal of Theoretical Biology”, Vol. 137, No. 2, pp. 215–219.
17. **De Wolf T., Holvoet T. (2004)**, *Emergence Versus Self-Organisation: Different Concepts but Promising When Combined* [in:] S.A. Brueckner, G. di Marzo Serugendo, A. Karageorgos, R. Nagpal (Eds.), *Engineering Self-Organising Systems: Methodologies and Applications. Conference Proceedings*, pp. 1–15.
18. **DeLanda M. (2011)**, *Philosophy and Simulation: The Emergence of Synthetic Reason*, Continuum, London-New York.
19. **Dorigo M., Stützle T. (2004)**, *Ant Colony Optimization*, The MIT Press, Cambridge, MA.
20. **Farmer J.D., Packard N.H., Perelson A.S. (1986)**, *The Immune System, Adaptation, and Machine Learning*, “Physica D: Nonlinear Phenomena”, Vol. 22, No. 1–3, pp. 187–204.
21. **Figielska E. (2006)**, *Algorytmy Ewolucyjne i Ich Zastosowania*, “Zeszyty Naukowe Warszawskiej Wyższej Szkoły Informatyki”, No. 1, pp. 81–92.
22. **Fogel M.J., Owens L.J., Walsh A.J. (1966)**, *Artificial Intelligence through Simulated Evolution*, John Wiley & Sons, New York.
23. **Frazer J.H. (1995)**, *An Evolutionary Architecture*, Architectural Association, London.
24. **Gardner M. (1970)**, *The Fantastic Combinations of John Conway’s New Solitaire Game of Life*, “Scientific American”, Vol. 223, pp. 120–123.
25. **Goldberg D.E. (2009)**, *Algorytmy Genetyczne i Ich Zastosowania*, Wydawnictwo WNT, Warszawa.
26. **Gutenbaum J. (2003)**, *Modelowanie Matematyczne Systemów*, 3rd ed., Akademicka Oficyna Wydawnicza EXIT, Warszawa.
27. **Hansmeyer M. (2003)**, *L-Systems in Architecture*, https://www.michael-hansmeyer.com/l-systems [accessed: 17.05.2024].
28. **Harrison L.G., Kolář M. (1988)**, *Coupling between Reaction Diffusion and Expressed Morphogenesis*, “Journal of Theoretical Biology”, Vol. 130, No. 4, pp. 493–515.
29. **Hensel M., Menges A., Weinstock M. (2010)**, *Emergent Technologies and Design: Towards a Biological Paradigm for Architecture*, Routledge.
30. **Herr Ch.M. (2015)**, *Second-Order Cellular Automata to Support Designing*, “Kybernetes”, Vol. 44, No. 8–9, pp. 1251–1261.
31. **Heylighen F. (1989)**, *Self-organisation, Emergence and the Architecture of Complexity,* Proceedings of 1st European Conference on System Science, http://pespmc1.vub.ac.be/Papers/SelfOrgArchComplexity.pdf [accessed: 17.06.2024].
32. **Hofmann C. (2012)**, *Animal Behavior: Collective Movement and Shapeshifting*, https://crhofmann.com/2012/11/13/animal-behavior-collective-movement-and-shapeshifting [accessed: 19.10.2024].
33. **Holland J.H. (1975)**, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI.
34. **Holland J.H. (1992)**, *Adaptation in Natural and Artificial System. An Introductory Analysis with Applications to Biology, Control*
35. **Hornby G.S., Pollack J.B. (2001)**, *The Advantages of Generative Grammatical Encodings for Physical Design* [in:] *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, Seoul, Korea (South), pp. 600–607.

36. **Ilachinski A. (2001)**, *Cellular Automata: A Discrete Universe*, World Scientific Publishing, Singapore.
37. **Januszkiewicz K. (2023)***, Generative And Evolutionary Models In the Design of Architectural Form - Insights from History*, “Architecturae et Artibus”, Vol. 15, No. 3, pp. 11–38.
38. **Januszkiewicz K. (2010)**, *O Projektowaniu Architektury w Dobie Narzędzi Cyfrowych. Stan Aktualny i Perspektywy Rozwoju*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław.
39. **Januszkiewicz K. (2012a)**, *Komputery i Architektura. Jak Było na Początku*, “Archivolta”, Vol. 1, No. 53, pp. 56–65.
40. **Januszkiewicz K. (2012b)**, *Komputery i Architektura. Systemy i Narzędzia Generatywne*, “Archivolta”, No. 4, pp. 44–51.
41. **Januszkiewicz K. (2013)**, *Komputery i Architektura. Naturalne Procesy Formotwórcze, Matematyka i Architektura*, “Archivolta”, No. 2, pp. 42–51.
42. **Januszkiewicz K., Paszkowska-Kaczmarek N. (2023)**, *Generative and Evolutionary Models in the Design of Architectural Form –*
43. **Jaśkiewicz T. (2013)**, *Towards a Methodology for Complex Adaptive Interactive Architecture* [PhD dissertation], Delft University of Technology.
44. **Jencks Ch. (1995)**, *The Architecture of the Jumping Universe*, Academy Edition, London.
45. **Kalay Y.E., Mitchell W.J. (2004)**, *Architecture’s New Media: Principles, Theories, And Methods of Computer-Aided Design*, The MIT Press.
46. **Kang K.A., Yoon J.E. (2008)**, *A Study on the Application of the Voronoi Diagram on Digital Space*, “Journal of Korean Institute of Interior Design”, Vol. 17, No. 3, pp. 155–164.
47. **Karaboga D., Basturk B. (2008)**, *On the Performance of Artificial Bee Colony (ABC) Algorithm*, “Applied Soft Computing”, Vol. 8, No. 1, pp. 687–697.
48. **Karperien A.L. (2007)**, *The FracLac. Faculty of Science*, http://rsb.info.nih.gov/ij/-plugins/fraclac/FLHelp/TheoryStartUpScreen.htm [accessed: 17.09.2024].
49. **Kauffman S.A. (1993)**, *The Origins of Order: Self-Organisation and Selection in Evolution*, Oxford University Press, New York-Oxford.
50. **Kauffman S.A. (1996)**, *At Home in the Universe: The Search for Laws of Self-Organisation and Complexity*, Oxford University Press, New York-Oxford.
51. **Kennedy J., Eberhart R.C., Shi Y. (2001)**, *Swarm Intelligence*, Morgan Kaufmann Publishers.
52. **Koza J.R. (1992)**, *Genetic Programming, or the Programming of Computers by Means of Natural Selection*, The MIT Press, Cambridge, MA.
53. **Krawczyk R.J. (2002)**, *Architectural Interpretation of Cellular Automata*, 5th International Conference Generative Art Conference Proceedings, Milano.
54. **Kudrewicz J. (2007)**, *Fraktale i Chaos*, Wydawnictwo WNT, Warszawa.
55. **Kurmann D. (2001)**, *Design in Space and Time* [in:] M. Engeli (Ed.), *Bits and Spaces: Architecture and Computing for Physical, Virtual, Hybrid Realms; 33 Projects by Architecture and CAAD*, Birkhäuser, Basel-Boston-Berlin.
56. **Leach N. (2009)**, *Swarm Urbanism*, “Architectural Design”, Vol. 79, No. 4, pp. 56–63.
57. **Ligmann-Zielinska A., Sun L. (2010)**, *Time Dependent Variance-Based Sensitivity Analysis of Development Aggregation Generated by Heterogeneous Land Use Agents*, https://project.geo.msu.edu/stsa/pubs/Accuracy2010.pdf [accessed: 17.09.2024].
58. **Mandelbrot B.B. (1977)**, *Fractals: Form, Chance and Dimension*, Freeman, San Francisco.
59. **Marzec Ch.J. (1999)**, *A Pragmatic Approach to Modeling Morphogenesis*, “Journal of Biological Systems”, Vol. 7, No. 3, pp. 333–351.
60. **McCormack J.P., Dorin A., Innocent T.Ch. (2005)** *Generative Design: A Paradigm for Design Research* [in:] J. Redmond, D. Durling, A. de Bono (Eds.), *Futureground*, Vol. 2, Monash University, Caulfield East Vic. Australia.
61. **McKay A., Chase S.C., Shea K., Hin G., Chau H.H. (2012)**, *Spatial Grammar Implementation: From Theory to Useable Software*, “Artificial Intelligence for Engineering Design, Analysis and Manufacturing”, Vol. 26, pp. 143–159.
62. **Menges A. (2006a)**, *Instrumental Geometry*, “Architectural Design”, Vol. 76, No. 2, pp. 42–53.
63. **Menges A. (2006b)**, *Polymorphism*, Architectural Design, Vol. 76, No. 2, pp. 78–87.
64. **Michalewicz Z. (2003)**, *Algorytmy Genetyczne + Struktury Danych = Programy Ewolucyjne*, WydawnictwoWNT, Warszawa.
65. **Millonas M.M. (1994)**, *Swarms, Phase Transitions, and Collective Intelligence* [in:] C.G. Langton (Ed.), *Artificial Life III*, *Proceedings of the Workshop on Artificial Life*, 2nd ed., Reading, MA.
66. **Müller P., Wonka P., Haegler S., Ulmer A., van Gool L. (2006)**, *Procedural Modeling of Buildings*, “ACM Transactions on Graphics”, Vol. 25, No. 3, pp. 614–623.
67. **Noser H., Rudolph S., Stucki P. (2001)**, *Physics-Enhanced L-Systems*, International Conference in Central Europe on Computer Graphics and Visualization, https://api.semanticscholar.org/CorpusID:15708331 [accessed:].
68. **Nowak A. (2015)**, *Application of Voronoi Diagrams in Contemporary Architecture and Town Planning*, “Challenges of Modern Technology”, Vol. 6, No. 2, pp. 30–34.
69. **O’Neill M., Mcdermott J., Byrne J., Brabazon A. (2009)**, *Shape Grammars and Grammatical Evolution for Evolutionary Design* [in:] *GECCO’ 09: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, pp. 1035–1042.
70. **O’Reilly U.-M., Hemberg M. (2007)**, *Integrating Generative Growth and Evolutionary Computation for form Exploration*, “Genetic Programming and Evolvable Machines”, Vol. 8, No. 2, pp. 163–186.
71. **Ostwald M.J.** (**1998)**, *Fractal Traces: Geometry and the Architecture of Ushida Findlay* [w:] Leon van Schaik (Ed.), *Ushida Findlay*, 2G, Barcelona, pp. 136–143.
72. **Ostwald M.J. (2001)**, *“Fractal Architecture”: Late Twentieth Century Connections Between*

Architecture and Fractal Geometry, “Nexus Network Journal”, Vol. 3, No. 1, pp. 73–84.

73. **Ostwald M.J. (2013)**, *The Fractal Analysis of Architecture: Calibrating the Box-Counting Method Using Scaling Coefficient and Grid Disposition Variables*, “Environment and Planning B: Planning and Design”, Vol. 40, No. 4, pp. 644–663.
74. **Oteíza P., Köppen M., Concha M.L., Heisenberg C-P. (2008)**, *Origin and Shaping of the Laterality Organ in Zebrafish*, *“Development”,* Vol. 135, No. 16, pp. 2807–2813.
75. **Özkar M., Stiny G. (2009)**, *Shape Grammars*, paper presented at ACM SIGGRAPH 2009 Courses, SIGGRAPH’09, New Orleans, LA, USA.
76. **Park J., Ji S., Kim E., Jun H. (2008)**, *The Application of Voronoi Diagram into the Space Planning for Urban Design*, The 7th International Symposium on Architectural Interchanges in Asia, ISAIA 2008, pp. 524–528.
77. **Paszkowska-Kaczmarek N. (2022)**, *Problem mimesis w architekturze w dobie morfogenetycznych narzędzi projektowania*, rozprawa doktorska niepublikowana, promotor K. Januszkiewicz, ZUT, Szczecin.
78. **Pawelec R. (1999)**, *Wieża Babel. Słownik Wyrazów Obcych*, Wydawnictwo Szkolne PWN, Warszawa.
79. **Prigogine I. (1967)**, *Introduction to Thermodynamics of Irreversible Processes*, 3rd ed., John Wiley & Sons, Hoboken.
80. **Prusinkiewicz P., Lindenmayer A. (1990)**, *The Alogirthmic Beauty of Plants*, Springer-Verlag, New York.
81. **Rechenberg I. (1973)**, *Evolutionstrategie: Optimierung Technisher Systeme nach Prinzipien des Biologischen Evolution*, Fromman-Hozlboog Verlag, Stuttgart.
82. **Reynolds C.W. (1987)**, *Flocks, Herds, and Schools: A Distributed Behavioral Model*, “Computer Graphics”, Vol. 21, No. 4, pp. 25–34.
83. **Rokicki W., Gawell E. (2016)**, *Voronoi Diagrams – Architectural and Structural Rod Structure Research Model Optimization*, “MAZOWSZE Studia Regionalne”, Vol. 19, pp. 155–164.
84. **Rosenberg L.B. (2015)**, *Human Swarms, a Real-Time Method for Collective Intelligence* [in:] *Proceedings of the European Conference on Artificial Life*, The MIT Press, pp. 658–669.
85. **Roudavski S. (2009)**, *Towards Morphogenesis in Architecture*, “International Journal of Architectural Computing”, Vol. 7, No. 3, pp. 345–374.
86. **Salingaros N.A. (1999)**, *Architecture, Patterns, and Mathematics*, “Nexus Network Journal”, Vol. 1, pp. 75–86.
87. **Salingaros N.A. (2008)**, *Connecting the Fractal City*, plenary talk at the 7o Laboratorio Internazionale d’Architettura, Università degli Studi Mediterranea di Reggio Calabria, Italy, 22 September 2008, pp. 10–11.
88. **Schwefel H.-P. (1981)**, *Numerical Optimization of Computer Models*, John Wiley & Sons, Chichester.
89. **Shannon C.E., Weaver W.W.** (1963), *The Mathematical Theory of Communication*, 5 ed., University of Illinois Press.
90. **Shelden D.R. (2002)**, *Digital Surface Representation and the Constructibility of Gehry’s Architecture* [PhD disseration], Massachusetts Institute of Technology.
91. **Silver M. (2006)**, *Building Without Drawings: Automason Ver. 1.0*, “Architectural Design”, Vol. 76, No. 4, pp. 46–51.
92. **Simon H.A. (1996)**, *The Architecture of Complexity*, The Sciences of the Artificial, The MIT Press, Cambridge, MA.
93. **Spaeth B.A., Menges A. (2011)**, *Performative Design for Spatial Acoustics: Concept for an Evolutionary Design Algorithm Based on Acoustics as Design Driver*, Presented at: *Respecting Fragile Places, 29th eCAADe Conference*, Ljubliana, Slovenia, 21–24 September 2011. Published in: T. Zupančič-Strojan, M. Juvančič, Š. Verovšek, , A. Jutraž (Eds.), *Respecting Fragile Places: Proceedings of the 29th Conference on Education in Computer Aided Architectural Design in Europe, Ljubljana, Slovenia, September 21–24, 2011*, eCAADe, Brussels, pp. 461–468.
94. **Spirov A.V. (1993)**, *The Change in Initial Symmetry in the Pattern-Form Interaction Model of Sea Urchin Gastrulation*, “Journal of Theoretical Biology”, Vol. 161, No. 4, pp. 491–504.
95. **Stankowski W. (2019)**, *Morfogeneza Powierzchni Litosfery. Aspekty Geomorfologii i Geologii. Aspekty Geomorfologii i Geologii*, Bogucki Wydawnictwo Naukowe, Warszawa.
96. **Stiny G. (1976)**, *Two Exercises in Formal Composition*, “Environment and Planning B: Planning and Design”, Vol. 3, pp. 187–210.
97. **Stiny G. (1977)**, *Ice-Ray: A Note on the Generation of Chinese Lattice Designs*, “Environment and Planning B: Planning and Design”, Vol. 4, pp. 89–98.
98. **Stiny G., Gips J. (1972)**, *Shape Grammars and the Generative Specification of Painting and Sculpture* [in:] O.R. Petrocelli (Ed.), *The Best Computer Papers of 1971*, Auerbach, Philadelphia, pp. 125–135.
99. **Stiny G.**, **Mitchell W.J. (1978)**, *The Palladian grammar*, “Environment and Planning B: Planning and Design”, Vol. 5, pp. 5–18.
100. **Szymski A.M. (1997)**, *Twórczość Architektoniczna. Wstęp do Teorii Projektowania Systemowego (Elements of System Designing Theory*), Prace Naukowe Politechniki Szczecińskiej, Instytut Architektury i Planowania Przestrzennego, Szczecin.
101. **Tatarkiewicz W. (1973)**, *Przedmowa* [in:] J. Żórawski, *O Budowie Formy Architektonicznej*, Arkady, Warszawa, pp. 5–10.
102. **Tepavčević B., Stojaković V. (2012)**, *Shape Grammar in Contemporary Architectural Theory and Design*, “Facta Universitatis, Series: Architecture and Civil Engineering”, Vol. 10, No. 2, pp. 169–178.
103. **Thompson D.W. (1961)**, *On Growth and Form*, Cambridge University Press (1st ed. – 1917).
104. **Trunz E.** (Ed.) **(1960)**, *Goethes Werke*, Wegner Verlag, Hamburg, Vol. 13.
105. **Turing A.N. (1952)**, *The Chemical Basis of Morphogenesis*, “Philosophical Transactions of the Royal Society B”, Vol. 237, No. 641, pp. 37–72.

106. **Ullrich W. (1973)**, *Zoopsychologia*, Wydawnictwo Naukowe PWN, Warszawa.
107. **Ushida E., Findlay K. (1996)**, *S-Project Parallel Landscapes*, Toto, Japan.
108. **Vehlken S. (2014)**, *Computational Swarming: A Cultural Technique for Generative Architecture*, *Dynamics of Data-Driven Design*, "Footprint", Vol. 8, No. 15, pp. 9–24.
109. **von Neumann J. (1951)**, *The General and Logical Theory of Automata*, John Wiley & Sons, London.
110. **von Neumann J. (1961)**, *The General and Logical Theory of Automata and Theory of Self-Reproducing Automata* [in:] A.H. Taub (Ed.), *John von Neumann*. *Collected Works*, Pergamon Press.
111. **Weaver W. (1948)**, *Science and Complexity*, "American Scientist", Vol. 36, No. 4, pp. 536–544.
112. **Weiner N. (1971)**, *Cybernetyka czyli Sterowanie i Komunikacja w Zwierzęciu i Maszynie*, Wydawnictwo Naukowe PWN, Warszawa.
113. **Weinstock M. (2004)**, *Morphogenesis and the Mathematics of Emergence*, "Architectural Design", Vol. 74, No. 3, pp. 11–17.
114. **Weinstock M., Menges A., Hensel M.U. (2004a)**, *Fit Fabric: Versatility through Redundancy and Differentiation*, "Journal of Architectural Design", Vol. 169, pp. 40–47.
115. **Weinstock M., Menges A., Hensel M.U. (2004b)**, *Fit Fabric: Versatility Through Redundancy and Differentiation*, "Architectural Design", Vol. 74, No. 3, pp. 40–45.
116. **Whitehead A.N. (1987)**, *Nauka i świat nowożytny*, Wydawnictwo Znak, Kraków.
117. **Wolfram S. (1984)**, *Cellular Automata as Models of Complexity*, "Nature", Vol. 311, pp. 419–424.
118. **Wolfram S. (2002)**, *A New Kind of Science*, Wolfram Media, Illinois.
119. **Wong S.S.Y, Chan K.C.C. (2009)**, *EvoArch: An Evolutionary Algorithm for Architectural Layout Design*, "Computer-Aided Design", Vol. 41, No. 9, pp. 649–667.
120. **Wooldridge M. (2001)**, *Introduction to Multiagent Systems*, John Wiley & Sons, West Sussex.
121. **Yang X-S. (2010)**, *A New Metaheuristic Bat-Inspired Algorithm* [in:] J.R. González, D.A. Pelta, C. Cruz, G. Terrazas N. Krasnogor (Eds.), *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, Studies in Computational Intelligence Series, Vol. 284, Springer, Berlin Heidelberg, pp. 65–74.
122. **Yi Y.-K., Kim H. (2015)**, *Agent-Based Geometry Optimization with Genetic Algorithm (GA) for Tall Apartment's Solar Right*, *Solar Energy*, "Elsevier", Vol. 113, pp. 236–250.
123. **Zaha Hadid Architects, Hadid Z. (2007)**, *National Performing Arts Center, Kaohsiung*, Archilovers 25 Years, https://www.archilovers.com/projects/2854/national-performing-arts-center-kaohsiung.html [accessed: 17.05.2024].