Computational Design Informed by Natural Systems

Sevil Yazıcı[1]

0000-0002-0664-4494[1]

[1] Istanbul Technical University, Graduate School of Science, Engineering, and Technology, Department of Informatics, Architectural Design Computing, Istanbul, Turkey

Animate and inanimate matters in nature are evaluated within systems' unity. Natural systems represent order and balance defined by computation. Mathematicians, biologists, material scientists and professionals from other fields investigate natural systems for problem-solving. Although nature has been used as a reference since historical periods, there are improper uses on how to integrate nature efficiently into the design process. The term biomimicry is used today by transcending its meaning from imitating nature towards learning from its intelligence. There is a necessity that architectural design students develop their skills on computation and evaluate natural systems from an analytical point of view to apply their finding in creative design solutions. This research is about an elective course on biomimicry developed for undergraduate architectural design education. The methodology consists of three stages, including investigation of the natural systems (1), the abstraction of the natural systems and extracting the system parameters (2), implementing the parameters in the computational design model (3). The proposed study was implemented from 2018 to 2020 into the student projects, of which outputs are discussed and grouped under four categories, including organization-, performance-, process- and motion-based computation. By examining the results, it is determined that the students gained skills in computational design and their awareness related to the natural systems were increased.

Keywords: Natural Systems, Computation, Algorithms, Architectural Design Education.

Received: 24.08.2020
Accepted: 25.09.2020

Corresponding Author:
sevilyazici@itu.edu.tr

Yazici, S. (2020). Computational Design Informed by Natural Systems. JCoDe: Journal of Computational Design, 1(3), 01-16.

Doğal Sistemlerle Bilgilendirilmiş Hesaplamalı Tasarım

Sevil Yazıcı[1]

0000-0002-0664-4494[1]

[1] İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilişim Anabilim Dalı, Mimari Tasarımda Bilişim, İstanbul, Türkiye

Doğada canlı ve cansız varlıkların tümü sistem bütünlüğü içinde değerlendirilmektedir. Doğal sistemler hesaplama ile tanımlanan düzen ve dengeyi temsil eder. Matematikçiler, biyologlar, malzeme bilimcileri ve farklı alanlardan profesyoneller problem çözme amaçlı olarak doğal sistemleri araştırmaktadır. Doğa, tarihsel dönemlerden beri tasarım sürecinde referans olarak kullanılsa da, doğanın tasarım süreciyle verimli bir şekilde nasıl bütünleştirileceğine dair yanlış kullanımlar bulunmaktadır. Biyomimesis terimi, günümüzde doğayı taklit etme anlamını aşarak, doğanın zekasından öğrenme anlamında kullanılmaktadır. Mimari tasarım öğrencilerinin hesaplama becerilerini geliştirerek, doğal sistemleri analitik bakış açısıyla inceleme ve bulguları yaratıcı tasarım çözümlerinde uygulama ihtiyacı bulunmaktadır. Bu araştırma, mimarlık lisans eğitiminde uygulanmak üzere geliştirilmiş biyomimesis konulu bir seçmeli ders üzerinedir. Yöntem, doğal sistemlerin incelenmesi (1), doğal sistemlerin soyutlanması ve sistem parametrelerinin çıkarılması (2), parametrelerin hesaplamalı tasarım modelinde uygulanması (3) dahil olmak üzere üç aşamadan oluşmaktadır. Önerilen çalışma, 2018 yılından 2020 yılına dek öğrenci projelerinde uygulanmış olup, çıktılar organizasyon-, performans-, süreç- ve hareket-tabanlı hesaplama olmak üzere dört grupta incelenmiştir. Sonuçlar irdelendiğinde, öğrencilerin hesaplamalı tasarım konusunda beceri kazandıkları ve doğada bulunan sistemler hakkında farkındalıklarının arttığı belirlenmiştir.

Anahtar Kelimeler: Doğal Sistemler, Hesaplama, Algoritmalar, Mimari Tasarım Eğitimi.

Teslim Tarihi: 24.08.2020
Kabul Tarihi: 25.09.2020

Sorumlu Yazar:
sevilyazici@itu.edu.tr

Yazici, S. (2020). Doğal Sistemlerle Bilgilendirilmiş Hesaplamalı Tasarım. JCoDe: Journal of Computational Design, 1(3), 01-16.

1. INTRODUCTION

Architects commonly use metaphors or analogies based on nature during their design processes. There have been continuous attempts to use nature as a reference as seen in well-known examples from historical periods, such as Crystal Palace by Joseph Paxton built in 1851 and columns of Johnson Wax Building by Frank Lloyd Wright built in 1939. Paxton and Wright were both inspired by water lily pads (Navarro-Mateu and Cocho-Bermejo, 2020). There were also some pioneers in the past, such as Gaudi, Fuller and Otto, who observed and interpreted nature wisely by developing analytical solutions for design problems. Physical models were mostly used to understand the behavior of the systems holistically.

The term biomimicry became popular and spread by J. Benyus (1997). Despite biomimicry is coined by two words as bio and mimicking, the meaning of the word goes beyond mimicking biological systems only. It is about learning from the intelligence of nature. Thus, the relationship between architecture and nature can be defined by two main approaches, including imitating the natural forms or learning from nature, in terms of its generative and behavioral properties (Selçuk and Sorguç, 2009). Biomimetic architecture is filled with a significant number of projects that are mostly imitating the formal features of natural systems. This approach does not obtain a deeper understanding of natural systems.

Architectural design is widely influenced by information technologies today. The intent is achieving a tectonic unity in design, where form, performance and material are integrated into one system, as observed in nature (Kolarevich and Klinger, 2008; Oxman, 2009). Design process obtains various stages, which differ by their details. During the process, problem-solving is identified by ill-defined problems that involve uncertainty, and well-defined ones that are for specific goals (Reitman, 1964; Suwa et. al., 1999). The computational design process is rational and critical parameters affecting design should be identified well at the beginning of the process. The use of biomimetic principles in the computational design process offers systems with high effectiveness and performance. These types of principles were implemented in a series of built projects and tested with state-of-the-art material assemblies (Schwinn, et. al, 2012; Krieg et al. 2012; Brugnaro et al. 2016; Castriotto, et. al, 2019).

An algorithm is defined by a process, consisting of a finite number of steps. Oxman (2017) described the term algorithmic thinking as a rule set, which uses computation for designing. Algorithms are generated for specific intents and solutions (Terzidis, 2006). Integration of computation, algorithmic logic, as well as scripting and visual programming languages into the architectural education were explored in the past (Kvan, et. al. 2004; Çolakoğlu and Yazar, 2007; Celani and Vaz, 2012). Digital design models and techniques, design theory and architectural discourse were also investigated, in terms of design pedagogy (Oxman, 2008). Since natural systems represent order and balance defined by computation, there were additional attempts to incorporate multi-faceted biomimetic principles with computational design into architectural curricula by using systems' thinking (Yazici, 2015) and by generating alternatives for space architecture (Varınlıoğlu, et. al. 2018). More studies should be undertaken in architectural design education, in terms of evaluating natural principles, understanding their computational logic and learning from them for their implementation into design problems. The research question remains unanswered: How can natural systems be used analytically for problem-solving in architecture and design, by improving computational design skills, as well as algorithmic thinking skills of undergraduate architecture students? Creativity in architectural design can be supported by using natural systems as a source for learning by using computation.

2. METHODOLOGY

The methodology of the course consists of three stages, including (1) investigation of the natural systems, (2) the abstraction of the natural system and extracting the critical parameters, and (3) implementing the parameters into the computational design model.

2.1 Investigation of the Natural Systems

In the first stage of the course, nature should be observed and analyzed based on its properties initially. As a part of the course, lectures are given covering the state of-the-art research and application projects in the field of biomimetic design systems. Students need to interpret nature, in terms of systems' thinking by discussing the terms, such as inputs and outputs, the transformation of inputs into the outputs, holism, goal seeking, differentiation and hierarchy (Yazici, 2011). Afterwards, students are asked to prepare a research project for a formation or an organization found in nature, such as an organism, inanimate entity or a phenomenon, by identifying the reason of their selection, parameters and rules of the system and by aiming to respond the question of, whether it can be applied to a design problem at the conceptual level.

2.2 The Abstraction of the Natural Systems and Extracting the Critical Parameters

In the second stage, students are required to investigate nature from a new perspective, in which parameters, rules and relationships of the system are identified. Students should explain selected research projects with diagrams; describe the system from an analytical point of view by introducing a computational framework. This process requires a certain level of abstraction, as well as simplification of the system since natural systems are highly complex. Thus, only critical parameters of the system are extracted, by serving to the goal of individual students.

As a part of the course, some precedent work and algorithms are investigated that are extensively used in the field, such as Voronoi and Fractal algorithms (Mandelbrot, 1977, p: 4), driven by the cell morphologies and growth in nature respectively. However, students are expected to investigate all possibilities in natural systems without providing prior constraints. The selected critical parameters may be associated with features related to the organization, such as geometrical principles and pattern; or performance, related to the structure and environment, such as wind flow and sun; or process, such as mathematical definitions; or motion, such as movement in X, Y, Z axes, rotation and twist.

2.2 Implementing the Parameters into the Computational Design Model

In the final stage, the mathematical properties of the natural systems should be understood and re-generated by the use of computation. By investigation and abstraction of the natural systems, computational design models can be established with critical parameters. The outputs may differ according to the principles extracted from nature, related to the organization, performance, process, and motion.

Introduction on the capabilities of the computer-aided design (CAD) methods, including algorithmic modelling, performance simulation, coding and three-dimensional (3D) geometric modelling, are given and related methods are discussed in the course. There is neither a specific computational tool nor a tutorial on particular software is provided. However, participants are required to explore tool ecologies flexibly and use them creatively in their processes. The research project in the first stage needs to be implemented into a computational framework. Figure 1 depicts the flowchart, which includes the stages of the methodology.

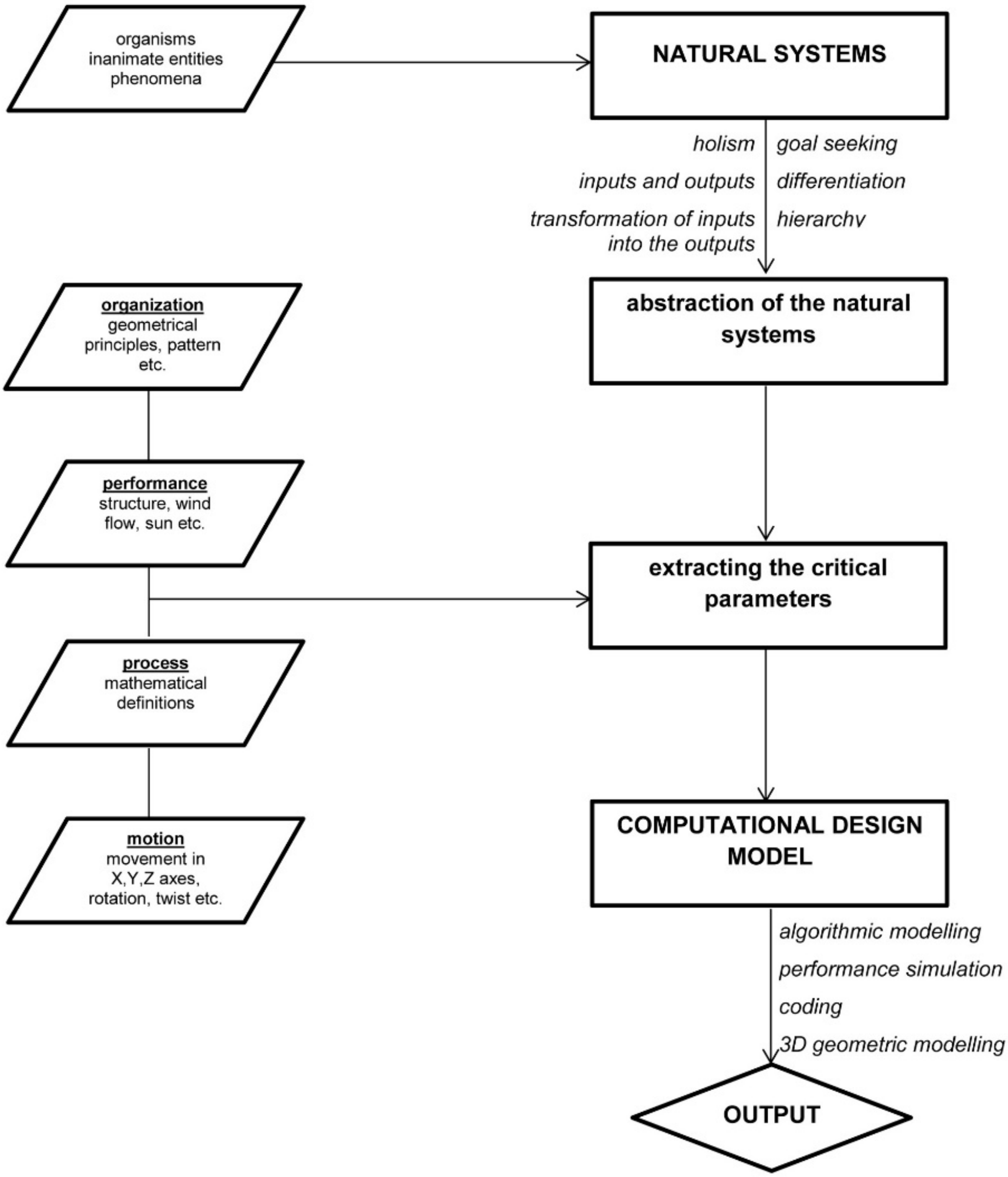


Figure 1: The methodology flowchart.

3. OUTPUTS: STUDENT PROJECTS

The methodology was implemented from the year 2018 to 2020 as an elective course. Since the course was open to the Faculty of Architecture and Design, students from different departments were able to join the course. Although the majority of the students were from the department of architecture, there were also participants from the departments of communication design and interior architecture. Selected seven output projects of students (**S.1-S.7**) were grouped under four, including organization-based, performance-based, process-based and motion-based computation. The groups were formed according to their characteristics on, how they translate natural principles into the computational design models. The computational tools, used in the student projects, were Rhinoceros for geometric modelling, Grasshopper (GH) for algorithmic modelling, Kangaroo GH and RhinoVAULT for form-finding, Autodesk Flow for Computational Fluid Dynamics (CFD) simulation, Processing for two-dimensional (2D) visual design and C# coding.

3.1 Organization-based Computation

Organization-based computation is driven by organizational principles found in natural systems. **S.1** used sea-corals, particularly brain corals, formed by the movements of the waves at the shores, which contain sedimental rocks. By erosion of water on the rocks, gaps are generated. They are filled with micro-creatures that use these parts for laying eggs and interbreeding. Corals act as buffers against large wave movements. The investigation of **S.1** was based on the geometry and growth of corals, identified by certain rules. The computational model was developed by interpreting the growth by algorithms, which resulted in the looped generations. The code developed at C# programming language enabled direct integration to the Rhinoceros 3D geometric modelling and Grasshopper (GH) algorithmic design environments. The application was implemented into subdivision logic of a 3D surface, which started from a flat surface and transformed sequentially into a subdivided mesh without having intersecting surfaces. The two-dimensional (2D) organization was translated into a plan layout to test as well.

3.2 Performance-based Computation

The projects under this group are based on the behaviour of the natural systems related to the performance criteria, such as structural or environmental performances, as presented by **S.2** and **S.3.**

Geomimicry uses geological process as a source for learning, similar to the term biomimicry that is used to examine nature for problem-solving. Landforms may vary from mountains, plains, valleys, to the canyons and deserts. Formation of land may continue thousands of years, as being informed by activities, such as erosion, plate tectonics, weathering and Aeolian processes. Dunes are made out of mixed size sand particles, formed by Aeolian or weathering processes. The formations of dunes occur in deserts and coast, where wind forces are active. The goal of **S.2** was to use environmental forces and landforms to extract critical parameters to generate a dwelling, which could be adapted to its surroundings, in extreme conditions, in deserts more specifically, Since dunes act as barriers against forces of wind and waves, the geometry of the dwelling was created accordingly, to increase the resistance of the mass against severe wind forces. The computational model was generated by using Kangaroo for GH, as physics-based spring system as the form-finding method. Additionally, Computational Fluid Dynamics (CFD) simulations were undertaken by Autodesk Flow software to assess the effects of wind on the geometry. **S.3** used spiral aloe, a type of succulent, of which leaves symmetrically grow in spirals. The plant can grow by obtaining up to 150 individual leaves, which can support and carry each other and create a self-supporting system. The participant asked the question of whether this feature could be translated into structural systems, in which neither additional supports nor fixtures might be necessary. The masonry structures were investigated further and tested by the use of RhinoVAULT, a plug-in developed for funicular form-finding of thin-shell structures in compression. The definition presented that the force distributions and optimization process could be controlled parametrically. The relationship between the geometry and active forces affecting the shell were explored through a series of tests.

3.3 Process-based Computation

This group investigates the natural processes, by aiming to understand its underlying mathematical principles, as presented by **S.4** and **S.5.**

The growth of multicellular systems represents similarities in nature. Complex systems found in nature are simplified by the use of Lindenmayer systems (L-systems), used by biologist, mathematicians and computer scientists widely. Fractals are associated with the L-systems and defined by fragmented shapes that can split into parts, in which each part contains the same geometrical information as the whole. By investigation of the morphological development of the plants by **S.4**, Fractals and Pentigree L-systems were explored further in Processing environment (URL1 and URL 2). The algorithm was translated into a code in the Processing Visual Design software environment to generate 2D graphics, in which the colour of the 2D geometry can be altered by the code, as well as the iterations of the system.

A rose is also called as a rhodonea curve, defined by polar coordinates. The development of a rose flower contains several stages. In the growing process, the petals and sepals gradually open and form a blossom. The blooming process of the rose flowers, informed by a combination of the sun, wind, water, temperature, was investigated by **S.5**. A mathematical model based on a radial table was developed to identify the process, in which the diameter of the flower was associated with the sin and cos values. The ratio provided the exact stage of the blooming process. The mathematical formula was translated into a code in the Processing software environment to generate the 2D graphical pattern.

3.4 Motion-based Computation

The projects under this group investigate the dynamic behaviour in nature, as presented in the projects of **S.6** and **S.7.**

Climbing plants use trees or other objects as supports to grow vertically. They are consisting of a root, stem, leaves and tendrils. Tendrils obtain a coiling morphology, which informs their behaviour to hold the hosts and stabilize their position by coiling around. Tendrils can easily stretch or contract, which provide flexibility to adapt to the surfaces of the host objects. Their length gets shorten, by curling into spirals or twisting into helixes. **S.6** focused on the capabilities of the movements of the tendrils.

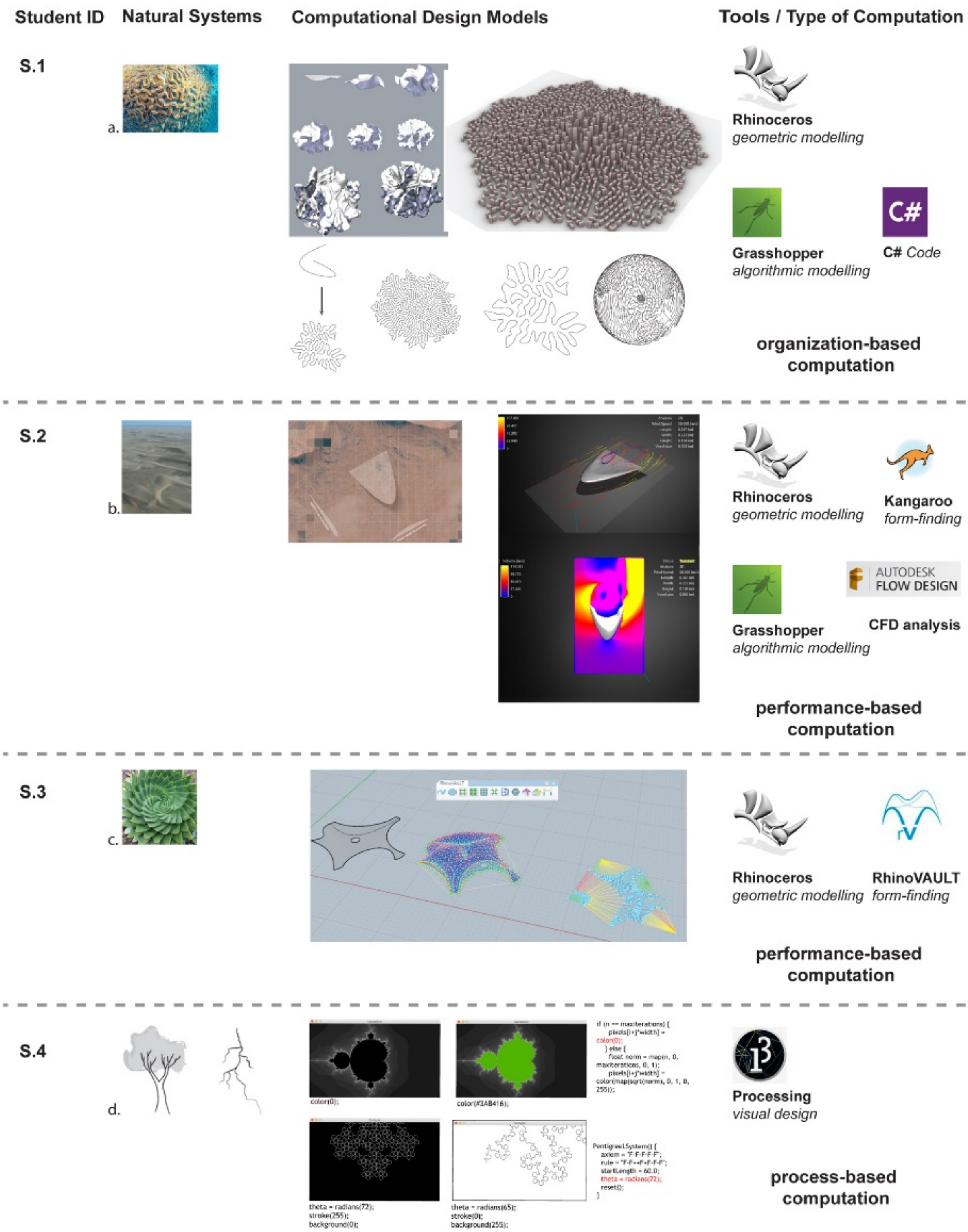
The adaptive properties of tendrils were examined in an architectural design project developed by Rhinoceros 3D geometric and GH algorithmic modelling tools. The movement and geometrical features of the tendrils were interpreted and abstracted in the generation of a responsive wall, in which each module could translate in one direction. Arachnids are evolved by enabling independent movements between their legs, chest and head and by accelerating their reflexes by environmental conditions. The skeletons and movements of these insects were investigated by **S.7**. The system was based on abstracting of the kinetic behaviour of the arachnids and translating it into a mechanical system for the operation of a responsive facade, in which window openings could be controlled. While the conveyors in the design proposal worked in a circular motion, the openings on the facade could be controlled in one axis. The circular motion was converted into direct motion by using inverse kinematics logic.

Table 1 shows the general evaluation of student projects. Additionally, **Figure 2 and 3** depict the outputs according to the computational design models, tools and type of computation. The output projects underlined that natural systems can be used as a source for learning, in terms of organization, performance, process and motion-based computations. Thus, the study extends beyond that nature is commonly used for formal resemblance in design projects.

Table 1: The evaluation of student projects.

Student ID	Natural systems	Abstraction of the natural system into the computational model	Tools	Output	Type of computation
S.1	Corals	Growth algorithm	Rhinoceros,GH, C# Code	Mesh surface	Organization-based computation
S.2	Dune formations	Physics-based spring systems/ CFD analysis	Rhinoceros,GH Kangaroo, Autodesk flow	Dwelling	Performance-based computation
S.3	Spiral aloe	Funicular form-finding	Rhinoceros, RhinoVAULT	Compressive shell structure	Performance-based computation
S.4	Growth of plants	Fractals/ L-systems algorithm	Processing	2D graphics	Process-based computation
S.5	Roses	Blooming algorithm	Processing	2D graphics	Process-based computation
S.6	Tendrils	Kinetic movement	Rhinoceros,GH: Translation of components	Adaptive wall	Motion-based computation
S.7	Arachnids: Insect skeletons	Kinetic movement	Rhinoceros, GH: Attractor curves	Mechanical system of an adaptive facade	Motion-based computation

Figure 2: The projects from S.1 to S.4, computational design models, tools and type of computation (Image credits from "a" to "d": URL 3-5; Shiffman, 2012, chapter 8).



Student ID
Natural Systems
Computational Design Models
Tools / Type of Computation
S.1
a.
Rhinoceros
geometric modelling
Grasshopper
algorithmic modelling
C#
C# Code
organization-based computation
S.2
b.
Rhinoceros
geometric modelling
Kangaroo
form-finding
AUTODESK FLOW DESIGN
Grasshopper
algorithmic modelling
CFD analysis
performance-based computation
S.3
c.
Rhinoceros
geometric modelling
RhinoVAULT
form-finding
performance-based computation
S.4
d.
color(0);
color(#3AB416);
theta = radians(72);
stroke(255);
background(0);
theta = radians(65);
stroke(0);
background(255);
Processing
visual design
process-based computation

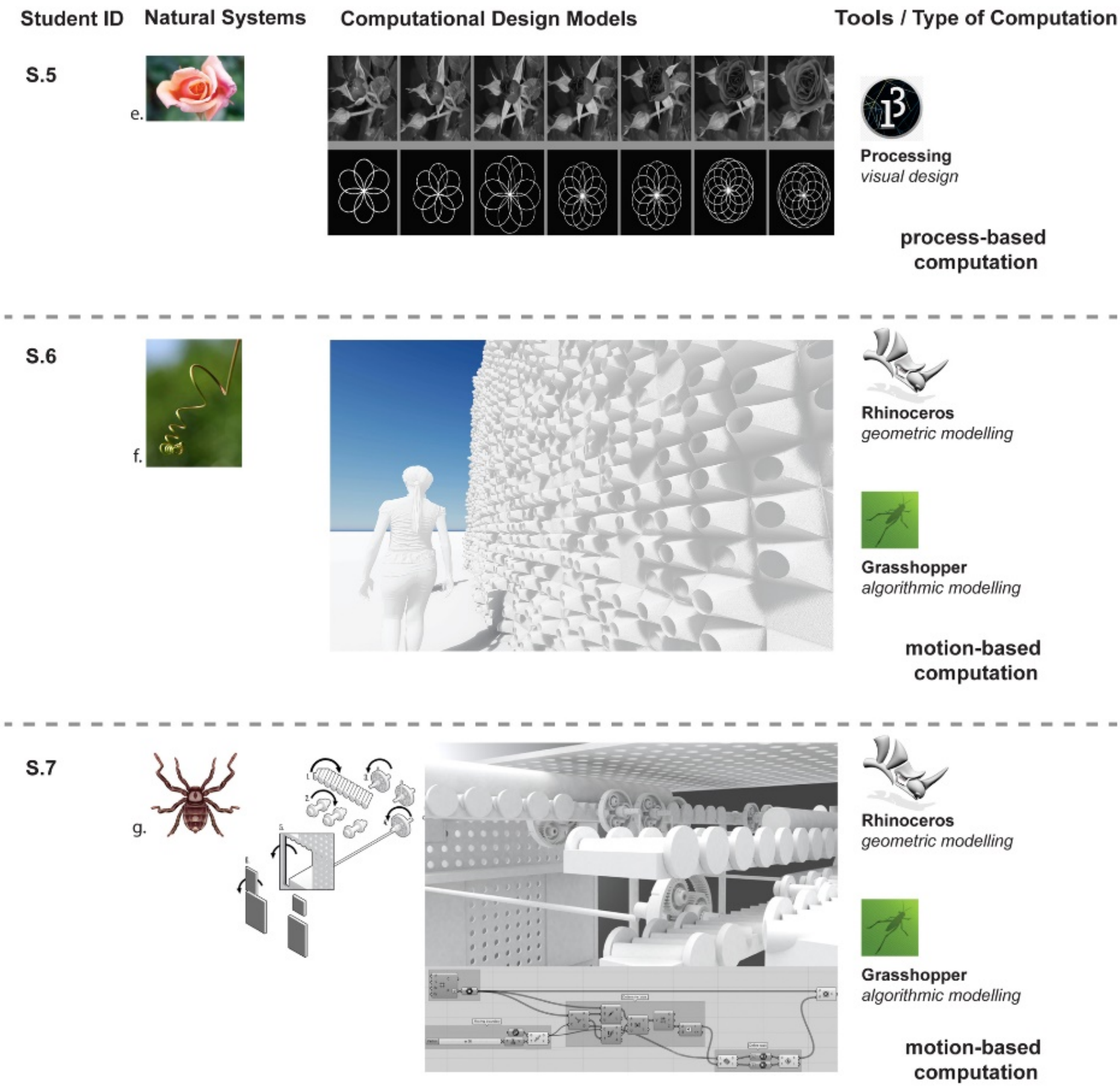


Figure 3: The projects from S.5 to S.7, computational design models, tools and type of computation (Image credits from "e" to "g": URL 6-8).

4. CONCLUSIONS

All matters in nature are evaluated within systems' unity. The intelligence of nature is considered as a source for problem-solving used by scientists, engineers, architects and designers. Although architects have been using nature as a reference since historical periods, there is a large amount of misunderstanding on, how to use natural principles in the design process efficient and not to imitate its aesthetics only. There is a necessity to increase awareness of architects, as well as architectural design students for this issue, by teaching the computational logic behind the natural systems.

This research shares the methodology and outputs of an elective course related to the biomimicry, which was implemented in undergraduate architectural education from the year 2018 to 2020. By evaluating the output projects, four groups were identified, differ in how natural principles are used and translated into the design process by using computation. This enabled to transcend the task beyond, how nature is used in the common architectural design process. The main difficulties encountered in the process for the students were to investigate natural systems with an analytical point of view and to generate an abstraction of the system by simplifying the system parameters because nature is highly complex. The other difficulty was that most of the students were not skilled, in terms of computational design. For future research, training on the computational tools and techniques might be given a priori to the course, to increase the capabilities of the students for establishing computational design systems efficiently and rapidly.

Acknowledgements
The studies were applied at Özyeğin University, Faculty of Architecture and Design, from the year 2018 to 2020. Special thanks to the students for their contribution to the course (Student names from S.1 to S.7 respectively: Benson Sanga, Tamer Kumaş, Özgün Şimşek, Burak Can Çatan, Ayşe Yasemin Şenalp, Doğa Çakmakçı, Mert Aydın).

References

Benyus, J. (1997). Biomimicry: Innovation Inspired by Nature, William Morrow and Company, New York.

Brugnaro, G., Baharlou, E., Vasey, L., and Menges, A. (2016). Robotic softness: An adaptive robotic fabrication process for woven structures. In: *36th the Association for Computer Aided Design in Architecture (ACADIA) Conference Proceedings*, *Ann Arbor* (pp. 154–163).

Castriotto , C., Giantini, G. and Celani, G. (2019). Biomimetic Reciprocal Frames A design investigation on bird’s nests and spatial structures, *eCAADe 37 / SIGraDi 23 Conference Proceedings,* Volume 1, 613-620.

Celani, G. and Vaz, C., E., V. (2012). CAD Scripting And Visual Programming Languages For Implementing Computational Design Concepts:A Comparison

From A Pedagogical Point Of View, *International Journal of Architectural Computing 1*(10), 121-137.

Çolakoğlu, B. and Yazar, T. (2007). Mimarlık Eğitiminde Algoritma: Stüdyo Uygulamaları, *Journal of the Faculty of Engineering and Architecture of Gazi University* 22(3), 379-385.

Kolarevich, B., and K. Klinger. (2008). Manufacturing Material Effects Rethinking Design and Making Architecture, 196–198. New York: Routledge.

Krieg, O. D., Mihaylov, B., Schwinn, T., Reichert, S., and Menges, A. (2012). Computational Design of Robotically Manufactured Plate Structures Based on Biomimetic Design Principles Derived from Clypeasteroida, Digital Physicality, 30th *Education and Research in Computer Aided Architectural Design in Europe (eCAADe) conference proceedings* (pp. 531–540). Prague.

Kvan, T., Mark,E., Oxman, R. and Martins, B. (2004). Ditching the Dinosaur: Redefining the Role of Digital Media in Education. *International Journal of Design Computing.*

Mandelbrot, B., B. (1977). The Fractal Geometry of Nature, W. H. Freeman and Company, New York.

Navarro-Mateu, D. and Cocho-Bermejo, A. (2020). Evo-Devo Strategies for Generative Architecture: Colour-Based Patterns in Polygon Meshes, Biomimetics, 5, 23; doi:10.3390/biomimetics5020023.

Oxman, N. (2009). Material-based Design Computation: Tiling Behavior. ReForm: Building a Better Tomorrow, *Proceedings of the 29th Annual Conference of the Association for Computer Aided Design in Architecture.* Chicago, pp. 122-129.

Oxman, R. (2008). Digital Architecture as a Challenge for Design Pedagogy: Theory, Knowledge, Models and Medium. *Design Studies*, 29, 99-120.

Oxman, R. (2017). Thinking difference: Theories and Models of Parametric Design Thinking. *Design Studies 52*(2017), 4–39.

Reitman, W. (1964). Heuristic Decision Procedures, Open Constraints, and the Structure of Ill-Defined Problems, in: M. Shelly and G. L. Bryan (eds), *Human Judgement and Optimality*, (pp. 282–315). New York: John Wiley and Sons

Schwinn, T, Krieg, O D, Menges, A, Mihaylov, B and Reichert, S (2012). Machinic Morphospaces: Biomimetic Design Strategies for the Computational Exploration of Robot Constraint Spaces for Wood Fabrication, *Proceedings of the 32nd Annual Conference of the ACADIA*.

Shiffman, D. (2012). *The Nature of Code.* Chapter 8. ISBN-13: 978-0985930806.

Suwa, M., Gero, J., and Purcell, T. (1999). Unexpected Discoveries and S-inventions of Design Requirements: A Key to Creative Designs, in: *Computational Models of Creative Design IV*.

Selçuk, S. A and Sorguç, A. G. (2009). Exploring Complex Forms in Nature Through Mathematical Modeling: A Case on Turritella Terebra. *Proceedings of eCAADe 27.*

Terzidis, K. (2006). Algorithmic architecture. Oxford: Elsevier.

Yazici, S. (2011). Computing through Holistic Systems Design Method: Material Formations Workshop, *Dearq Journal of Architecture*, Universidad de Los Andes, 90-101.

Yazici, S. (2015). A Course on Biomimetic Design Strategies, *33rd eCAADe 2015 Conference*, Vienna University of Technology, 16-18 September 2015, Vienna, Austria, 111-118.

URL 1 <https://processing.org/examples/mandelbrot.html>

URL 2 <https://processing.org/examples/pentigree.html>

URL 3 <https://www.nationalgeographic.org/encyclopedia/dune/#coastal-dunes>

URL 4 <https://oceanservice.noaa.gov/facts/brain-coral.html>

URL 5 <https://garden.org/plants/view/117109/Spiral-Aloe-Aloe-polyphylla/>

URL 6 <https://www.britannica.com/plant/rose-plant>

URL 7 <https://www.britannica.com/science/tendril>

URL 8 < https://www.britannica.com/animal/arachnid>