

Tools for automating the design of construction objects

Natalia Knyazeva¹ and Aleksey Orekhov^{2*}

¹Moscow State University of Civil Engineering (National Research University), 26, Yaroslavskoye shosse, 129337 Moscow, Russia

²Russian Biotechnological University, 11, Volokolamskoe highway, 125080 Moscow, Russia

Abstract. The digitalization of the construction industry is affecting the designer's profession - today it is not enough for him or her to be proficient in information modeling systems; the increasing complexity of projects requires him or her to have knowledge in developing plug-ins and scripts. Even if an organization has programmers on staff, an ordinary architect or a designer must understand how to issue a technical specification for writing an appropriate module. However, in most cases, designers have to optimize routine operations and automate individual tasks in-house using visual programming or algorithmic modeling tools.

1 Introduction

The range of software products designed to automate the work of participants in the lifecycle of capital construction projects is quite wide today. And the development of various extensions for them is quite common, because with their help specialists can solve highly specialized tasks. API (Application Programming Interface) is a set of methods, protocols and tools which are used by different programs to exchange data and perform various functions.

Algorithmic modeling (it is also called "computational design") is gaining popularity all over the world because of the possibility of performing a large number of different tasks with certain input parameters, conditions, performance criteria and solution search constraints [1]. With the help of algorithm the whole process of object creation is described: by means of a set of instructions the transition from one state to another is performed until the desired result is achieved [2].

The main possibilities of algorithmic design methods:

- Processing of large data sets to make an informed decision.
- Formation of adaptive models with parameters adjustable at any design stage.
- Automation of labor-intensive routine tasks.
- Detailed design of complex geometry.
- Multivariant design solutions generation.

The algorithms are based on mathematical models described in scripting languages. The first scripting languages appeared in the 1960s, and in the construction industry, the graphical

* Corresponding author: nknyazeva@mgsu.ru

programming language AutoLISP, designed for automating work in AutoCAD, is widely used.

In 2001, the text programming language Processing appeared, which is based on a combination of Java and C++, for designers, artists and architects to create images, animations and interfaces. With the help of Processing programs (sketches) the advertising of such brands as Nike, Hewlett-Packard, etc. was created. [3].

The next most significant milestone was the development in 2012 of DesignScript, an internal programming language for AutoCAD, a domain language for architectural computing. With DesignScript, experts developed a script, the execution of which led to the creation of a model.

However, the use of text and scripting programming languages required knowledge of the appropriate language syntax and semantics, which was often difficult for architects and engineers. For such specialists appeared the concept of node programming, allowing to create programs by means of work with ready blocks or nodes controlling graphic elements.

One of the most popular among such applications are the visual programming editors Grasshopper - a plug-in for Rhino and Dynamo - a plug-in for Revit. Work in their environment is carried out on the principle of "plug-and-play", the outputs of some nodes (nodes) are connected to the inputs of others, forming a graph. Dynamo workspace allows you to visualize the preliminary results of the graph.

Grasshopper is a visual programming tool developed for architects and designers as a plug-in for Rhinoceros 3D CAD [4]. Programs written in this language are a data stream consisting of a group of components and links between them. These components can be selected from a number of menus and dragged into the workspace. The components can represent functions, parameters or even geometry. The functionality of this application can also be extended with scripting components to write code using VB.NET, the C# or Python programming languages

The Grasshopper definition is displayed in Rhino by approximating the actual geometry. Nodes can be used to define parameters that store various data: numbers, lists of numbers, colors, geometry. The data in the parameters can be variable or constant. Some nodes have their own interface through which the user can specify data: numeric slider, graph.

Geometry is data defined with numbers, relations and formulas in coordinate space. Therefore, to work with it in an algorithmic modeling environment, it is necessary to understand how this geometry was created. It is possible to import data, for example, from a CSV or SAT file, as well as from typical geometric objects in another environment (Revit, Rhino, ArchiCad) in categories of certain types (e.g. walls and slabs), etc.

The data are required for processing actions and are fed to the input of the corresponding nodes, with which the creation of geometric elements, both simple 2d primitives and complex shapes with the help of logical and mathematical calculations, etc., takes place.

The library of ready-made nodes for different tasks is extensive, Grasshopper and Dynamo are open-source.

With the help of scripts designers today are solving many routine tasks: design drawings, automatic relabeling of grids of axes, rooms, structural elements, equipment, filling window and door openings, transfer of parameters between types and instances, etc.

Modern experience in the application of algorithmic design has a fairly wide range. For example, architects often resort to complex fractal architecture, creating new patterns of algorithms. For example, the design of the Great Egyptian Museum at Giza, opened in 2015, used Sierpinski's triangle.

One way to implement it in a computer is to use the Subdivide Triangle component in Grasshopper. The input is a closed triangular curve or surface, the number of divisions (dimensionality), and 4 logical variables responsible for ignoring the newly created internal triangle. The output from the component is two lists of triangular cells.

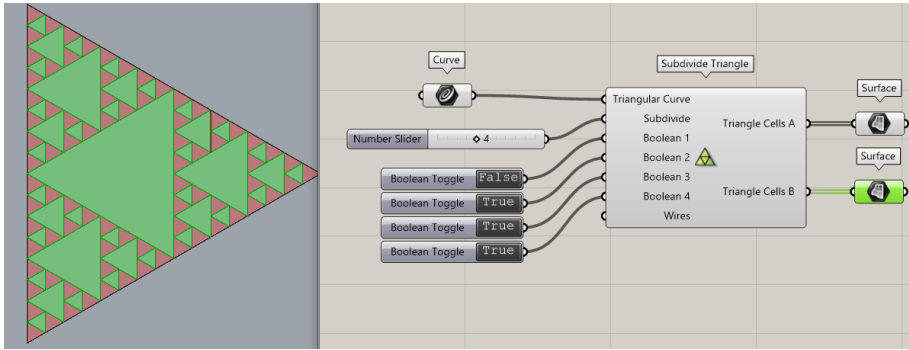


Fig. 1. The Subdivide Triangle component in Grasshopper.

Computer graphics uses Crow's Diagrams to randomly partition surfaces, a method of partitioning a plane into regions that contain all points that are closer to a given point than to all others. Voronoi diagrams have found use in architecture, including the creation of unusual interior elements as well as building facades. In Grasshopper, this method is implemented with the Voronoi component. A Surface is pre-defined and a group of Populate 2D points is placed on it with an adjustable number of Count and an initial value for generating a random number Seed.



Fig. 2. The Voronoi component in Grasshopper.

Based on the created geometry in the parametric model, it is easy to obtain numerical data: lengths, areas, volumes, and then compare them, make calculations and analyze. Among mathematical algorithms used for shaping are bifurcation, L-systems, attractors, tessellation, patterns, trigonometric curves, etc.

Practical examples of using Grasshopper for automating monotonous work are developing grid structure for Serpentine pavilion of London gallery, selection of shell pattern and calculation of materials for summer pavilion of Garage museum in Gorky park and others.

Leading CAD and BIM software developers are trying to incorporate computational modules into their systems, so, for example, ARCHICAD Live Connection, GH_SapfirComponent, Tekla Live Link, and other extensions have been created for integration with Grasshopper.

If we talk about the most interesting examples of practical implementation of the combination of information modeling and algorithmic design, it is worth noting the use of Tekla Structure and Grasshopper in the design of the following foreign projects:

- ARC is a recycling plant in Copenhagen with a ski slope on the roof. The client needed to model and place hollow floor slabs on the roof with a double bend. The gaps between the slabs had to be within tolerances.

- The tunnel of the Paris subway line. The process of placing profiles and other structural elements aligned to the rail track was automated, allowing the model developers to interact more effectively with the surveyors.

- Santiago Chile International Airport. Designers achieved a significant reduction in the time required to model the gable and roof systems.

2 Materials and Methods

Creating a parametric model using algorithmic tools in a digital environment has opened up the possibility of using automatic generation of solution options - generative design [5]. Different technologies are used: from evolutionary algorithms to agent-based systems. The field of evolutionary computation emerged at the junction of biology (genetics) and applied mathematics. The first mentions of the field of evolutionary solvers and genetic algorithms date back to the early 60's, and by the 80's evolutionary principles had become common, but remained a tool for programmers. With the development of visual programming (algorithmic design), architects and engineers began to use evolutionary algorithms to solve various problems [6].

In contrast to the standard design approach, when using generative design tools, the designer gets rid of manual searching for possible solutions. For automatic generation, he needs to set the input parameters, use a script to build an algorithm to construct a solution and evaluate it. Most often, the logic of generation of complex forms is comparable to the logic of development of natural objects and phenomena: the movement of a bird flock, leaf forms, etc. [5]

There are many tools for implementing the principles of generative design. The built-in plugin Galapagos in Grasshopper makes it possible to implement evolutionary calculations. In them, variables are called genes, and a certain value of each gene is called a genome. The solver's job is to remove the worst genomes and reproduce the best ones. Galapagos uses 5 interconnected parts to do this:

1. Fitness Function - this is the goal we are aiming for or evolutionary fitness.
2. Selection Mechanism - selection - natural selection - influences the direction of the gene pool.
3. Coupling Algorithm - coupling - finding a partner by genomic distance.
4. Coalescence Algorithm - coalescence - genes in evolutionary solvers behave like floating point numbers that can take all values between two numerical extremes, so inheritance mechanism is important.
5. Mutation Factory - Mutation is a mechanism for introducing biodiversity into a population.

The work of genetic algorithm can be demonstrated on a simple task: it is necessary to find such an arrangement of a point in relation to a set of points at which the distance from each point of the set to a given point will be minimal.

The coordinates of the point are given by the set of three sliders Gene Pool, the set of points placed in Rhino space is selected using Geometry Pipeline, the segments between the sought point and the points of the set Line are constructed, the length of each segment Length is calculated and these values are summed up Mass Addition. The sum of lengths of the segments is the target function for Galapagos, and the genome or variables on which the fitness function depends are the coordinates of the sought point.

After the solver is started, the list of solutions is enumerated, the list of individuals and the fitness of each individual is displayed (bottom right window), the graph shows the genome of the selected individual (3 points on it are the coordinates of our desired point), the points and red crosses show the number of selected and discarded individuals [7].

The main drawback of Galapagos is the inability to connect several different target functions.

An alternative plugin is Wallacei, an evolutionary multi-objective optimization engine that allows access to data at different stages of computation. From a single window, you can

start an evolutionary algorithm, analyze results, run selection methods, export phenotypes, and access an online forum.

The input to the Wallacei component is Genes parameters - genes, Fitness Objectives - calculation goals, Data - any type of data to be stored for each solution in the population, Phenotype - geometry, data to be exported through the solver.

One of the most popular applications of this component is solving complex situations in urban design, shaping buildings while maintaining environmental quality, taking into account building codes and keeping construction and operation costs as low as possible. Wallacei is integrated with Rhino and Revit [8].

To work with projects created in Revit, there is an application for generative design Project Refinery. It runs locally and exports data to Revit after work. Refinery covers work with two areas: Space Planning, which is generative space planning processes; and Massing toolkit, which is the optimization of massing processes.

Refinery can be used, among other things, to minimize the number of panels needed to design surfaces, optimize the placement of equipment on the construction site, and other tasks.

The Refinery toolkit is divided into 4 categories:

- Generate: for creating design parameters.
- Analyse: for analyzing results, comparing them to targets.
- Rank: ordering or ranking based on analysis results.
- Explore: to visualize the results.

In this study, the goal was to conceptually model the shading of the adjoining territory depending on the number of stories of apartment buildings in the design of the microdistrict. For the purpose of the task we will use the Rhino - Grasshopper environment.

Using the standard Rectangle, set in Rhino the contour of the area that we are going to analyze. Beforehand, for convenience, you can load the geometry of the city buildings, using data from Open Street Maps (OSM), and a little visualization of the volumes of existing buildings. OSM are maps, from which you can freely download any data that we need: for example, the contours of buildings and roads in a vector, information about the height of buildings and their functional purpose. Advantages of OSM maps - free data. The disadvantage - a small amount of data and their incompleteness.

Mark the contours of the projected apartment buildings on the plot. In the Grasshopper environment we will place Curve nodes, one of which will be assigned to the geometry of the site, and the other to the contours of the designed houses.

Next, set the initial volume of the projected buildings. Now any change in the location and size of the defined contours will automatically rebuild the display of the buildings. For the further work of the evolutionary algorithm we will be interested not in the size of the shadows themselves, but in the size of the unshaded space, the size of which should tend to the maximum, as well as the sum of the areas of all the floors of the residential buildings. We will use the Grasshopper Ladybug plugin in order to correctly display shadows. It should be noted that it should not be used for calculations at the design stage, since the principles used for its operation do not meet the requirements of the RF regulatory and technical base, but for the purposes of concept development and presentation of basic solutions to the customer Ladybug is an indispensable tool.

3 Results

We transform and visualize shadows and unshaded area and calculate its size.

Assign a bright color material to the resulting illuminated surface using Colour Swatch and Custom Preview for better visualization.

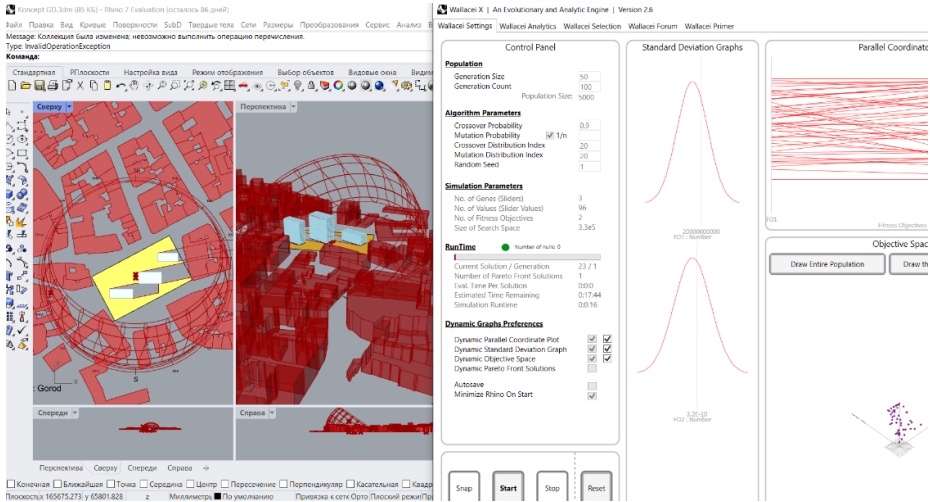


Fig. 3. Visualization of the script operation (Wallacei).

In this example, it is not possible to use the Galapagos plugin, since we have two target functions. With the Shift key we connect them to the Objectives water, and we connect the changeable parameters - the number of floors of each building - to the Genes input.

Based on the calculation results, you can use the Wallacei Analytics tab and other tabs of the window to select the most preferable solution options from the given analysis.

If the designer is faced with the task of multi-criteria choice, it is necessary to use Wallacei, but if you want to study only one target function, you can use the Opossum plugin. Or the above mentioned Galapagos.

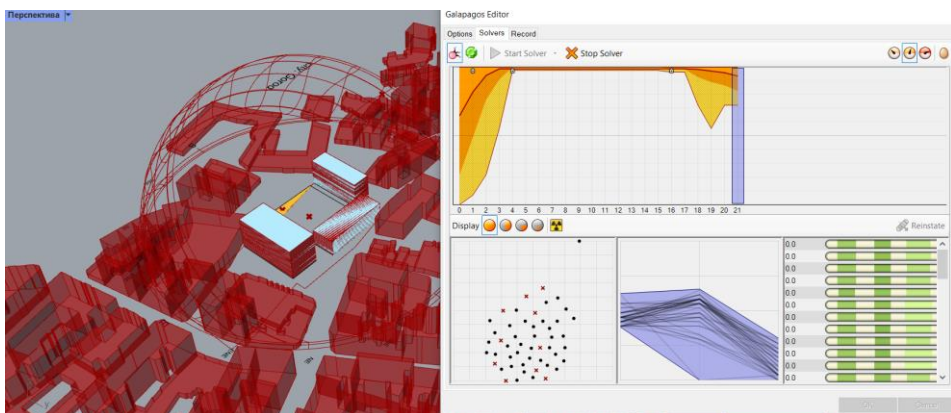


Fig. 4. Visualization of the script operation (Galapagos).

After setting it up, you should run it in the Solve tab. The calculation process here is also illustrated with graphs and tables.

4 Discussion

The article presents an approach to automating the design of routine tasks that require selecting the best of a variety of possible solutions, using the most common evolutionary solvers in the Grasshopper environment. While five years ago this approach was used mainly

by architects to create complex geometry, today this direction is of interest both to designers and employees of other departments of construction organizations.

In order to implement and regulate the principles of development and use of design automation tools in the construction industry organizations, it is necessary to provide appropriate sections in the BIM standard of these organizations. For example, in BIM&DESIGN STANDARD Development Systems (DS) the business processes related to tools that automate design activities are described in great detail. Its life cycle is considered:

1. Receiving an idea/order - collecting ideas and suggestions from the DS team;
2. Drawing up the ToR - analysis of received ideas, development of extended ToR, assessment of cost-effectiveness and feasibility, setting deadlines and responsibilities, etc;
3. Development - process of developing tools and testing internal code;
4. Testing - testing of the tool;
5. Pilot operation - using the tool in a pilot project;
6. Mass deployment - using the tool in all projects;
7. Product packaging - preparation of the tool for export;
8. External market entry - positioning of the tool, sales, support of external users.

It is also noted that getting ideas/orders is an ongoing process within the company, and the tools themselves are developed in batch once a quarter.

5 Conclusions

Plug-ins to implement evolutionary computation perform sampling based on given parameters without additional code knowledge.

The complicated term "computational process using evolutionary algorithms" is gradually moving from the everyday use of programmers to the familiar use of designers. Working with evolutionary algorithms helps to solve such problems as simultaneous search for multiple targets and generation of a wide range of solutions.

The use of evolutionary algorithms opens up a wide range of possibilities in a variety of fields, including the construction industry, and allows for more informed decisions at all stages of data handling.

References

1. A.N. Salingeros, Algorithmic sustainable design Twelve lectures on architecture Excerpts from the course (knigi-x.ru) URL: <http://net.knigi-x.ru/24tehlicheskie/862453-1-nikos-salingeros-algoritmicheskoe-proektirovanie-dlya-ustoychivogo-razvitiya-algorithmic.php>
2. D. Bulka, Grasshopper. Tutorial for Beginners. Fundamentals. Revision V3.2, URL: [https://rhino-help.com/help/GrasPrim/GrasshopperPrimerV3-2\(RUS\).html](https://rhino-help.com/help/GrasPrim/GrasshopperPrimerV3-2(RUS).html)
3. R. Aish, S. Hanna, *Comparative evaluation of parametric design systems for teaching design computation*, URL: <https://www.sciencedirect.com/science/article/pii/S0142694X17300327>
4. *Grasshopper program review*, URL: <https://junior3d.ru/article/grasshopper.html>
5. D. Corus, D. Dang, A.V. Eremeev, P.K. Lehre, *IEEE Transactions on Evolutionary Computation* (2017) URL: <http://www.doiserbia.nb.rs/img/doi/0354-0243/2017/0354-02431700003E.pdf>
6. D. Rutten, *Evolutionary Principles applied to Problem Solving*, URL: <https://www.grasshopper3d.com/profiles/blogs/evolutionary-principles>

7. A.V. Eremeev, *Genetic algorithms and optimization: tutorial* (Omsk state university, Lan' electronic library system, Omsk, 2020) <https://e.lanbook.com/book/136351> ISBN 978-5-7779-2439
8. *An Evolutionary Multi-Objective Optimization and Analytic Engine for Grasshopper 3D*, URL: <https://www.wallacei.com/>