Function Follows Performance in Evolutionary Computational Processing-Vertical Evolution

Anke Pasold\textsuperscript{1} and Isak Worre Foged\textsuperscript{2}
Function Follows Performance in Evolutionary Computational Processing-Vertical Evolution
Anke Pasold\textsuperscript{1} and Isak Worre Foged\textsuperscript{2}

Abstract
As the title ‘Function Follows Performance in Evolutionary Computational Processing’ suggests, this paper explores the potentials of employing multiple design and evaluation criteria within one processing model in order to account for a number of performative parameters desired within varied architectural projects. At the core lies the formulation of a methodology that is based upon the idea of human and computational selection in accordance with pre-defined performance criteria that can be adapted to different requirements by the mere change of parameter input in order to reach location specific design solutions.
I. Introduction

Recent developments in the domain of digital architecture show a wide and growing interest in algorithmically described processes as a basis for emerging morphologies that attempt to replicate recursive processes found in nature [1][2]. The aim of such approaches is often directed towards constructing complex, adaptive geometries that explore architectural language and inform various performative aspects upon active elements [3]. Most commonly, this is achieved via a method of creating code that informs a repetition developing shapes and their organisations into complex geometrical assemblies [4].

The code itself thereby utilises recursive principles similar to those observed in nature, [5] whilst, in contradiction to natural processes evolving in ‘free’ space. This means that most such approaches seem to neglect to include the very driving parameter for a morphology’s genesis-its specific environment and the resulting adaptive correspondence of the system to the same. Such a narrowed view results in displaying a lack of contextual relation. A point that is already strongly expressed in the work of John Frazer’s ‘An Evolutionary Architecture’, where he is emphasising the importance of external, environmental influences on any evolutionary process [6] in systems being part of their corresponding environment.

The paper at hand focuses on this perspective in order to discuss the validity of extending algorithmic processes in architecture beyond the currently employed level. This will allow them to instead act as adaptive parameters in accordance with the specific environment they are placed within similar to those found in nature.

Approaching algorithmic-based systems in such a way would implicate that it is only code and environment combined that can facilitate towards environmentally influenced morphological compositions. Such approaches could eventually further extend the idea and potential of performative environmental adaptation by introducing an internal spatial organisation according to external factors and the potential of functional use according to performance.

This paper therefore argues for two main points. Firstly, it is to approach emergence in architecture through a combination of evolutionary algorithms and a specific predefined environment within which structures will evolve and, secondly, it will successively include geometrical rationalisation and climatic environmental simulation into the designing process to achieve functional spatial use according to performance.

As a result, this will lead towards an implementation of evolutionary explorative form finding in performative architecture in an attempt to integrate a number of contextual factors into algorithmic architectural processes and their corresponding solutions.

The paper therefore presents a progression methodology that supports the above-described approach and subsequently seeks proof in a ‘verifying’ design proposal on an exemplified building structure that was generated and submitted within the context of the eVolo competition 2009.
It will start by formulating the methodological approach in the chapter describing the ‘Morphogenetic Platform’ to thereupon further elaborate in the ‘Processing’ section in order to show the design progression, clarify the digital environment and the concepts employed for the explorations at hand.

2. Morphogenetic Platform

The desire to let both, architectural process and designed product, be influenced by nature is predominant in contemporary architectural explorations. The code of nature, the DNA, is translated into scripted processes letting a morphology evolve on the basis of the criteria set by the designer.

Code in nature constructs the platform for morphogenetic processes in order to support the principles of emergence through bottom-up descriptions [7]. Attempts to imitate and illustrate such growth processes were undertaken a.o. by Aristid Lindenmayer, who, in 1968, invented the Lindenmayer System (L-System), a branching system able to illustrate the formations of plants [8]. This method is still being used as the predominant means of representation and exemplification of such growth processes. The L-System thereby describes the rules for connections and the evolutionary steps while a mapping system is required to advance the representation of L-System evolutions. Simple rules, called Turtle rules, guide the evolutionary process via commands telling each line that connects an evolutionary step to go in a particular direction in 2D. Such combinations give an endless foray of 2D patterns including such famous fractals as the Koch-curve and the Sierpinski Triangle that a.o. have resulted in an experimental method for computer graphical artists [4].

The current idea of natural inspiration within architecture, however, is predominantly of a visual character. This is due to the absence of the environment influencing the evolution of above-mentioned process. The very idea of code or Evolutionary Algorithms (EA), on the other hand, is to mutate and adapt according to such a specific environment, which inversely means that a code without an environment cannot take use of adaptive elements. EA are thus in need of an environment to adapt to, just as plants grow under gravity and other natural factors [9].

Programming EAs and their specific environments lies well outside the scope of this investigation. The aim is much rather to investigate the use of such systems for architectural purposes through the development of a design methodology. The research is therefore utilising Genr8, designed by Martin Hemberg et al., as a tool that can provide the basis for the EA, the environment and HEMLS (Hemberg Extended Map L-System), allowing the Map L-System to grow in 3D and subsequently construct surfaces as phenotypic output to the genotypic EA [4]. The user is thereby given the chance to self-define a series of growth parameters relevant to his own design task, such as branching angles and lengths, number of populations and generations, which, just as observed in nature, will cause a number of
mutations and adaptations throughout the evolution process to produce an endless number of different results. It is therefore the designer that is composing the evolutionary grammar as a combination of several growth parameters to form the evolution. These growth parameters consequently react within the given environment, to which a new varied output is created for each ‘run’ as a result of the EA. The software Genr8 runs within the 3D environment of MAYA, which also hosts the environment’s boundaries in form of a ‘space container’, which in itself could be understood as and adjusted to an urban environment or the spatial boundaries within an architectural brief.

A reading and understanding of the generated surfaces, best assisted with a thorough research protocol documenting the utilised parameters and corresponding effects, has great importance to construct alterations within the grammar towards the desired and thereby most optimised design solutions that will fit the predefined criteria. This is further helped by the implemented IIR ( Interruption/ Intervention/ Repetition ) system allowing the designer to take active part in form of a human selection, while the surfaces are generated on the basis of a computational selection [4]. This is especially useful as the generic process records each generation of the running algorithm, defined by the grammar, visualising the evolutionary steps. The designer can thereby identify the effects of the composed grammar in a more detailed fashion and accordingly alter the fitness criteria, such as geometrical composition, surface densities etc. Design criteria of structural integrity and contextual sensitivity is consequently continuously ‘tuned’ during the computational process, contrasting to a more traditional way of letting the algorithm ‘run’ and wait for the result. The explorative environment is based upon parameters of natural growth and geometrical fitness expressed in surface connectivity and surface direction. Their definition within the grammar of GenR8 can therefore, according to Hemberg and O’Reilly [10], substitute the initial structural analysis adding an empirical performative aspect to the evolutionary search. Grammar definition within this work thereby attempts to explore architectural language, contextual sensitivity and structural integrity in order to construct a multi-performative morphogenetic platform. This will be further described in the ‘Processing’ section. Emphasis is laid on the idea of performance as a generator for further architectural detailing leading to a design methodology where ‘Function Follows Performance in Evolutionary Computational Processing’.

3. Processing

This principle has been set up as a design methodology in consecutive preconceived steps leading from an initial data input in Genr8 to several outputs that are re-fed into the system cycle to progress into a functioning final solution for a specific geographic location through a number of steps.
In order to reach a well-performing design output, the result of each step is undergoing an evaluation process either through the user (Human Selection) or the computer (Computed selection) in order to select the fittest outcome for the relevant functional requirements. This selection process itself thereby resembles nature’s concept of selection, according to which she takes her decisions on which species or individuals are fittest for survival in a particular ecology and will consequently be performing best within it.

The nine consecutive steps, explained in more detail below, thereby run through a cycle of different tools: starting from the generation of surfaces within Genr8, that are being merged and rationalised by the user to create a construction logic that in its resulting form can be analysed within a variety of simulation programmes for a multitude of its performance potential. In the example further explored within this paper, Ecotect was used for the analysis of daylight factors across the different parts of the constructed object.

**Step 1: Designing the Grammar (Human Selection)**

At the very start lies the definition of a project specific evolutionary grammar that has multi-performative capacities. For this purpose, the user is provided with a comprehensive choice of items, that each will influence the eventual result in specific ways and thus give a basis for broad experimentation as to how certain changes in the initial parameter definition alter the outcome.

This grammar will then serve as a mapping device for the genome that comes to contain all the information for algorithmically growing geometries on the basis of the processed input data [4].

**Step 2: Generate (Computed Selection)**

This user-defined grammar is thereupon interpreted within the system and consequently serves to create the digital phenotype in form of constructed surfaces according to the user input data. Repeated grammar cycles within the same environment construct a catalogue of possible formal outcomes.

The relevant results are thereby explicitly described in growth protocols accounting for such parameters as branching angles, populations and generations.

This serves as a detailed research base into the generated surfaces that are thereupon studied in terms of their geometrical logic and structural integrity as well as their spatial arrangement. Special attention is given to surface densifications, intensities, directions and scale to organise performative emergent space in three dimensions.

**Step 3: Describing the Environment (Human Selection)**

The generative process can be induced within open space or inside a restraining boundary where gravity and other forces are applied. In order to
design a location specific geometry, the environment would now be introduced to the system as a next step within the proposed methodology in order to function as a restricting borderline beyond which the growth is not to expand. This serves to respect the existing urban setup instead of creating interference in order to generate a geometric shape that constrains itself within the plot area in specific or a design brief in general.

**Step 4: Generate (Computed Selection)**

This pre-defined grammar is thereupon repeatedly run within the boundary box for a catalogue of location specific results.

In the exemplified project of *Vertical Evolution*, the constraining box marked a vertically extended geometry resulting in a growth that was strongly vertically directed therefore generating the apparent skyscraper structure.

**Step 5: Evaluating Results (Human Selection by Expression and Structure)**

The results are studied for their potentials with regards to structure and architectural expression and the fittest result is chosen for continuation within the process. Should there be no suitable outcome will the cycle start once again from Step 1 to be repeated until a positive selection can be made: one that will fit the required performance criteria within the specific geographic location and according to the predefined selection requirements.
Step 6: Rationalise (Human Selection-Bringing the Design in a Construction Logic)

In the exemplified project, \textit{Vertical Evolution}, the cycle was run several times for a simultaneous composition of structure, space and expression that can be seen in Step 4.

Out of the generated range, three suitable results were chosen and merged to create spatial definitions, enclosures, intersections and divisions. Those were in turn to provide the basis for a rationalisation process, within which the constructed singular surfaces were translated into constructable triangulated pieces whilst strictly following the structural build-up in order to keep Genr8’s internal structural integrity. As a result structural planes, that enter a synergy of inside and outside, where created.

Step 7: Vertical Internal Division from Spatial Evaluation (Human Selection)

The resulting high-rise structure could now be interpolated with the existing urban fabric in order to introduce local dimensions for internal levelling divisions in form of horizontal floor planes at sensible intervals and with partial jumps.
Step 8: Environmental Evaluation of Spatial Division

Each resulting individual layer of the building could accordingly be subjected to environmental analysis in order to describe the internal performance properties in relation to their received energy (solar radiation) depicting the specific light and temperature levels at all positions in the building.

Step 9: Internal Spatial Programming from Environmental Spatial Performance (Computed Selection)

This information could thereupon be used to assign functions within the building in accordance to their requirements resulting in a plan layout that is programmed into zones of thermal comfort levels, suppose to traditional ways of organisation. This approach is leading to the suggestion of an open content use with multi-programming possibilities for the internal spatial arrangements. Determining factors would be such parameters as frequentation and daylight requirements, rather than pre-arranged, inflexible functionality.

This is taking the building’s content to a new level of openness in terms of use while, through the analysis themselves, connecting it more strongly to its geographic surroundings and thereby to its corresponding environment.
4. Implications

Combining design intents with emergent form evolution not only requires the utilisation of evolutionary algorithms within an environment but also calls for a methodology that supports the interaction between designer and computational processes for a multi-performative design outcome.

This type of methodology thereby incorporates the human capacity to continuously evaluate and translate both empirical data and phenomenological affects to advance the evolutionary computational process and extend it by the relation to its environment. The achieved architectural result is thus in direct relation to the human understanding of and resulting selection from the number of generated geometries.

Architectures ‘born’ from such evolutionary processes have an evident connection to their intended location through adaptation to the environment within the described process. Thermal conditions of internal environments, however, are merely a byproduct of the complete generated morphology as growth processes based upon L-Systems naturally neglect to include important aspects such as temperature gradients and other thermal balances.

The presented methodology and the therewith connected exemplified project Vertical Evolution, attempt to extend Genr8’s processes by ‘mapping’ the environmental properties of the site specific generated model. As there are no indications of the internal environmental performance during the generic process, the internal spatial performance is being analysed through the introduction of Ecotect into the design process. Analysis data is thereupon employed as a guide to internal programming with regards to space specific requirements. Much the same way as the specifically created environment guide the EA in generating the surfaces.

5. Advancing the progression

As an approach, methodology and architecture, the work presented here is focusing on emergence with the aim of engaging in the complexity of traditional architectural questions of contextual relations, structural
integrity, spatial definition, internal environments and functional programming within one single explorative generative model. It has been trialed within the frame of a competition and been found valid as a methodology to be applied to an arbitrary project based on an external brief.

Within a next step, this methodology could be further extended by implementing characteristics of the host environment. Internal environmental factors would then induce behaviours to the EA on the basis of the same. This would most likely imply the creation of a link between environmental simulation software and the EA as a feedback system, allowing a different level of environmental adaptation through direct data input at relevant points in the process. The implementation of more performance factors, such as daylight spreading will result in a more explorative character of architectural design. Care should, however, be taken to not limit or decelerated the otherwise dynamic process between fastly computed models and immediate decision taking by the designer in order to maintain an experimental trial-error environment [11]. This will uphold the evaluating position of the designer at each singular step of iteration with the attempt of finding the fittest, most suited solution for a specific design task and thereby a desired location specific performance. For a successful, empirical architectural process, the designer will be required to balance between in-depth studies of the evolutionary behaviour and the fast iterations between computed and human selection to make valid and informed decisions at each singular step of the process instead of judging by mere formal expressions of the generated geometries. Evaluating the results in the pre-described manner will allow for the methodology to construct a geometry circumscribing functions fit for the desired performance and thereby construct an architectural result on the principle of Function following architectural Performance and thereby reach beyond purely formal solutions.

References


Anke Pasold1 and Isak Worre Foged2

Studio AREA
Copenhagen
Denmark
Institute of Architecture
Design & Media Technology
Aalborg University
Denmark
1 anke@studio-area.net
2 isak@studio-area.net