political, cultural, and social notions.

As an "ism," Performalism may allude to autonomous and reciprocal procedures—procedure for its own sake (as in formalism—form for the sake of form). The works presented in the exhibition apply performative aspects in architecture for the sake of performance. Nevertheless, since the idea of performance initially attempts to incorporate multiple layers of reality, the outcome exceeds the limitation of autonomous operation and provides a wide range and inclusive possibilities for formal existence in architecture.

As a manifesto, the exhibition calls for performance in architecture. Living in a time in which the digital tool allows to design and integrate architectural properties and aspects in high resolutions, we can reach a highly personalized yet shared architecture. Performance as a conceptual and practical mode of operation provides us with the means to create an architecture that is in-between the individual and the collective, in-between utilitarian and symbolic functions, the intuitive and the rational, the sensual and the analytical. In this architecture, objects and subjects act as performers, creating environments for future growth.

Yasha Jacob Grobman is an architect and a partner in Grobman Architects and Axelrod Grobman Architects. He is a PhD candidate at the Technion – Israel Institute of Technology. His doctoral dissertation concentrates on performance-based form generation and optimization in digital architecture. He is a cofounder and director of T_CODE (Technion's Computer Oriented DEsign Laboratory), and serves as a director in OCEAN, a trans-disciplinary international research network.

Eran Neuman is an architect and a Senior Lecturer at the David Azrieli School of Architecture, Tel Aviv University. He holds a PhD from the University of California, Los Angeles (UCLA). He is a cofounder and director of T_CODE (Technion's Computer Oriented DEsign Laboratory), as well as principal and a cofounder of Open Source Architecture.

THE VARIOUS DIMENSIONS OF THE CONCEPT OF "PERFORMANCE" IN ARCHITECTURE

Yasha Grobman

>> The transition to computerized object-based design[1], and the improvement in the processing ability of computers, have led, in the past decade, to a significant increase in the quantity of information embodied in the form and the process of architectural design. Information-rich architecture based on "smart forms"[2] exists in a new dimension that is built on information hierarchies, from the level of the single parameter through to algorithms and programs that define relationships among numerous parameters. The use of parameters or algorithms as bases for production of forms, and in the architectural design process, as well as the increasing complexity of programs of architectural creation and the growing use of computers in architectural design, call for a re-examination of the system of laws in which architectural creation is conducted.[3] This time, however, in contrast to precedents such as the design methods of Christopher Alexander[4] and others who attempted to arrive at a comprehensive, logocentric, theory, attempts are being made to define these laws in terms of specific, local, understandings. This kind of understanding continues the parametric logic of the computer in a way that makes possible a deconstructive use—i.e., disassembly and creation of new programmatic and formal complexities.

In this way a new kind of architectural database is gradually developing, which—in contrast to classical databases, such as those that focus on typologies—contains tools and methods of form creation that are based on a computer code. This database exists and develops in the free world of the open code on the Internet, and, as in other disciplines (the computer sciences, for example), makes possible free adaptation and downloading of architectural codes for local, particular, needs.

This article proposes a definition of the concept of performance in architecture based on the logic of parameters, while making a first examination of the possibilities of using the various dimensions of performance in computer-based architecture, and a first examination of the meanings and implications of these possibilities.

[1] As distinct from design based on lines defined by two points in space.
[2] A "smart form" incorporates quantitative information connected to the form's performance as well as information on the form's geometry. See Guedi Capeluto, "Energy Performance of the Self-Shading Building Envelope," *Energy and Buildings* 35 (2003), pp. 327-336.
[3] In the early 1960s the computer was perceived as an intelligent problem-solving machine that in the not too distant future would equal and even surpass human capability. That period saw the development of a large number of theories and models for the automatization of the design process and the optimization of its products. See Alfredo Andia, *Managing Technological Changes in Architectural Practice: The Role of Computers in the Culture of Design*, Ph.D. dissertation, University of California, Berkeley, 1997.
[4] Christopher Alexander, *Notes on the Synthesis of Form* (Cambridge, Mass.: Harvard University Press, 1964), p. 9.

Static Information and Dynamic Information: Form, Function and Performance

In order to examine the way computer code is used in architecture we need to define the kinds of information or the kinds of parameters on which it is based. A possible basic division relates to two kinds of components—static and dynamic. Static components describe a fixed, inert, situation that may be connected to the architectural object or form. Dynamic components focus on an action—on a changing of the form or on the relation between the form and the space it is in. The latest developments in the use of computers in architecture have to do mainly with parameters of the latter kind.

Another possible distinction divides information into descriptive and performative. The increase in the quantity of information embeded in the architectural form began in the descriptive dimension, when software borrowed from other disciplines made possible the presentation and alteration of complex forms. Today, however, its major influence is expressed in the performative dimension, which relates to advanced possibilities of form development that are connected with simulation, optimization and generation of an architectural form through examination and alteration of relationships in the realm of performance.

As in code development in the computer sciences, alteration of parameters has a meaning mainly when it is channeled to achieve a particular goal. A computer code without a goal is like a meaningless collection of words or lines. According to the modernist discourse, which preceded the computer era, the "goal" of a form means a search for its function or actualization. The emergence of the computer was one of the major reasons for the diversion of the architectural discourse to forms of thinking that go beyond form or function, in a way that does not discard the discussion of these, but attempts to define the connection between them. It may be argued that the connection between form and function is meant to define the way in which the form sustains the function, and that a connection of this kind may be actualized by means of an examination of the performances, so that by means of the performances required by the function it becomes possible to arrive at the form. Indeed, the prevalent and narrow definition of the concept of performance relates to the quantitative-binary character of the computer code, and focuses on measurable, empirical, performances. A broader definition of the concept, however, contains three dimensions of performances: an empirical dimension, which focuses on directly measurable performances that usually relate to physical data such as strength, temperature, the quantity of light etc; a cognitive dimension, which focuses on the way human cognition can be translated into space, and, conversely, the way space can be translated into human cognition; and a perceptual dimension, which focuses on the way human perception can be translated into space, and, conversely, the way space can be translated into human perception.

The empirical dimension is immediately translatable into computer language, but translation into computer language of the cognitive and the perceptual dimensions, which can be measured principally by a statistical method (which, for example, examines numerically the preferences or the aesthetic evaluations of a group of people in a particular space), still constitutes a complex problem for which there are no immediate solutions.

Form-Based Design and Performance-Based Design

Performance-based architectural design relates to the three dimensions of the concept. The personal interpretation of a program may prefer one particular dimension of performance over the others in different parts of the design process. The final product in the process of creating a form depends not only on the dimensions chosen and on the kind of parameters of which use was made, but also on the order of their appearance in the design process. It is possible, of course, to concentrate and to use only one dimension of performance throughout the entire design process. But a project that has been developed in a one-dimensional manner is based essentially on inadequate information and will not sustain its function satisfactorily.

It is probably impossible to prove directly that computer-based design, which makes use of the various dimensions of performance, leads to a better outcome than use of a different design method that may or may not entail use of computer. Proof of a claim of this kind would require a hierarchical definition of parameters and a comparison of the various outcomes, and such a definition would in its essence be subjective. At the same time, it can be claimed that the more aware a designer is of the way the form he has created functions in terms of the three dimensions of the concept of performance, the better he can control the object being designed and adapt it to his wishes and to the way he interprets the program.

Form-based design, which develops a form while ignoring or not relating to the three dimensions of performance, is possible in certain parts, mainly at the beginning of the process of creating the architectural form. A method of this kind, such as a "shape grammar,"[5] may lead to a greater complexity of form, which will have to be given meaning during later stages of the design process while examining the way that the form fulfills the requirements of the various dimensions of performance.

The Use of the Various Dimensions of Performance for the Simulation, Optimization, and Production of Forms

One of the foreseeable effects of the transition to computerized design and production is a rise of the architect's status in the set of forces operating in the building discipline. If before this transition the architect was responsible for the design and production of plans that it was the builder/contractor's job to realize, in object-based design and production the architect in fact produces the file from which the real object is produced, without any need for mediators.

One of the ramifications of the enhancement of the architect's status consequent on the transition to object-based design and production and the increasing connectivity among computer programs is the proliferation of possibilities of using tools and processes such as simulation, optimization, and production of forms, which until now were the exclusive domain of researchers, advisers, and engineers. Although at the start architects used simulation primarily for visualization, with the increase of programmatic complexity and simultaneously of the performative demands from the architectural form, the use of simulation of performances has expanded. The incorporation of the simulation processes as part of the architectural design process performed by the architect does not do away with the need for professional advisers, but it does lead to a professionalizing and a fine tuning of the examined parameter.

[5] A "shape grammar" is a design method that was first presented by Georgy Stiny and James Gips in 1971. The method is based on a multiplication of changes of forms by means of rules, in order to create complex compositions. See Georgy Stiny and James Gips, "Shape Grammars and the Generative Specification of Painting and Sculpture," *Information Processing* 71, ed. C. V. Freiman, North-Holland Amsterdam, pp. 1460-1465; reprinted in *The Best Computer Papers of 1971*, ed. O. R. Petrocelli, Auerbach, Philadelphia,1972, pp. 125-135; see also www.shapegrammar.org/biblio.html.

[6] Peter Eisenman, "Visions Unfolding: Architecture in the Age of Electronic Media," *Domus* 734 (1992), pp. 17-21.

The expanding use of computer codes for optimization and production of architectural forms entails much potential, but also a danger. The products of the processes of optimization and production cannot be predicted in advance, demonstrating the validity of Peter Eisenman's vision about the need for loss of the human eye's control in the design process.[6] At the same time, since it is impossible to define the totality of the architectural problem,[7] it is also impossible to solve it empirically as is done in modern science. Hence it is hard to speak of optimization of form in the scientific/empirical sense. In an optimization process that entails more than one parameter belonging to the empirical dimension of performance, there needs to be a subjective definition of preferences in order to arrive at the "optimum." And even then the optimum will always be specific, since, as already noted, the order in which the processes are activated, and the kinds of parameters chosen, change the final product.

Hence, because of the subjective definition and the complexity of the architectural problem that entails reference to many parameters, the idea of optimization in architecture takes on a different meaning. The problem is even more difficult in the cognitive and the perceptual dimensions of the concept of performance, because the initial definition of the parameters is done subjectively by a statistical translation of human desires and impressions.

Heretofore, processes of producing architectural forms have focused primarily on production of forms that relate to the building's envelope. Likewise, a considerable portion of form-producing processes focuses on function by relating to a single component, such as wind, or sun, or stability of the construction. It appears that the developing of methods that incorporate a number of parameters to create a form on the basis of the concept of performance is the next stage in the development of form production in architecture.[8] At the same time, production of complex typological forms which, beyond the building's envelope, also include a division into secondary interior spaces, probably remains at the present stage a challenge for future generations. Today, the architect at a certain stage of the production process has to decide to "freeze" the formal configuration and switch to a process of analogical design that relates to the additional dimensions of performance which at present cannot be incorporated into the production process.

The Architect of the Future and the Moral Dimension of Performance

The transition to computer-based design hints that the architect of the future will require a greater mastery of mathematics and of computer languages in a way that will enable him to at least adapt existing tools to his own needs, if not to improve skills of writing new code. These skills will not require the qualifications of a programmer or a mathematician, but will need an understanding and an ability to use computer-based parametric processes that already today are being used on interdisciplinary levels.

At the same time, a reliance on parameters in architectural design that is based on use of a code should raise questions regarding the moral dimension of this kind of design. A danger exists of a transition to a pre-set, deterministic design that is based on parameters while neglecting the human aspect which, as already mentioned, is still difficult to express in parameters. Indeed, the use of algorithms can lead to the creation of new ideas, forms, and perceptions.[9] But the attempt to imitate the way we think in terms of parameters is by its very nature limiting, and may lead to the preferring of certain forms of thinking and to the neglecting of others that are not easily translatable into computer language.

In addition, parametric thinking that relies on a limited number of computer languages tends by its nature to the universal. It is based on uniform languages and patterns that need to communicate with one another and to serve the global consumer. Although as a language it constitutes an opening for local-specific possibilities of expression, the paucity of languages and the aspiration for uniformity may lead to the ignoring of subjective, local-specific needs, and of forms of thought that are not commensurate with the logic on which the language is built. For this reason it is important to continue developing the computer based processes while understanding that these are being added to the developing store of tools and methods of architectural design in a way that will enable the architect to choose and to adapt the chosen tool/method to the particular problem, while remaining aware of the advantages and the disadvantages of the unique situation.

The attempt to translate the connection between the form and the function through the various dimensions of performance constitutes a great challenge for architecture. Response to the challenge will cause a further heightening of the architect's spatial awareness, by increasing the information about the architectural form and decreasing the entropy of the architectural problem. What is important in this response to the challenge is the way, not the goal. The way, in this case, is by its nature not linear, and it must allow for the concurrent existence of many directions of development.

----

[7] Bruce Archer, "Whatever Became of Design Methodology?," *Design Studies* 1 (1979), pp.17-19; reprinted in *Developments in Design Methodology*, ed. Nigel Cross, John Wiley & Sons. Chichester, 1984, pp. 347-349; Gabriela Goldschmidt, "Capturing Indeterminism: Representation in the Design Problem Space," *Design Studies* 18 (1997), pp. 441-445; Peter Rowe, *Design Thinking* (Cambridge, Mass. and London: MIT Press, 1987).
[8] See Yasha Jacob Grobman, *Building the Digital World—Architectural Design Methods Based on the Use of Digital Tools—Performance Based Form, Generation and Optimization*, Ph.D. dissertation, 2008, Technion (IIT), Haifa.
[9] Kostas Terzidis, *Algorithmic Architecture* (Oxford: Architectural Press, 2006), p. 20.