METU JFA 2012/1
(29:1) *207-224*

DOI: 10.4305/METU.JFA.2012.1.12

CREATIVE DESIGN EXPLORATION BY PARAMETRIC GENERATIVE SYSTEMS IN ARCHITECTURE

İpek GÜRSEL DİNO

Received: 03.01.2012, **Final Text:** 02.04.2012

Keywords: parametric design; generative systems; design exploration; performance based design.

This paper discusses parametric design systems as a generative tool in architectural design. Parametric tools are algorithmically based, and therefore offer increased computational control over design geometry during design activity. Their adaptability and responsiveness to changing design criteria and requirements make parametric models especially useful for design exploration in complex and dynamic design settings. In performance based design, parametric control of form is particularly valuable, such that they allow the integration of performance analysis into design synthesis. However, parametric systems are often incorrectly mistaken as an emerging architectural style rather than a computational method, ascribing it a false skin-deep character that overshadows its true merits. Moreover, parametric models come with a price, posing limitations regarding representational flexibility and design complexity, which hinder effective design exploration. A critical awareness on both the potentials and limitations of parametric systems is therefore critical in their effective use during design.

INTRODUCTION

Computational systems have emerged as a fundamental keystone in architectural design during the last decades, marking the rise of a new area of study that engages with design cognition, computation and generative principles in contemporary design practice. Gero (Gero, 1994) enlists two main areas in the development of computer aided design: "the representation and production of the geometry and topology of designed objects" and "the representation and use of knowledge to support or carry the synthesis of designs". While the first category relates to the general-use off-the-shelf CAD tools that aim to increase the efficiency or aim to automate design and drafting activities, the second has given birth to novel generative approaches that regard computation as an aid to the design process and to explore design ideas. Generative design systems allow the formation of complex compositions, both formal and conceptual, through the implementation of a simple set of operations and parameters. This new understanding marks the emergence of innovative modes of design thinking. Here, the main challenge lies in the cultivation of computation as a tool that complements the designer's capabilities in the conceptualization and production of design artifacts in the contemporary architectural agenda (Ahlquist and Menges, 2011).

This paper discusses the role of parametric modeling as a generative design tool in architecture. Parametric design is a computational method that can act as both a generative and analytical method during design exploration, and has recently gained great acceptance from both practice, research and education. However, this instantaneous attention resulted in the redundant and superficial application of parametric principles. Therefore it is necessary to reframe the subject, explicating its potentials and limitations in architectural design. To this end, parametric systems will be discussed from the perspectives of design exploration, performative design and parametric design representation, while presenting some examples of their application. Finally, some of the limitations of parametric systems as an exploratory tool are discussed, mainly concerning its role in architectural design, design flexibility and complexity.

GENERATIVE DESIGN SYSTEMS IN ARCHITECTURE

Design has a dual meaning. It simultaneously means the act of designing an object (design as an activity), and the designed object as an end result of the design act (design as an artifact). This distinction is central in generative design systems: A generative system is a production system that does not specify the design artifact, but instead specifies a higher-level specification that encodes the "making" of the artifact, or the design procedure. Therefore, generative systems are said to precede formation over form, which indicates a fundamental shift from the modeling of a designed "object" to modeling of the design's "logic" (Leach, 2009). Generative design systems require the computational specification of the principles of the formation of a design (artifact), which opens up a design space for the exploration of design alternatives and variations. As such, generative systems suggest the delegation of some design tasks and intelligence from the human designer to the generative system, thereby claiming a degree of autonomy. However this does not mean that the generative system now becomes the designer, but that the human designer externalizes and encodes some of its working intelligence into the "generator" to carry out certain design tasks or solve problems. These specifications can be rules, constraints, parametric dependencies, genetic structures, case-bases etc.

Generative logic is nothing but new to design and architecture. Mitchell traces the roots of generative systems in general to philosophy, literature and musical composition, and architectural generative systems in particular to Leonardo da Vinci (Mitchell, 1979). According to Hanna and Barber, Jean-Nicolas-Louis Durand followed an analogue generative approach for the creation of neo-classical architecture by applying different combinations of building elements (Hanna and Barber, 2001). Louis Sullivan's plates that describe processes for reproducing floral ornamentation based on geometrical constructs, and Le Corbusier's Five Points of Architecture in which he formalized his style are accounted as examples of analogue generative systems before the use of computation in architecture by El-Khaldi (2007). A more recent example is Peter Eisenman, who used analogue transformational rules in architectural design synthesis (**1**). Eisenman's design concept operates on a system (a language) that permits creative action, generating an infinite number of utterances and making infinite use of finite means (Hays, 2000). Eisenman reflects this practice on the design of a series of houses (House I - X), where he states that "the house is not an object in the traditional sense - that is the end result of a process – but more accurately a record of a process" (Eisenman, 1977). This emphasis on the process over the end product, and the act of conceiving of architectural form suggests a generative principle as the essential driver during architectural synthesis.

The process of generative formation requires four elements: the start conditions and parameters (input), a generative mechanism (rules, algorithms etc.), the act of generation of the variants (output), and the selection of the best variant. The design artifact does not materialize until the fourth step, therefore a generative system is considered as a production system rather a representational construct. Moreover, "[t]he generative role of new digital techniques is accomplished through the designer's simultaneous interpretation and manipulation of a computational construct… The capacity of digital, computational architectures to generate "new" designs is, therefore, highly dependent on the designer's perceptual and cognitive abilities, as continuous, dynamic processes ground the emergent form" (Kolarevic, 2003).

Generative systems can be roughly classified into two categories: linguistic and biological (Shea, 2004; Oxman, 2006; Arida, 2004). A linguistic system is a grammar-based formalism where a set of compositional rules (syntax) govern and shape the design (semantics). The computational implementation of linguistic generative systems primarily manifests itself in shape grammars. Shape grammars define and apply a set of modification rules on a starter object (a shape) in order to generate new complex design. According to Knight, shape grammars are descriptive and generative in a way that the modification rules both describe the forms of the generated designs, and generate or compute designs (Knight, 2000). Biological generative design systems, on the other hand, adopt a different generative strategy, which takes nature and complex living organisms as a precedent and applies its principles in the derivation and transformation of architectural form (Hensel et al., 2010). Vincent further articulates on the emphasis placed on the becoming of the form rather than the resulting form itself (Vincent, 2009). Natural emergence, describing the ways in which complex natural systems evolve, self-organize and grow, contribute to architectural knowledge creation towards the production of complex architectural, and especially performative design (Weinstock, 2010). As such, a deeper engagement with the nature is pursued, which investigates the ways in which the principles of nature present useful concepts such as functional integration, performative capacity and material resourcefulness (Ahlquist and Menges, 2011).

This paper argues that parametric design can be classified as the third class of generative systems, due to their algorithmic foundation and their potential to expand the design exploration space through the variation of variables of the algorithm, namely the parameters. The rest of the paper will discuss parametric generative design systems.

1. For an extended discussion on the houses of Eisenman, Gandelsonas, 1982.

PARAMETRIC DESIGN SYSTEMS

Parametric systems are principally based on algorithmic principles. Therefore, it is necessary to first argue the role or algorithms and algorithmic thinking in design, to be able to discuss further parametric systems. An algorithm is a finite set of instructions that aim to fulfill a clearly defined purpose in a finite number of steps. An algorithm takes one value or a set of values as input, executes a series of computational steps that transform the input, and finally produces one value or a set of values as output. Furniture assembly instructions or cooking recipes are simpler informal forms of algorithms, where the furniture parts or the ingredients are the inputs, the finished furniture or the meal is the output, and the assembly or cooking process is the procedural steps to be followed. The power of algorithms lay in the ability to solve a wide range of computational problems including but not limited to sorting and searching, data structure operations, combinatorial problems, numerical problems (including random number generation), and computational geometry (Cormen, 2001). In terms of procedural workflow, three basic control instructions that implement fundamental operations are sequence (executing instructions in order), selection (choosing which instructions to execute based on conditional if-then statements) and iteration (repeating instructions in a linearly or recursively) (Chang, 2003).

Algorithmic thinking and algorithmic design has much relevance to the concept of generative design. Terzidis argues that the inductive strategy of algorithms can explore generative processes or to simulate complex phenomena. Algorithms can be regarded as extensions to the human brain and may facilitate a leap to the areas of unpredictable potential (Terzidis, 2011). Burry suggests two motivations for scripting in design: increasing productivity to iterate faster, and gaining control design to liberate oneself from the limitations of black-box modeling software (Burry, 2011). Algorithms can computationally generate and manipulate design entities such as geometric form, design variables, data structures that contain numeric or geometrical entities, mathematical expressions and operations, and logical operations. This level of control over design in an 3D modeling environment allows the designers (or in this case developers) extend functionality, or evaluate certain conditions and respond appropriately. Therefore, an algorithm can effectively deal with the complexities of design much beyond form with precision, and translate these into architectural properties.

Parametric design is a subcategory of algorithmic design, and is strictly based on an algorithmic construct. Computationally speaking, there is no difference between algorithmic and parametric systems; algorithms by default operate on parameters, and a parametric system's fundamental component is the algorithm itself, called the schema or definition. However, different than algorithmic design, parametric systems emphasize the explicit and direct manipulation of the parameter values in order to induce a change on the design artifact. This simple difference between a purely algorithmic versus parametric design manifests itself only during the design process, where the parameter values are changed by the designer in order to manipulate the design geometry in search of the optimal design solution.

DESIGN EXPLORATION BASED ON PARAMETRIC LOGIC

Parametric (algorithmic) design allows the articulation of procedures for solving both well-defined problems with a clear target, and complex ill-defined problems having several workable solutions. While the former can be regarded rather as a deterministic approach suitable for well-structured problems, it is the second type that is more representative of the creative design process. Design problems offer no single best solution, but a class of satisfactory solutions (Simon, 1969). Therefore, the designer needs to be able to define, redefine and change the design problem in the light of the solution while navigating within this design space of possible solutions (Cross, 2001b). Similarly, the ability to investigate a large number of design alternatives is critical to finding successful designs, and "a main distinguishing mark of expert designers compared with novices is the creation of comparatively more alternative problem formulations" (Akin, 2001). Furthermore, design is considered as an iterative *divergence / convergence* process, where designers are supported and encouraged to generate the widest possible range of concepts, and then to explore, evaluate, and modify these (emphasis added by the author) (Liu et al., 2003).

Conventional CAD tools' primary aim is the representation of the final design form, and they rely on single-state design. On the other hand, parametric modeling as a design synthesis method allows for the divergence of the design space in order to explore many variants of the same parametric model. As such, the design principle expressed in parametric associations allows the designer to explore an array of design options through time, revisit previous design alternatives and improve the design artifact during the design process (Aish and Woodbury, 2005).

Parametric modeling can facilitate a wider search area for design exploration by allowing the automatic generation of a class of alternative design solutions. A change in an input parameter triggers a simultaneous change in the form, generating variations on the form while maintaining the underlying coherence of the schema. **Figure 1** shows the numerous geometric arrangements of the British Petrol Headquarters in Sunbury by Adams Kara Taylor, where the creative design exploration of the roof structure is based on a parametric approach that takes into consideration both aesthetic and structural quality and structural viability (Vanucci, 2008). The focus is to establish a set of relationships so to allow the easy update of the overall geometry as the parameters are varied during the creative design exploration process (**Figure 2**). As such, the parametric model becomes a controlled environment on design exploration in which the search for a fitter – better design alternative can be carried out. Similarly, Oxman et al. articulate that "associative geometry may support a design approach in which a geometrically, or tectonically defined series of dependency relationships is the basis for a generative, evolutionary design process. One can manipulate a particular shape or form and study many alternatives by changing the variables, or parameters, defining the geometry of an object or assembly" (Oxman and Oxman, 2010).

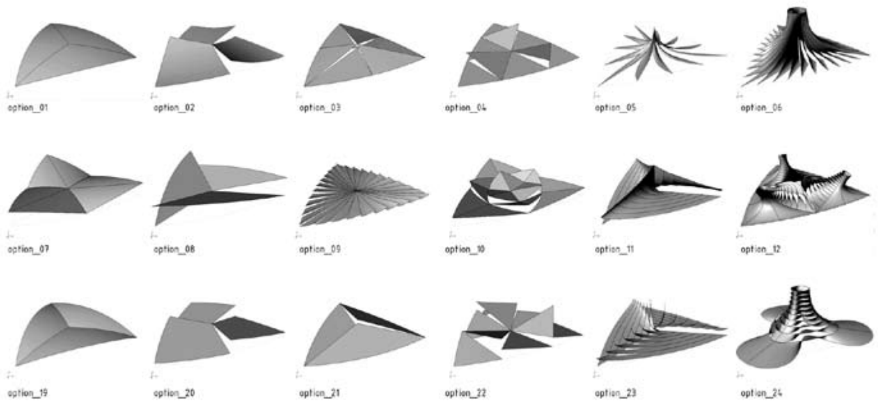

Figure 1. British Petrol Headquarters in Sunbury by Adams Kara Taylor (Adams Kara Taylor, personal communication, February 2012).

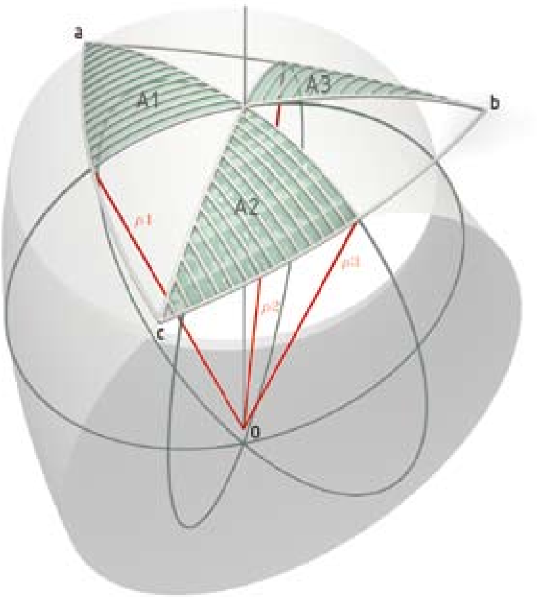

Figure 2. The diagrammatic representation of the associative geometric elements.

PARAMETRIC SYSTEMS FOR PERFORMATIVE DESIGN EXPLORATION

The advent of computer-aided design tools in architecture has led way to a new aesthetical understanding that is freed of Euclidean geometry, and an enthusiasm over the unconstrained experimentation on a fluid n-dimensional design space. Computational tools are falsely characterized by this tendency, and are believed to only generate form that is sculptural at its best. However, design is not only a creative but also a rational act, one that still responds to *firmitas* and *utilitas* as much as *venustas* in Vitruvian terms. The understanding that prioritizes how the design artifact looks like (form) over how it behaves (performance) eventually leads to forms that are unbuildable, unhabitable and therefore trapped in the digital world. Such classes of design methodology are self-referential and do not take into consideration the context within which the design artifact exists such as the site, the owner requirements, the users, function, environment, ergonomics and performance. Thereby lies the danger of designing in a context-free world that is primarily driven by the designer's aesthetic and plastic senses. Conversely, a much wider range of external factors should be addressed during design where performance is the driving force behind form. Architecture should "perform rather than simply form in multiple contextual and performative areas… formal distortions need to have purpose or cultural relevance..." (Meredith, 2008).

Parametric modeling has great potential in addressing performative issues in architecture as a tool that allows the navigation of the parametric search space with respect to measurable performance criteria. Parameterization can be based on performance metrics to allow the solution space be meaningfully traversed while taking performance as a guiding tool. Moreover, the conventional interaction between the architect and engineer based on post-rationalization can be altered for a parallel (and even cyclic) process that integrates multi-disciplinary design synthesis and analysis (see the Aviva Stadium). Computational tools, and particularly parametric design, can facilitate the iterative and dynamic coordination of cross-disciplinary intelligence that is distributed across various analytical tools and methods (Kocatürk and Medjdoub, 2011). In this scenario, performative principles can either be encoded in the parametric model itself, generating alternatives that all comply to the performance criteria at hand, or the parametric model can be coupled with external performance simulation tools that probe the search space that is generated by the parametric model. In the latter case, the process comprises a generate / evaluate / modify cycle in search of the optimal performance, and is more suitable for addressing complex performance issues that exceed well beyond the performative knowledge limit of parametric modelers, such as structural performance.

In performative parametric design, the divergence of the design space is facilitated by the parametric combinations, and the convergence takes place on the basis of performative optimality. For a performance-integrated design process, some critical issues are multi-disciplinary thinking, early performance integration and a mutual agreement on the driving performative parameters. In order to form such a mutual agreement, there are two important issues to be considered: to parameterize the design suitable for downstream analysis, and to structure the schema to support continuous transformation during design exploration. The designer needs to explicitly describe and resolve both the parametric schema and the performative parameters prior to form exploration.

Performative design principles can be integrated either early in the design process where design concepts and main geometry are being worked out, or later on during detail design where performance optimization of systems is carried out. The Aviva Stadium in Dublin, Ireland and the façade of the Kilden Performing Arts Center in Kristiansand, Norway are two works that are exemplary of these two approaches.

Aviva Stadium (Dublin, Ireland) is a sports stadium designed by Populous and engineered by Buro Happold (**Figure 3**). A performative parametric approach was adopted at the onset of the project, which enabled integrated multi-disciplinary design communication between project participants. During the design process, the architects ultimately were driving the overall form and cladding of the building, and the engineers driving the structural member sizing / positioning (Shepherd et al., 2011). On the architectural side, certain form explorations were being made in response to certain criteria such as concourse width requirements, floor area ratios, or simply beautifying the shape (**Figure 4**). On the engineering side were

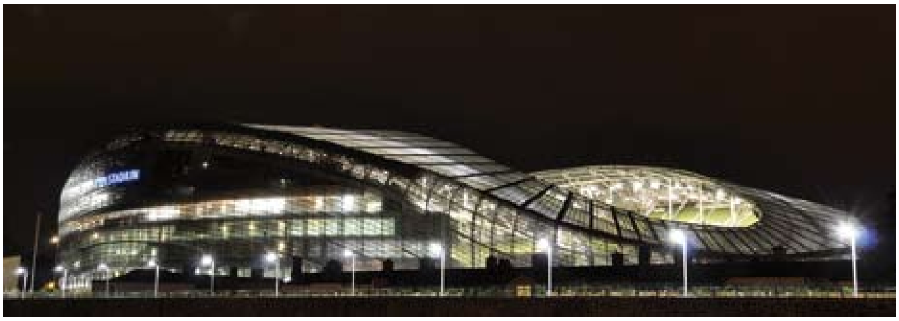

Figure 3. Aviva Stadium (http://en.wikipedia.org/wiki/File:Aviva_Stadium_by_Night.jpg), retrieved 18 December 2011).

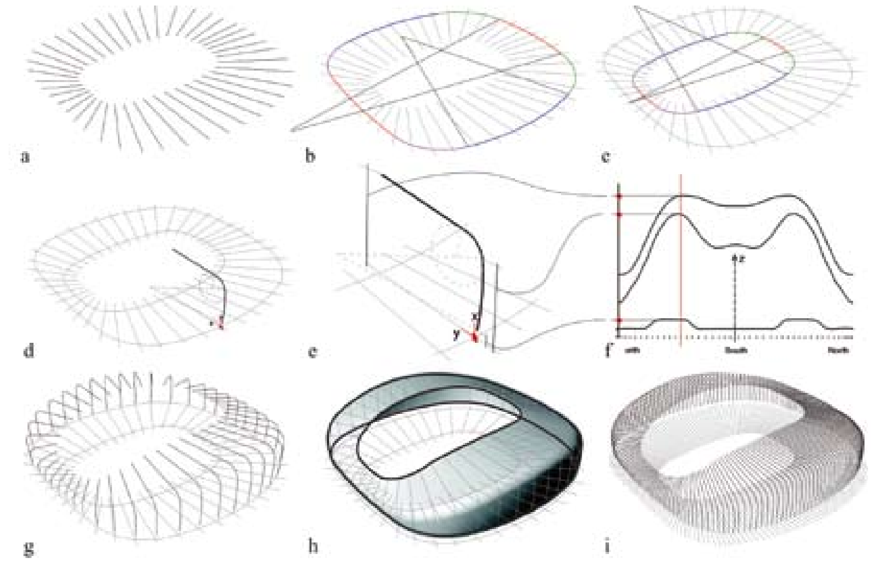

Figure 4. The geometric definition of the Aviva Stadium envelope (Shepherd et al., 2011).

the structure of the roof trusses and cladding system designed as a rain screen consisting of inter-locking louvers. A single parametric model was shared between the architectural and engineering offices, which acted both as a design tool and a coordination platform. This allowed the integration of the design processes of the form, structure and façade, allowing rapid response to design changes. Analysis tools were coupled with the parametric model and provided quick analytical feedback to the geometry. The sharing of the parametric model across the other design members and the fully integration of the engineering analysis applications could realized the benefits of a parametric approach (Hudson et al., 2011)(**2**).

Designed by ALA Architects and engineered by the designtoproduction company, the façade of Kilden Performing Arts Center in Kristiansand, Norway is another recent example of parametric modeling. The façade (also called the Wave Wall), a monumental curved wall that separates the foyer from the outside (**Figure 5**). Cantilevering up towards the waterfront, the timber façade intersects with a vertical glass and steel facade in both interior and exterior parts. The geometry of the façade is a ruled surface that spans between a straight upper and a curved lower edge. Here, a parametric system was used not during the form-finding process of the curvilinear roof, but during detail design for the parametric optimization of form and performance. Designtoproduction team got involved in the project during the detailed description of the wooden façade system, and closely collaborated with the architects, wood manufacturers and

2. For a detailed discussion on the Aviva Stadium, see (Shepherd et al., 2011) and (Hudson et al., 2011).

the structural engineers during the process (Fabian Scheurer, personal communication, 23 December 2011).

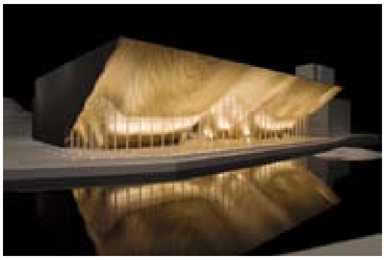

Figure 5. Model photo of Kilden Performing Arts Center (http://www.ala.fi/works/project/88-kilden#gallery-anchor, retrieved 18 December 2011).

There were a number of criteria that led to the use of a parametric / algorithmic approach, including the budget constraints, ease of manufacturing (so that the beams could be produced from straight laths), the number of wooden beams (consisting of 305 straight primary beams connected to the building's steel structural system, 1769 single-curved secondary crossbeams 12248 and straight but twisted oak cladding boards), the high level of precision required to bring parts together seamlessly, and finally the performance requirements that are posed on the beams in order to carry the oak cladding boards and withstand frontal and lateral wind loads. The roof is clad by straight oak boards that are only twisted around their longitudinal axis. Therefore, "all generatrices had to be aligned with the building axes, a demand that could not be met with the default 'loft' method found in standard CAD packages, but needed a custom NURBS definition" (Scheurer and Stehling, 2011) (**Figure 6**). A modular system was developed that guarantees the continuity and straightness of all cladding boards and gaps, which is also the basis for all subsequent planning steps. The mathematically precise definition and the exact positions of the roof beams, as well as their assembly details were worked out with parametric modeling. The parametric system consisted of a set of highly specialized tools that covered the complete workflow from input geometry to fabrication data and also could be adjusted or substituted individually without affecting the functionality of the system as a whole. Structural analysis software was used for structural optimization, which required the execution of several rounds of exporting, validating and applying changes until a satisfactory solution was found.

For a detailed discussion on the Kilden facade (Stehling and Scheurer, 2011; Scheurer and Stehling, 2011; designtoproduction, 2011).

DESIGN REPRESENTATIONS IN PARAMETRIC MODELING

The build-up and manipulation of external design representations has always been considered as a crucial component in enhancing design cognition and exploration. Akin articulates that architecture is a representation saturated problem domain, mentioning two distinct modes of representation in design: analog and symbolic (Akin, 2001). Analog representations are drawings, sketches, physical and electronic models, which have a strong correspondence to reality and therefore lend themselves easily to the evaluation of important design performance issues such as composition, contextual congruency, and

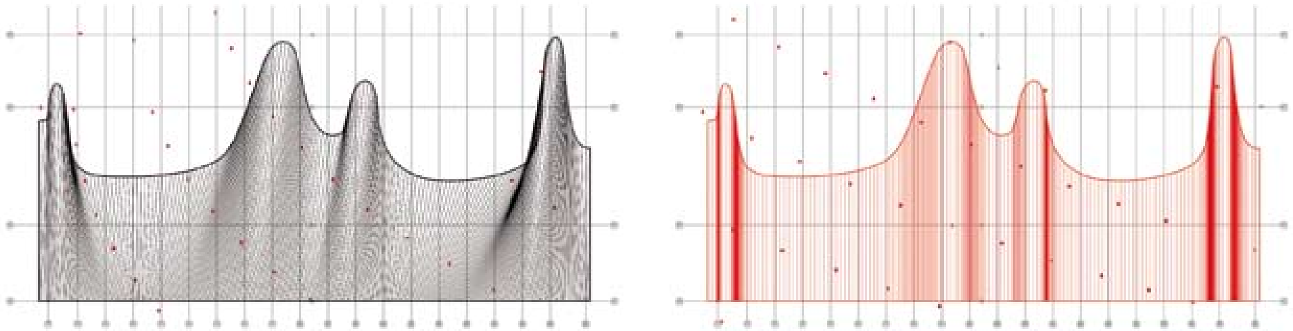

Figure 6. The default loft method and the custom NURBS definition developed for Kilden Performing Arts Center (Scheurer and Stehling, 2011).

constructability. Symbolic representations are physical or mathematical descriptions related to definition of performance such as heat transfer, light and sound distribution etc. A parametric system can be considered as a tool to mediate between the content of both analog and symbolic representations, capturing the procedural knowledge about form, and declarative knowledge on formation and performance in the definition of algorithms and parameters. Therefore, parametric design tools employ multiple views of the design space. On one side, there is the 3D model view that displays the geometric (analogic) representation. On the other side, there exists an editor that allows the designer to encode the algorithm, namely the schema. This editor is either textual or visual. Visual schema editors (e.g. Grasshopper3D plug-in for Rhinoceros by McNeil or GenerativeComponents by Bentley) have found widespread use due to the low level of technical knowledge required on programming languages. Here, the schema is based on a directed graph data structure, where the graph nodes are geometric components or parameters, and the edges are the paths of the one-directional data transfer.

Decoupling of the geometric view and the algorithmic view enables the declaration of different types of parameters independent from the geometry, and becomes particularly useful when the geometry needs to be defined and generated considering non-spatial contextual parameters such as occupancy load, proximity constraints, land-use ratio or color range (Madkour et al., 2009), as captured by symbolic design representations. As such, parametric systems facilitate a dual communication between the geometry as the core design representation, and the symbolic design requirements.

SOME LIMITATIONS AND MISCONCEPTIONS ON PARAMETRIC DESIGN SYSTEMS

Lately, there is great attention directed towards parametric modeling. This is mainly due to the recently emerging visual parametric modeling tools, which hide the algorithmic complexity of parametric models behind a visual programming interface, thereby lowering the necessary technical skills for the use of computation and contributing to its widespread adoption in architectural design. However, groundless and superficial application of parametric principles poses a threat to its proper use as a design tool. Therefore, it is necessary to discuss some of the misconceptions and drawbacks of parametric modeling as a generative tool to be able to realize its potentials to a full extent.

PARAMETRIC DESIGN IS A DESIGN METHOD, NOT A STYLE OR AN -ISM

In his well-known essay, Schumacher refers to parametric design as “a profound style [that] has been maturing within the avant-garde segment of architecture”, coining the word parametricism as a newly emerging mode of architectural expression that “offers a credible, sustainable answer to the drawn-out crisis of modernism that resulted in 25 years of stylistic searching” (Schumacher, 2009). Although there is a degree of stylistic consistency with much of parametric architecture -one that offers a viable way to generate free-form design and complex geometry-, it certainly is not exclusive to any sort of formal language, and definitely is not a style either. Meredith points out to the threat against parametric design that it can quickly devolve into a totalizing visual aesthetic built upon a formalist tendency (Meredith, 2008). Similarly, Moussavi states that parametric design as a style disposing itself of the restraints of external parameters and promotes the autonomy of architectural forms, while it cannot advance beyond new ways of shaping matter to produce unexpected spaces (Moussavi, 2011). It is artificial to label parametric design as a global movement, as a brand new paradigm, or as a profound avant-garde architectural style. On the contrary, parametric modeling is a method to control design complexity, and can be applied to any design process that has something to do with design exploration. The design problems that parametric methods can be directed at range from buildings that have complex performative, functional and contextual requirements (as illustrated in the examples above), to much simpler forms of design that deal with basic geometry and need to satisfy much simpler design constraints (**Figure 7**).

At the same time, it can be argued that design has always been parametric, and the algorithmic parametric design logic is not exclusive to computational tools. In general, design is an iterative activity that involves the continuous generation of a design artifact and the evaluation of its fittingness to requirements. During this refinement process, every new iteration entails either the radical reformulation of the design concept, or -most of the time- the modification of some design parameters to generate a variation of the previous design state (Cross, 2001a). The latter case represents a mode of design thinking that is based on the unintentional articulation of a parameter space that steers the design process. Lawson demonstrates how a similar informal generative principle is applied by Bill Howard in the design of University Center building in Cambridge, following an informal parametric constructive logic that explores the relationship of building components of different scales including internal partitioning, structural beams, cladding, building site and even its neighborhoods (Lawson, 1980). Such relational aspects and the ways in which these aspects interact with each other have always been integral to the design process and design exploration. The recent advance of computational systems have only proliferated and expedited the explicit use of parameters due to the underlying algorithmic structure, with increased precision and ease of application.

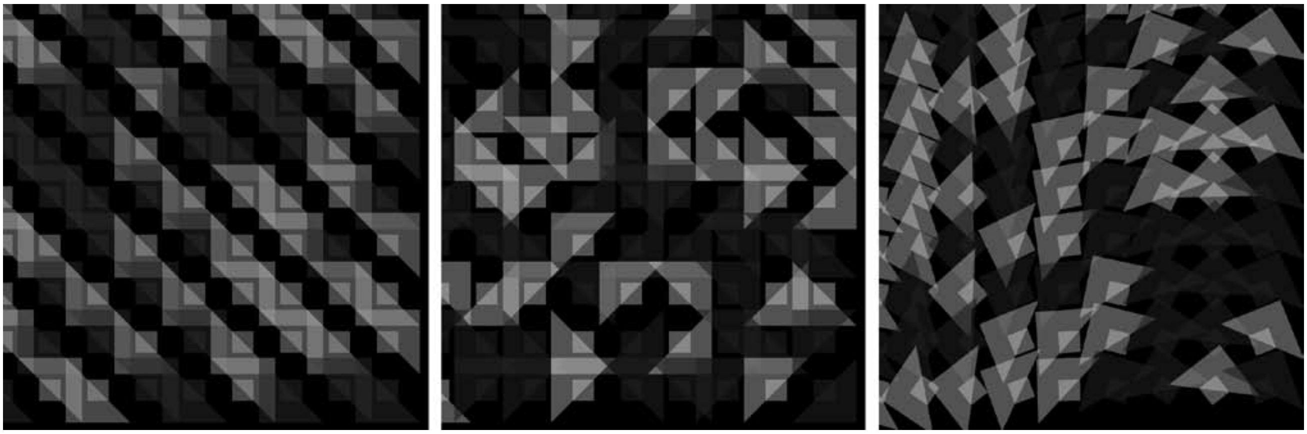

Figure 7. Design explorations on "variation and repetition" with algorithmic design. Student work from the Basic Design course in 2011 at METU Faculty of Architecture. Different design variations are explored by varying the rotation angle of the design elements. Generated with Processing.

PARAMETRIC DESIGN EQUALS COMPLEX GEOMETRY

Another misconception is that parametric design is the only way to generate complex geometry. As a matter of fact, complex geometry in architecture was present even before computation was an applied field in architectural design, as can be seen in the work of Frei Otto, Jorn Utzon, Pier Luigi Nervi, Felix Candela, Anton Gaudi and others'. Each of these architects found innovative techniques to deal with the complexities of free-form geometry in their work. Hanging chain models and stacking plaster pieces, for example, was physical explorative tools for Antonio Gaudi during his form-finding process (Jos, 1989). As expected, however, this was neither a smooth, nor an efficient process. The findings of these scale models had to undergo major changes while transitioning them into the material structure, due to the lack of a formal method to map between these two realms (Kilian, 2006). Therefore, it relied solely on the expertise and skills of the architect to negotiate between a system based on a material abstraction (the models) and the real building. Another similar example is the Sydney Opera House by Jorn Utzon, a competition project that he was awarded first prize for in 1957 and opened on 1973. Today the building is considered as a masterpiece cherished by many architects and engineers. However, the process of design exploration generated a great deal of controversy during the time of its construction (Akin, 2004). The geometry of the roof structure was originally undefined, and was initially deemed to be unbuildable (Yeomans, 1973). During the following 5 years of concept design, the engineers and the architect had to rationalize the roof into a constructible form that allows for the use of a single formwork and therefore a single curvature during the construction process (Arup and Jenkins, 1968). Moreover, its design and construction required significant compromise and adaptation, while spanning over sixteen years and exceeded the budget by almost fourteen times. These two examples by Gaudi and Utzon represent the type of geometry that have the potential respond to computational approaches, and particularly to performative parametric systems. The design of such complex and non-standard geometries strive for parametric approaches, and can be efficiently dealt with using a higher-level manipulation of geometry with parameters and algorithms. Computation can eliminate the need for physical modeling or trial-and-error strategies in form optimization, and increase the efficiency of design analysis and synthesis with increased precision of form and details.

PARAMETRIC DESIGN DOES NOT OFFER UNLIMITED FLEXIBILITY

As discussed above, design exploration necessitates a certain level of agility to change and the ability to reformulate the design problem and to react to external forces. Design exploration with parametric modeling requires a flexible and responsive representation that can respond to change. In an ideal situation, the representation, or the parametric model, should evolve and update its internal structure in parallel with the explorative path. However, the algorithmic principles that underlie parametric models operate in a very definitive way, such that "each step of an algorithm must be precisely defined; the actions to be carried out must be rigorously and unambiguously specified for each case" (Knuth, 1997). In consequence, the flexibility of parametric modeling is a result of -and therefore limited to - the parameters that it internally describe, whereas the schema is relatively more resistant to change. Gerber points out to the need to remake the parametric model over and over as the topology of the project changes, and Burry argues that it is commonplace in parametric modeling that the designer is forced to edit or remodel the relational graph (Gerber, 2007; Burry, 2007). The repair of the schema during design exploration is a particularly expensive task. It is particularly time consuming if the modifications propagate through a complex network of relationships, changing the flow of data with downstream consequences throughout the schema (Davis et al., 2011b).

Parametric modeling requires intentionality: It imposes the designer to generate the algorithm of the design with a great deal of explicitness prior to parametric exploration. Parametric models are not infinitely flexible; on the contrary change can be accommodated only when it is within the current problem definition. A drastic problem reformulation that requires algorithm alteration might cause the parametric model collapse at once. Kilian emphasizes the risk that parametric modeling approach might even prematurely freeze design due to the investments previously made into the parametric description, rather than supporting exploration in a higher level of detail In the later phases of design (Kilian, 2006). Therefore, Kilian claims, it offers a constrained explorative area based on dimensions or composition, therefore falling short on generating conceptual variety.

PARAMETRIC MODELING DOES NOT REDUCE DESIGN COMPLEXITY

Complexity is probably one of the central terms that describe the contemporary design problems in architecture. The increasing design complexity in architecture is not only due to external stimuli such as increasing building performance requirements, new building functions, user requirements, urban settings, spatial configurations, integrated design processes etc., but also due to new formal interest in free-form geometry and the underlying mathematical and geometric concepts. The need to address a growing number of concerns and integrally deal with them introduces increased levels of complexity in architecture (**3**). Design computation has the claim to be able to help the designer take complex and complicated decisions with greater confidence compared to the conventional case (Bittermann, 2009).

While providing the designer with a set of tools to expand her cognitive and operational abilities in complex design settings, computation - and also parametric modeling- poses new challenges both in the practical use of this new medium and some of the consequences computation brings about. First and foremost, parametric design requires a solid technical knowledge on computation and mathematics, one that architects are not typically educated on or even accustomed to. The designer often needs to switch between two opposite cognitive states: creative design thinking that highlights novelty and innovation, and systematic formalization of the design into a computational realm. According to Aish and Woodbury, parameterization "increases complexity of both designer task and interface as designers must model not only the artifact being designed, but a conceptual structure that guides variation". The necessity of explicitly externalizing the relevant conceptual and constructive structure contradicts the ambiguity that is part of healthy design processes (Aish and Woodbury, 2005).

3. Simon (1969) describes complex systems as made up of a large number of parts that interact in a nonsimple way, where the whole is more than the sum of its parts and, it is not a trivial matter to infer the properties of the whole.

The mathematical and computational complexity of the (free-form) geometry that is being designed needs to be worked out in in every detail during algorithmic / parametric modeling by the designer herself. From a process point of view, parameterization requires the mapping of a design problem into a limited number of parameters that lead to a solution space large enough to contain all necessary design solutions, and then the instantiation of these parameters with the "correct" values (Scheurer, 2010). Thereby arises the question of correctness: what are the determinant parameters that can describe a design problem, and how to deal with a set of parameters that interact with each other in complex ways, especially those in conflict with each other? Moreover, the number of parameters in the schema can be also problematic, such that with each new parameter the design solution space grows exponentially and becomes unmanageable. The problem of excessive amount of possible solutions in the parameter domain (also known as combinatorial explosion), and the multi-objective nature of design problems pose a challenge for computational methods (Bittermann, 2011). Moreover, the problem of the breadth of parametric solution spaces hamper the systematic evaluation of each design alternative due to time and other restrictions (Turrin et al., 2011).

To tackle these limitations, several computational approaches were proposed as linked to parametric systems. Woodbury develops parametric design patterns as reusable, abstract solutions to problems that commonly occur during parametric modeling (Woodbury, 2010). Turrin and Bittermann propose to link parametric models with genetic algorithms to deal with extensive search spaces for performance-based design exploration (Bittermann, 2009, Turrin et al., 2011). Davis aims to improve the illegibility of unstructured parametric schemata with modular programming principles in complex and collaborative environments (Davis et al., 2011a). However, in either of these cases, design complexity persists, moved to a higher level of abstraction.

CONCLUSION

Although the technological requirements have already been met for their effective use in the field of architectural design, the potential of parametric generative systems are yet to be explored in detail in architectural design. Parametric design tools emphasize diversity of design strategies, rather than providing the designer with a comprehensive framework of technologies or concrete approaches. The potential of computational design to change the way we design, rather than to merely augment or replace human designers (Akin, 1990) holds true for parametric generative systems as well.

This paper discussed parametric approaches as a generative tool in architectural design. Parametric design systems distinguish themselves from other generative systems in the way they allow a stepwise control over the form during the design process, which proves to be useful especially during design exploration. Their responsiveness and adaptability to both internal and external stimuli, namely the dynamicity of the creative design process and other contextual conditions, make these systems a suitable strategy while designing in complex design settings. Moreover, parametric manipulation of form is particularly valuable in performance-driven design processes, facilitating the rapid cycling of performance analysis and design synthesis as an integrated process.

In the future, the development of generative design systems will undoubtedly occupy the agenda of computational design in both theory and practice, as new paradigms connect generative methods with the acts of architectural making. In a time of rapid technological change, architects can regain control over such novel computational strategies by positioning themselves in a constructive yet critical position, so to be able to effectively integrate these into their design processes.

REFERENCES

AHLQUIST, S., MENGES, A. (2011) Computational design thinking, in A. Menges and S. Ahlquist (eds.) *Computational design thinking,* Wiley.

AISH, R., WOODBURY, R. (2005) Multi-level interaction in parametric design, in A. Butz, B. Fisher, A. Krüger and P. Oliver (eds.) *SmartGraphics, 5th Int. Symp., SG2005, Lecture Notes in Computer Science,* Springer, Berlin.

AKIN, O. (1990) *Computational Design Instruction: Towards a Pedagogy*. Electronic Design Studio; 301-16.

AKIN, O. (2001) *Variants in design cognition*. Design knowing and learning: Cognition in design education, 105-124.

AKIN, O. (2004) Three Fundamental Tenets for Architectural Ethics, *invited paper for the ACSA Teacher's Conference.* Cranbrook Academy of Art, MI.

ARIDA, S. (2004) *Contextualizing Generative Design.* MSc thesis, Massachusetts Institute of Technology, Boston, MA.

ARUP, S.O. , JENKINS, R. (1968) *The evolution and design of the Concourse at the Sydney Opera House,* Ove Arup and Partners.

BITTERMANN, M. (2009) *Intelligent Design Objects (IDO): a cognitive approach for performance-based design,* Delft University of Technology, Delft, Netherlands.

BITTERMANN, M.S. (2011) *A Computational Design System with Cognitive Features Based on Multi-objective Evolutionary Search with Fuzzy Information Processing*. Design Computing and Cognition'10; 505-24.

BURRY, J.R. (2007) *Mindful Spaces: Computational Geometry and the Conceptual Spaces in which Designers Operate*. International Journal of Architectural Computing, 5:4; 611-24.

BURRY, M. (2011) *Scripting Cultures: Architectural Design and Programming.* Wiley.

CHANG, S.K. (2003) *Data structures and algorithms.* World Scientific River Edge, London.

CORMEN, T.H. (2001) *Introduction to algorithms.* The MIT Press.

CROSS, N. (2001a) *Design cognition: results from protocol and other empirical studies of design activity*, in C. Eastman, W. Newstatter & M. McCracken (eds.) *Design knowing and learning: cognition in design education,* Elsevier: Oxford, UK.

CROSS, N. (2001b) *Designerly ways of knowing: design discipline versus design science*. Design issues, 17:3; 49-55.

DAVIS, D., BURRY, J., BURRY, M. (2011a) Untangling Parametric Schemata: Enhancing Collaboration through Modular Programming, in P. Leclercq, A. Heylighen & G. Martin (eds.) *Proc. of CAAD Futures 2011: Designing Together,* Liege, Belgium.

DAVIS, D., BURRY, M., BURRY, J. (2011b) The flexibility of logic programming, *Proceedings of the 16th International Conference on CAADRIA.* Newcastle, Australia.

DESIGNTOPRODUCTION (2011) *Kilden Performing Arts Center, factsheet 2010/12* [WWW document]. URL www.designtoproduction.com ,accessed 20 December 2011.

EISENMAN, P. (1977) House VI, *Progressive Architecture*, 58**;** 57-67.

EL-KHALDI, M. (2007) *Mapping Boundaries of Generative Systems for Design Synthesis.* MSc thesis, Massachusetts Institute of Technology, Boston, MA.

GANDELSONAS, M. (1982) *From structure to subject: the formation of an architectural language*, Eisenman, P. House X, New York: Rizzoli.

GERBER, D.J. (2007) Parametric practices: models for design exploration in architecture, PhD thesis, Harvard University.

GERO, J. (1994) Preface, in J. Gero & E. Tyugu (eds.) *Formal Design Methods for CAD.* Elsevier: Amsterdam.

HANNA, R., BARBER, T. (2001) An inquiry into computers in design: attitudes before-attitudes after, *Design Studies*, 22:3**;** 255-81.

HAYS, K.M. (2000) *Architecture theory since 1968,* MIT Press: Cambridge, MA.

HENSEL, M., MENGES, A. ,WEINSTOCK, M. (2010) *Emergent technologies and design: towards a biological paradigm for architecture,* Routledge, UK.

HUDSON, R., SHEPHERD, P. & HINES, D. (2011) Aviva Stadium: A case study in integrated parametric design, *International Journal of Architectural Computing,* 9:2**;** 187-204.

JOS, T. (1989) The Model: Antoni Gaudi's hanging model and its reconstruction, PhD thesis, Universitat Stuttgart, Stuttgart.

KILIAN, A. (2006) Design exploration through bidirectional modeling of constraints, PhD thesis, Massachusetts Institute of Technology, Boston, MA.

KNIGHT, T. (2000) *Shape grammars in education and practice: History and prospects* [WWW document]. URL http://www.mit.edu/~tknight/IJDC/, accessed 6 December 2011.

KNUTH, D.E. (1997) *Art of Computer Programming, Volume 1: Fundamental Algorithms.* Addison-Wesley, Reading, MA.

KOCATÜRK, T., MEDJDOUB, B. (2011) *Distributed intelligence in design,*Wiley Online Library.

KOLAREVIC, B. (2003) Digital Morphogenesis, in B. Kolarevic (ed.) *Architecture in the digital age: design and manufacturing,* Taylor & Francis.

LAWSON, B. (1980) *How designers think,* Architectural Press London.

LEACH, N. (2009) *Digital Morphogenesis*, *Architectural Design*, 79:1; 32-7.

LIU, Y.C., CHAKRABARTI, A., BLIGH, T. (2003) Towards an ideal approach for concept generation, *Design Studies*, 24:4;341-55.

MEREDITH, M. (2008) Never enough, in M. Meredith, M. Sasaki (eds.) *From control to design: Parametric/algorithmic architecture,* Actar-D.

MITCHELL, W.J. (1979) *Computer-aided architectural design,* Van Nostrand Reinhold.

MOUSSAVI, F. (2011) Parametric software is no substitute for parametric thinking., *The Architectural Review,* http://www.architectural-review.com/issues/october-2011/parametric-software-is-no-substitute-for-parametric-thinking/8620000.article.

OXMAN, R. (2006) Theory and design in the first digital age, *Design Studies*, 27:3; 229-65.

OXMAN, R., OXMAN, R. (2010) New Structuralism: Design, Engineering and Architectural Technologies, *Architectural Design*, 80:4; 14-23.

SCHEURER, F. (2010) Materializing Complexity, in R. Oxman and R. Oxman (eds.) *New* Structuralism: Design, Engineering and Architectural Technologies, *Architectural Design*, 80:4.

SCHEURER, F. & STEHLING, H. (2011) Lost in Parameter Space? *Architectural Design*, 81:4; 70-79.

SCHUMACHER, P. (2009) *Parametricism: A new global style for architecture and urban design*. Architectural Design, 79:4; 14-23.

SHEA, K. (2004) *Directed randomness,* in N. Leach, D. Turnbull and C. Williams (eds.) *Digital Tectonics,* Wiley-Academy, UK.

SHEPHERD, P., HUDSON, R., HINES, D. (2011) *Aviva Stadium: A parametric success*. International Journal of Architectural Computing, 9:2, 167-186.

SIMON, H.A. (1969) *The sciences of the artificial,* MIT Press.

STEHLING, H., SCHEURER, F. (2011) *Waved Wooden Wall*, in G.R. and S.B. (eds.) *Fabricate: Making Digital Architecture,* Riverside Architectural Press.

TERZIDIS, K. (2011) *Algorithmic form*, in A. Menges and S. Ahlquist (eds.) *Computational Design Thinking,* Wiley.

TURRIN, M., von BUELOW, P. and STOUFFS, R. (2011) Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms, *Advanced Engineering Informatics*, 25-4; 656-75.

VANUCCI, M. (2008) *Open systems: approaching novel parametric domains*, in M. Meredith and M. Sasaki (eds.) *From Control to Design: Parametric / Algorithmic Architecture,* Actar.

VINCENT, J. (2009) Biomimetic Patterns in Architectural Design, *Architectural Design*, 79:6; 74-81.

WEINSTOCK, M. (2010) *The architecture of emergence: the evolution of form in nature and civilisation,* Wiley & Sons, Oxford.

WOODBURY, R. (2010) *Elements of parametric design*, Routledge, Oxford.

YEOMANS, J. (1973) *The Other Taj Mahal: What Happened to the Sydney Opera House,* Longman Australia.

Alındı: 03.01.2012, **Son Metin:** 02.04.2012

Anahtar Sözcükler: parametrik tasarım; üretken sistemler; tasarım araştırmaları; performansa dayalı tasarım.

MİMARLIKTA PARAMETRİK ÜRETKEN SİSTEMLER İLE YARATICI TASARIM ARAŞTIRMALARI

Bu yazı, üretken bir araç olarak parametrik tasarım sistemlerini tartışmaktadır. Algoritmik altyapıları nedeniyle parametrik sistemler, tasarım geometrisi üzerinde daha etkin sayısal denetim sağlarlar. Değişen tasarım ölçütlerine yanı verebilme özellikleri, parametrik modellerin karmaşık ve dinamik tasarım gereksinimlerini karşılayabilmelerine olanak sağlar. Tasarım sürecinde formun parametrik denetimi, özellikle performansa dayalı tasarımda performans değerlendirmesinin tasarım sentezinde kullanılmasını sağlayabilmesi özelliği ile öne çıkar. Buna rağmen, sayısal bir metot yerine yeni ortaya çıkmakta olan mimari bir stil olarak görülmeleri, parametrik sistemlerin gerçek potansiyelini gölgelemektedir. Ayrıca, parametrik modeller, temsil esnekliği ve tasarım karmaşıklığı konularındaki sınırlılıkları nedeniyle, etkili tasarım araştırmasını aksatabilirler. Bu bağlamda, parametrik sistemlerin potansiyel ve eksiklikleri konusunda eleştirel bir farkındalık oluşturmak, tasarımda verimli olarak kullanılabilmeleri açısından önemlidir.

İPEK GÜRSEL DİNO; BArch., MSc., PhD.
Received MSc. in Architecture from Carnegie Mellon University (Pittsburgh, USA) with an interest on computational design, and PhD in Architecture from the Design Informatics chair of Delft University of Technology (Delft, the Netherlands). Has led and participated in research projects on computational support for building performance assessment. Research interests include information and knowledge modeling, generative design systems, algorithmic design, performance assessment of buildings in use.