Computational Design, Parametric Modelling and Architectural Education

Şebnem Yalınay-Çinici, Yıldız Technical University
Fulya Özsel-Akipek, Yıldız Technical University
Tuğrul Yazar, Yıldız Technical University

Abstract

Architectural education in its relation with computational technologies is both becoming a part of these studies and having the potential of renovating itself with the knowledge of emerging technologies. This paper, in this framework will be presenting a design research studio that aims at developing the relational thinking capabilities in computational design process. In this studio, topics of parametric design, parametric modelling and relationality was questioned through design process.

In 2006-2007 spring semester in the Graduate Studio II course of Computational Design Program at Yildiz Technical University, parametric modelling software, *GenerativeComponents*, was introduced to question the pros and cons of parametric design in architecture. In this software there are no ready-made geometric primitives or shapes. Point is the only ready-made drawing element to use. A geometric shape is constructed by associating elements with each other. Each geometric association can be done through a number of commands with different parameters. The end-product is a parameter-based entity. Parameters can be changed easily, which in turn changes the whole geometric shape. In this design research studio held by 10 students, the topics of parametric design and associative thinking are questioned to explore their effect on the practices of architectural design and education.

Keywords

Computational design, parametric modelling, parametric design, associative thinking, architectural education

Başlık

Hesaplamalı Tasarım, Parametrik Modelleme ve Mimarlık Eğitimi

Özet

Mimarlık eğitimi hesaplamalı teknolojiler ile ilişkisinde hem süregiden tasarım araştırmaları sürecinin bir parçası olabilmekte hem de gelişmekte olan yeni teknolojilerle kendini her an yenileme potansiyelini taşıyabilmektedir. Bu bildiri, bu bağlamda ilişkisel düşünme pratiğinin oluşturulmasını hedef alan bir tasarım stüdyosunu anlatmaktadır. Bu araştırma stüdyosunda parametrik tasarım, parametrik modelleme ve ilişkisellik proje sürecinde tartışmaya açılmıştır.

2006-2007 bahar döneminde Yıldız Yeknik Üniversitesi Mimarlık Fakültesi, Bilgisayar Ortamında Mimarlık Lisansüstü Programı'nda yer alan YL Proje II lisansüstü stüdyosunda parametrik tasarım sürecinin deneyimlenmesi amacıyla parametrik modelleme yazılımı *GenerativeComponents* ile çalışılmıştır. Bu yazılımın özelliği geometrik kurgunun oluşturulmasında noktadan başlayarak her bir birimin birbiriyle ilişkisi üzerinden tariflenmesidir. Program hazır geometrik biçimler sunmaz. Her geometrik ilişkilendirme farklı parametrelerle oluşturulur. Sonuçta parametreleri değiştirilebilen ve her an yenilenebilen bir kurguya ulaşılır. 10 öğrenci ile yürütülen bu araştırma stüdyosunda parametrik tasarım ve ilişkisel düşünmenin mimari tasarım pratiklerine ve mimarlık eğitimine ne tür katkılar sağlayabileceği tartışmaya açılmaktadır.

Anahtar sözcükler

Hesaplamalı tasarım, parametrik modelleme, parametrik tasarım, ilişkisel düşünme, mimarlık eğitimi

Recent studies on computational technologies in architecture are leading a new understanding of academy and office collaboration which should be considered as a future condition that will effect the conventional routines of architectural education. It is a kind of understanding that reminds the necessity of a re-union and cooperation of both sides of the profession, that is academicians and office architects for a further understanding of architecture to develop.

One of the important reasons for this collaboration to develop is the recently emerging concept of parametric design in architecture. Though it is becoming one of the very common terms of computational thinking and design, its consequences in architectural education and design process are still under question. One of the early use of term for architecture was done by Mark Burry and Zolna Murray in 1997 at eCAADe-15. In this early introduction Burry and Murray (1997:1) equate parametric design with 'associative geometry'. The question of parametric design is equally a question of geometry according to them, since its introduction to architectural design process requires a different understanding of geometric ideation to develop.

At that point the initiation of *SmartGeometry Group (SGG)* gains importance, since the parametric modelling software -*GenerativeComponents (GC),* they introduced, is based on the understanding of associative geometry principles. Besides that the formation of *SGG* contributes to university and office collaboration because, though with educational purposes group has been initiated by the directors of architectural offices, KPF, Foster and engineering office Arup. They claimed that 3D modelling programs are not enough to enhance the design thinking capabilities and creativity in architecture. According to them further possibilities of geometric modelling and programming should be provided by a software to free architects from the 'constraints' of custom modelling programs. So, they started their own studies to develop that software, under the leadership of Robert Aish, the chief scientist of *Bentley Systems* then. Nowadays *GC* is having an increasing impact both on the architectural education curriculums and building industries.

So, for this paper *SmartGeometry* group and *GenerativeComponents* software they develop, are significant since there arises a dual effect from that movement to architecture. First, the understanding of parametric design, parametric modelling and associative geometry in architecture is enhanced by *GC,* which turned into significant topics in the design research agenda of architectural education circles. Second, the collaboration of university staff and practicing architects is strengthened to explore the impact of this software both in architectural education and practice.

In 2006-2007 Spring semester in the Graduate Studio II course of Computational Design Graduate Program, conducted by the authors of this paper and Asst.Prof.Dr. Togan Tong, at YTU to question the potentials and the deficiencies of parametric design, parametric modelling and associative thinking, *GC* was worked with. Two exemplary projects of this studio experience will be presented in this paper to discuss the effects of this emerging new situation for architectural thinking, design and education.

Parametric Modelling and Parametric Design

Parameter in its definition in *ScienceDictionary* is "a quantity or number on which some other quantity or number depends" [1] . So a parameter exists only in its relation with others. On the other hand in *Hyperdictionary,* parameter is defined as "any factor that defines a system and determines (or limits) its performance [2].

In the domain of architecture, since design is an issue of organizing and responding a set of relations to maintain and sustain the performance of a building; parameters are always essential constituents even if not defined as so. As mentioned above, parametric design in architecture can be equated with associative geometry, but it should also be noted that parametric design is essential in performance based design approaches (Kolarevic, 2005:149). The parameters such as sun altitude angles, climatic datas, structural limits or acoustic pre-requisites are also equally essential aspects of design, that can be computed in association to each other. Computational design sets the ground for various research growing on that field. We believe that the research on computational design will be contributing more, when it works for building stronger relations with the existing world. Hence the existing environmental conditions will be focused on more and their subsistence will be elevated by a logic that seeks the ways to appropriate with them in the form of buildings. In that sense parametric design is one of the research topics which should be considered in relation with the environment and the interrelations of its constituting factors. Consequently, parametric design will provide an understanding that builds the continuuum between design and production; which is embedded in the nature of associative or relational thinking. Nevertheless, it is still quite hard to define what parametric design is in architecture in spite of numerous publications spreading over.

Nonetheless one of the fundamental notions of parametric thinking is 'design of a family'. Javier Monedero (1997) defines 'family' as a set of elements that only differ in dimensions of their parts. According to his description there should be two things to define a family. One of them is "a topological description specifying the parts that constitute it and the relations they maintain with each other" and the second is "a dimensional scheme specifying priorities and dimensional constraints" (1997:3). This kind of an approach opens up the way for geometric research and motivates variation other than a rigid geometrical composition. Thus, parametric design breaks the single-object-oriented thinking habits and becomes a catalyst for the development of CAD/CAM technologies for which the 'non-standard mode of production' is the key issue (Cache, 1995). Non-standard elements are mass customized, but while belonging to the same design family, each of the objects can be produced in slight measurement differences by the use of parametric models which support multiplicity and variation.

Parametric design from the software side is the setting up of a parametric model through an understanding of associative geometry. Burry and Murray (1997:3) define a parametric model in which the constituting elements are referenced to each other using a number of clearly defined variables and constraints. Thus the completed model can be changed, modified and regenerated, while conforming to pre-set conditions. A parametric model can be 'updated' by changing the values of the parameters while keeping the relations. Custom CAD programs are also under renovation to develop a parametric capability that can enable the users to work with parameters of the standard geometric shapes, but since they still depend on the libraries of their ready-made geometries, they can offer limited skills for parametric set-ups.

GenerativeComponents on the other hand is a program that has been developed with an intention to provide an environment of parametric modelling. Another software enterprise to develop parametric design approach is *ParaCloud.* And it seems that software industry will be facing such endeavours in the coming future. These parametrically prospected software packages help designers develop associative geometric models that are subject to change according to the values users assign to the parameters they define for the geometric system.

With each 'experience' of its toolsets the potential of parametric design will be revealed which in turn will affect architectural notions that will open up further spatial topics to be discussed and explored in universities and offices. Thus, the future of architectural education will be facing different modes of thinking and design to deal with.

Computation, Parametric Modelling and Architectural Education

Computational design is an approach that operates mostly through the facilities of mathematical thinking due to the calculation skills of computers. It requires a mode of thinking, based on well-defined steps, algorithms and parameters, which necessitates a design strategy to be developed at the initial phase of design process.

Since the last decade, studies on computational design technologies began to deviate from the research done on computer aided drafting technologies; and are initiating that new form of 'design thinking'. "Computerized" simulation of traditional design methods are intentionally left behind to develop new digital tools that enhance computational thinking strategies (Terzidis, 2006: xi).

Parametric modelling is one of the recent tools of computational design technology. However, there are different levels of using computational capabilities in custom computer aided architectural design tools. According to this evaluation, lowest level for the use of computation is 'computerized design', in which computer algorithms are used only for drafting functions. No or only limited computational power is used in this most common utilization. Hence, custom CAD tools are examples of lowest algorithmic level in design, and the parametric modelling skills are not capable enough. High level of parametric modelling skill means using the computational power in a design process. This level requires the use of explicit definitions, algorithms and thus a dynamic design computation strategy. Today, there are academic studies on this issue, that question the use of scripting and programming as computational design technology and its pedagogical consequences. But it also brings the question of 'is designer becoming a coder?' as one of the most mind-busying concerns in the schools of architecture.

Thus, one of the most important aspects of computational design is "the increasing number of levels of indirection" introduced thereby according to Robert Aish (2005:11). These levels of indirection mean the introduction of "higher levels of expression and control". In his comparison between conventional design and computational design, Aish differentiates the latter design approach as an indirect one, where "the designer is not directly drawing the geometry, but has an indirect, and more arguably more powerful way of controlling that geometry using computational design tools" (2005: 11).

On the other hand, these levels of indirection can be seen as cognitive obstacles, since the traditional design education is based on direct hand-eye coordination. This leads us to think on design pedagogy, and its adaptation process to the contemporary approaches in design. Especially in the last decade, tool development became an important area for computational design researchers. These studies seem to seize an intermediate level of computation, where the geometric constructs of visual computation, are equally operative as script-based algorithmic operations. In the perceptual level, it offers an instant visual platform of design besides coding. We believe that, the intermediate level of computation, is an important attempt to ease computation by making it more 'user-friendly' even in its most highest level of use. *GenerativeComponents* is one of these developments, creating an important in-between among these two levels of both computerized and algorithmic use. Thus, it provides an intermediate level of computational use. So there arises an answer to the question 'designer or coder?' in terms of which designer can also code algorithms in a visual and symbolic way without losing control over the design object. Therefore, *GC* comprises the advantages of both computerized design and computational design. We think that this level is especially important for educational purposes.

SmartGeometry Group (SGG) **and** ***GC***

To understand the geometric and computational accomplishment achieved by *GC,* a brief overview on how the idea of such a software developed will be helpful. The *SGG* was initiated by the three directors: Lars Hesselgren from KPF, Hugh Whitehead from Foster and Partners, and Jay Parrish from Arup [3]. The group was started with educational purposes to enhance "the level of 3D modelling skills in the architectural industry" [3]. They collaborated with Robert Aish to realize that. The main impetus for this group movement was the need of an advanced computational design understanding in the schools of architecture and the lack of capable and progressive software packages, that respond to the dynamic and non-linear process of design. So they began with the development of such a software, former *Custom-Objects* later *GenerativeComponents*, where they aimed at fusing "geometric modelling and programming" [3].

The main concern of the group is the problem of modelling in the process of architectural design. The logic of 3D modelling programs in use, which are working with precise dimensions and pure forms, are not capable of developing architectural thinking and design creativity according to them. Hesselgren mentions that "the research is aimed at liberating architects from the constraints imposed by the CAD industry. Most modelling tools are 'prescriptive' in their use…and make it difficult for architects to work with pure complex geometries. The industry is trying to define for architects what architecture actually is. What we want to build are geometric relationships which are not dependent on notions of architecture, but are only dependent on geometric assumptions." [3] Whereas, according to Aish, "at the conceptual phase of design, architecture is much more about geometric relationships and proportions than concerned with precise dimensions. *Custom-objects* allows the creative designer to capture these relationships not as a single static model, but as a live and re-executable configuration . . . as a series of geometric what-ifs?"[3]. So, according to their point of view "architecture is fundamentally about relationships" and "many of those relationships are geometric in nature or find a geometric expression", hence *SGG* "has been created in the belief that Computer Aided Design lends itself to capturing the geometric relationships that form the foundation of architecture"[6].

Concepts of associative geometry and parametric modelling are the tools to operate with to fulfill that purpose in design. Because in parametric modelling, associations are key to construct the geometric relations in a design system. Thus *GC* does not offer any ready-made geometries or shape libraries. Point is the only ready-to-use drawing element. There are numerous commands for creating different geometric shapes with different parameter sets. Each element is created in association to the previous element(s), all depending on user-defined parameters. So any change of parameter in one of the associated elements is reflected in the whole system. To trace the operations, *GC* works with three types of windows: command, symbolic view and model view. Command window is for picking the commands that appropriate most with the geometric and design system associations. In 'symbolic view' window user can see the associated relations (Fig.1). Since building geometries through relations is the main concern in *GC*, tracing it in 'symbolic view' is indispensable to have control on the whole system. Other graphic windows are for the different views of parametric model.

GC uses scripting as the interface, but in *GC* it is not a pre-requisite to have the knowledge of coding and scripting. Though the user creates the script file while the modelling, s/he does not need to know scripting. In *GC* there is an instant interaction between the two different modes of work, e.i. graphical and text or script-based. Any change done in the visual model is instantly reflected in the script file. And any use of commands is added to the scripting file in its coded form. If the user has the knowledge of scripting, s/he can easily access the script file and continue work with coding. Any change in script file is also reflected instantly on the model.

Hence, for educational purposes *GC* provides an intermediate level of use, which helps students to develop a high level of computational logic. Since, it accommodates both a script-based understanding of computation and a capable graphic medium to proceed with design as a problem of associative system construction.

Studio Work

In 2007 spring semester in the Graduate Studio II of Computational Design Graduate program at YTU, parametric design was the topic of research to explore its advantages and deficiencies. *GC* was in its beta version at the time being. It was introduced by Onur Yüce Gün (KPF) in a workshop held at METU in 2006, hence its objectives in enhancing computational capabilities in a design process were informed beforehand and decided to be questioned in education.

As a pre-knowledge, it should be noted that the logic and the tools of operation *GC* offers, were not easy to adapt by the students. Since custom 3D modelling programs do mostly operate with ready made geometric shapes and offer ways of easier 3D modelling, students had a difficulty of shifting their mental habits of using a CAD program. When conceived by an object-oriented 3D modelling mentality, *GC* is a program, which is very hard to move with. However, when one begins to shift the design mentality to a computational logic of dynamic system modelling on variable data, *GC* provides rare possibilities. When approached like that, an architectural problem turns into a design of relations through defined variables and constraints. So it is not about using a program, it is principally about shifting the mentality in design. Hence, for this first shift in design understanding to be realized, students were asked to define possible parametric architectural design problems in an environmental setting. It should be mentioned that there aroused various approaches at that stage, which ended up as different projects. Nevertheless, final projects can be categorized under two topics: responsive structures and data-dependent transformable constructions.

For the first category Caner Kutsal's and for the second category E. Sedat Özdemir's projects will be illustrated here as exploratory examples. To begin with, Caner's proposal was a responsive structure, a bus stop which extends its roof and seats when someone comes closer, and turns into a flat surface when no one is around (Fig.4). He approached to problem from a dynamic geometric understanding and design. Caner constructed a set of points in line with each other as control points for the bspline curve sections of its surface (Fig.2). These bsplines are lofted to have the surface of the bus stop (Fig.3). Then he placed the points that represent people coming closer to the bus stop. The point-set he created at the first stage are associated with these free moving points by an if-then-else coordinate defining algorithm in the model. Thus according to the position of the person approaching, the coordinates of the related points are re-defined to extend the roof and seat structures of the bus-stop (Fig.4). Finally he modelled a dynamic geometric system through the variables and constraints he set.

Projects in the second category are also designed as generic systems, but not as real-time responding structures. They are to be transformed according to the rules set by the designers. Sedat's project in this category is a beam system that can be shaped according to the section requirements or span conditions of the building. Control points for section curves are set first, and then they are located in association with each other both in X, Y and Z axis (Fig.5). So when a coordinate of a control point on a section curve is changed, whole system changes. This provides a parametric structural system. It is not a fixed structure, but is open to change of data in relation to site, topography, climatic conditions, functional requirements, structural constraints and preferences etc. (Fig.6). Sedat's project is also an example of how the understanding of associative geometry enhances design understanding.

Thus, the projects produced in the studio also validate *SGG*'s concern with geometry and the understanding of associative geometry to develop for the advancement of computational thinking in architecture; since the capability gained in associative geometry as an 'indirect' control is observed as a gained level of control both over design and computation. And from an educational view, this helped the students to develop higher levels of computational skills. Because, besides enhancing the associative thinking capability, the use of program forced students to develop algorithmic skills.

Conclusion

This studio experience on parametric design caused the initiation of ideas for what the future of architectural education in computational use can be. The capability of parametric modelling not only enhances the higher levels of computational use but also appropriates with the non-linear process of design. Thus, design process as organizing relations becomes an important topic for education. Because the skill in setting relations and designing data-interacting-systems require a new mode of design thinking to be developed by education.

Besides that, custom CAD programs are setting another dimension of 'cognitive obstacle' in software use. Ready-made geometric shapes offered by these custom programs though ease drawing and modelling procedures, in fact do block the ways for relational thinking. Thus, *SGG*'s attempt to free architects from the 'constraints' of these programs is valuable in educational prospects since it opens up a new understanding of "architectural geometry" (Pottmann, 2007). Though seems hard to use, *GC* in fact liberates architects in design ideation through the introduction of that new geometric understanding.

There also arises another effect in relation to practice, construction and manufacturing. Besides universities, the consequences of this software are being searched for in professional project processes also (Hight and Perry, 2006). Thus studies on computational thinking in architecture are not limited to the problems of design ideation only. Since in current computational design debates how one designs also involves the question of how one manufactures, parametric modelling works as a catalyst to enhance that emerging architectural condition. And the 'experiential' studies on the possibilities offered by computational technologies both in CAD and CAM industries are gaining importance. Because the possibilities offered by computational technologies need to have a dimension of experiencing since its production is an indispensable requirement for these studies to be fulfilled. Studies on architectural thinking and computational design have to extend into the conditions of construction and manufacturing to demonstrate their actuality. It is that very naturally evolving condition, which began to bring the academy and office interest into the same pot.

Thus, according to this emerging condition, for the future of architectural education it can be claimed that schools of architecture will not be only for breeding architectural ideas but will be turning into concrete sites of research for advanced architectural construction and fabrication technologies also. This condition has the potential to change the so long accepted picture of academy as the serene world of architectural ideation. The rooted belief on academy as the glasshouse of 'theoretical' expertise, detached from the concrete realities of 'practice', seems to be under severe transformation through the steady-fast developing studies on computational technologies.

Acknowledgement

Authors of this arcticle would like to thank Onur Yüce Gün (KPF) who made an early introduction of the GC software to the students in Turkey, by the workshop held on 26-28 March 2006, in the Digital Design Studio, Faculty of Architecture at METU. He also contributed to the Graduate Studio II at YTU by video conferencing and e-mails. We also would like to thank Bentley Systems for providing the GC software for this work to be realized.

References

Aish, Robert (2005), "From Intuition to Precision", *eCAADe-23 Conference Proceedings, Digital Design: The Quest for New Paradigms*, Eduardo Costa (ed.), Portugal, p.10-14.

Burry, Mark and Murray, Zolna (1997), "Computer AIDED Architectural Design Using Parametric Variation And Associative Geometry", *eCAADe-15 Conference Proceedings*, Vienna University of Technology, Bob Martens, Helena Linzer, Andreas Voigt (eds.), Österreichischer Kunst-und Kulturverlag, Vienna. (http://info.tuwien.ac.at/ecaade/proc/burry/default.htm) *Accessed on April 14, 2008.*

Cache, Bernard (1995), *Earth Moves: The Furnishing of Territories*, MIT Press, Cambridge.

Hight, Christopher and Perry, Chris (eds.) (2006), *AD: Collective Intelligence in Design*, John Wiley & Sons, West Sussex.

Kolarevic, Branko (ed.) (2005) , *Architecture in the Digital Age: Design and Manufacturing*, Taylor & Francis, New York.

Monedero, Javier (1997), "Parametric design: A review and some experiences", *eCAADe-15 Conference Proceedings*, Vienna. (http://info.tuwien.ac.at/ecaade/proc/moneder/moneder.htm). *Accessed on April 14, 2008.*

Pottmann, Helmut et.al. (2007), *Architectural Geometry*, Bentley Institute Press, Pennsylvania.

Rotheroe, Kevin (2002), "A Vision for Parametric Design", *Architecture Week*, (http://www.architectureweek.com/2002/0710/tools_1-1.html). *Accessed on April 14, 2008.*

Silver, Mike (ed.) (2006), *AD: Programming Cultures*, Artmedia Press, London.

Terzidis, Kostas (2006), *Algorithmic Architecture*, Elsevier and Architectural Press, Oxford.

e-Sources

[1] http://www.answers.com/parameter?cat=technology. *Accessed on April 14, 2008.*

[2] http://www.hyperdictionary.com/search.aspx?define=parameter . *Accessed on April 14, 2008.*

[3] http://www.architectureweek.com/2001/0919/tools_1-2.html . *Accessed on April 14, 2008.*

[4] http://ww.smartgeometry.com/bd.htm. *Accessed on April 14, 2008.*

[5] http://www.smartgeometry.com/about.htm. *Accessed on April 14, 2008.*

[6] http://www.smartgeometry.com. *Accessed on April 14, 2008.*

FIGURES

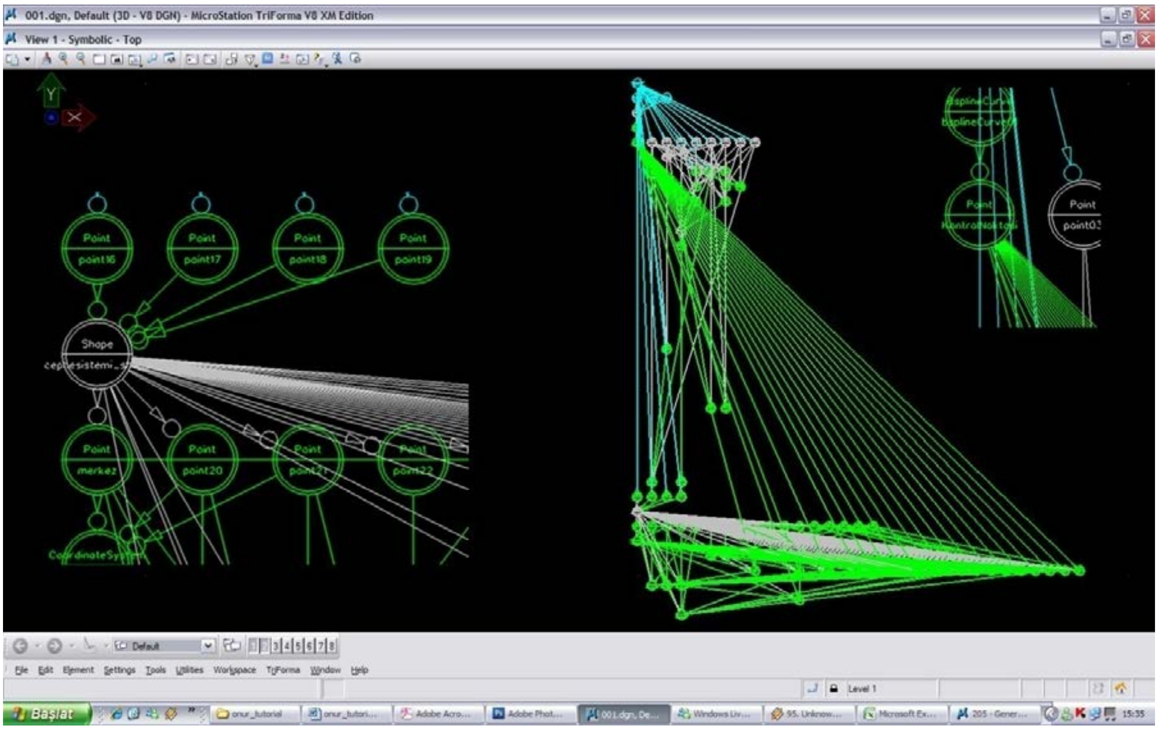


Figure 1: Symboic View of a Model

Responsive Model of a Bus-Station Design / Caner Kutsal

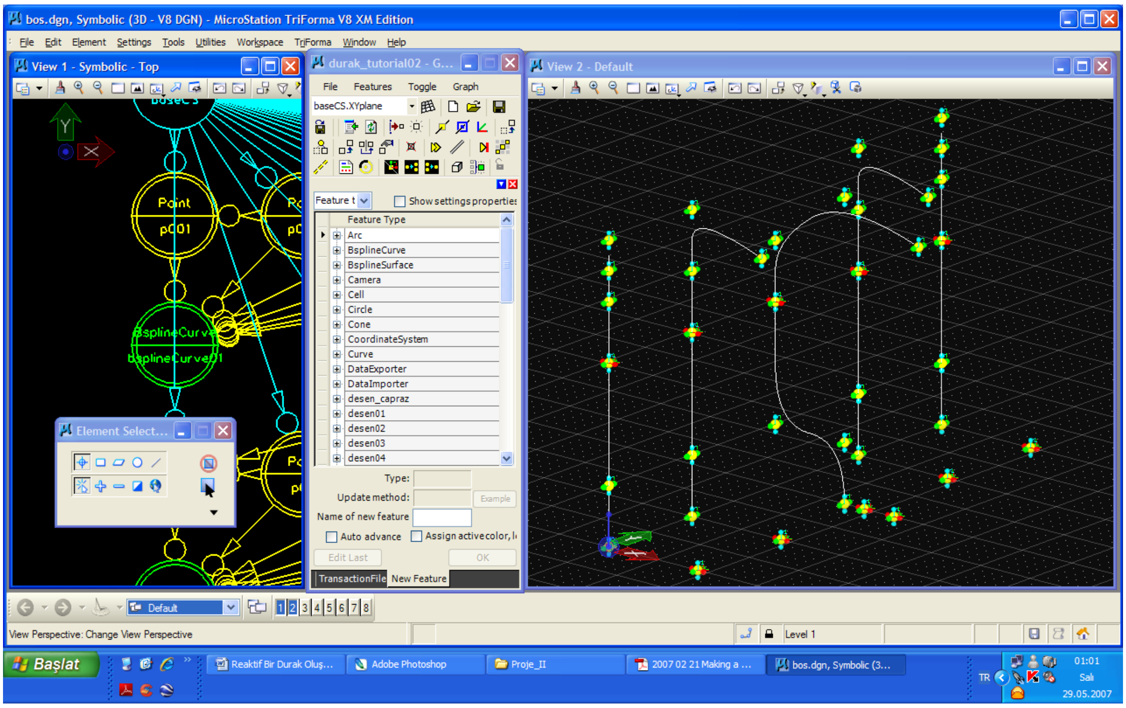


Figure 2: Geometric construction of the model

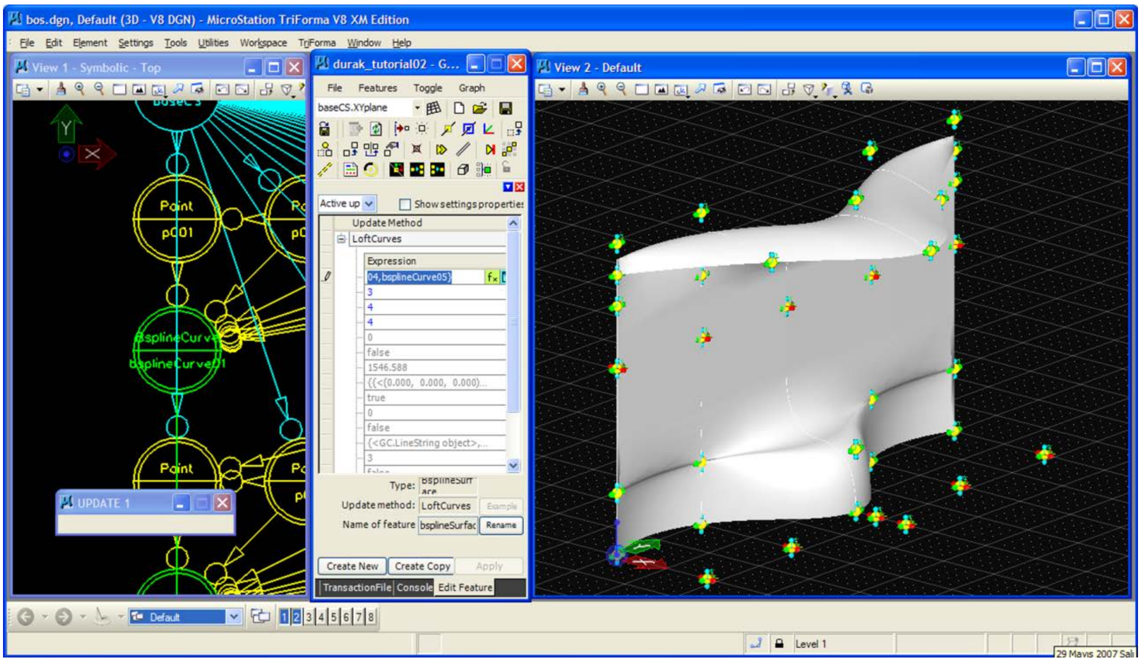


Figure 3: Symbolic view and parametric model

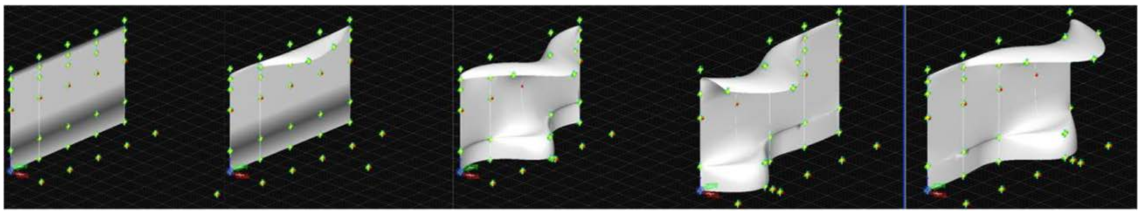
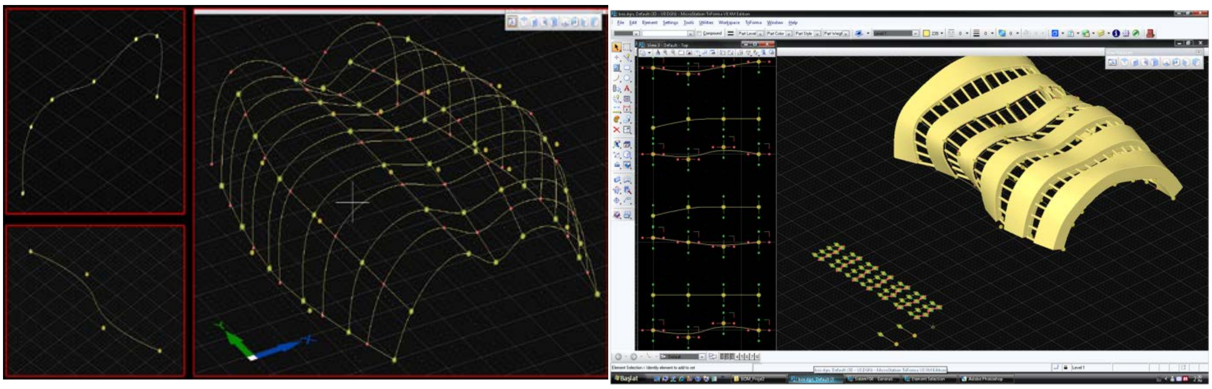


Figure 4: Parametric variations

Parametric Model of a Long-Span Structure System / Engin Sedat Özdemir



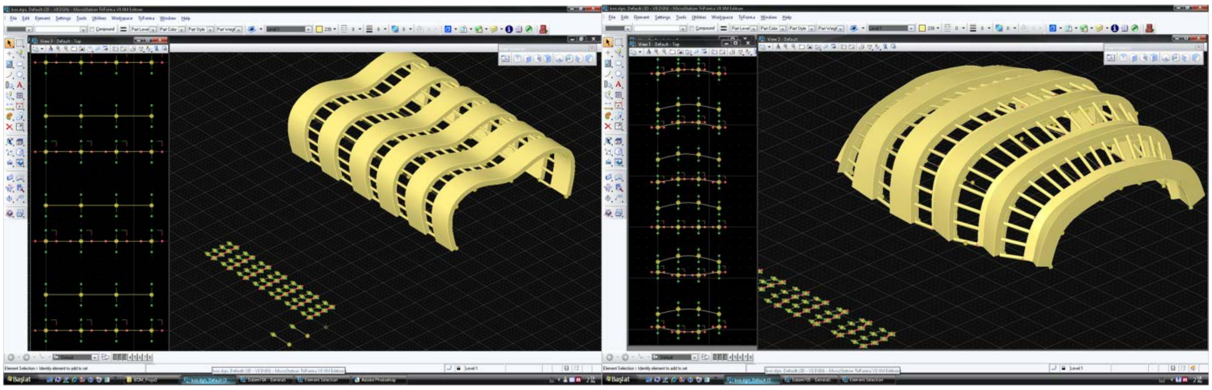


Figure 5: Parametric model of the long-span structure system

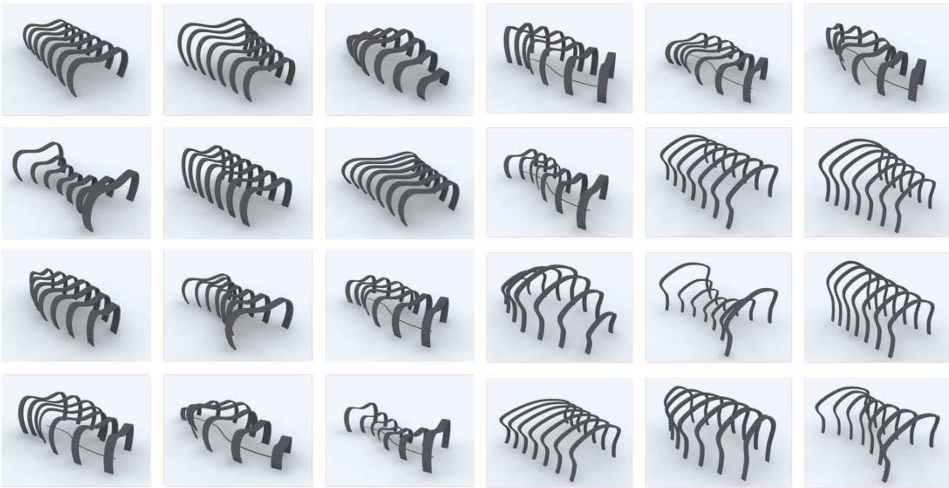


Figure 6: Parametric variations of the model structure