Composing the Bits of Surfaces in Architectural Practice

Methodologies and Codes for Generation, Rationalization, and Analysis of Non-Standard Geometries

Onur Yüce Gün
Kohn Pedersen Fox Associates PC New York, USA
http://www.kpf.com (company webpage)
http://algotecture.com (personal webpage)
ogun@kpf.com

Emergent design tools; with enhanced modeling and parametric manipulation capabilities are encouraging the exploration of new geometric typologies in the field of architecture. Designers are not only finding more opportunities to work with geometries of higher complexities but also becoming able to update their designs with simple formulations. After a decade of proximity with free form modeling tools, architects now have to become more aware of the critical relationship of design and construction. When the design is performed without taking the constraints of the construction the inefficient method of geometric post-rationalization unavoidably has to take place. So, the knowledge of the rationale should be applied from the very beginning of the design processes, and the digital models should be informed and controlled while being developed.

This paper will present analytical strategies and methods developed for working with non-standard geometries in a geometrically and parametrically controlled environment. Each method is supported with custom scripts which run in both parametric and non-parametric computer aided design (CAD) platforms. Each script and method is manipulated for the next project over time and the computational tools created build up a library of surface generation, manipulation and subdivision tools.

Keywords: Parametric; surface; construction; Generative Components; Rhino Script.
Introduction

Early design exploration happens with large gestures of sketching or modeling. The smaller pieces and components that build the global form are designed and defined later in the design process. However the act of construction logically goes vice versa, from piece to the whole. The discrete pieces put together in construction make the building.

While represented and perceived as smooth continuous surfaces in CAD platforms, the constructed building envelopes are composed of numerous pieces differentiating in size, shape and material. A design process that incorporates all the ideas related to the discrete building blocks of the building would enable the creation of well-informed design solutions. Thus, the whole should be designed in awareness of the piece, in other words, the piece and the whole should be simultaneously designed in the design process.

Approach

The term computation has widely been used within the architectural terminology in recent years. Growing interest in tool making encouraged computational and analytical approaches in design. The digital tools with no doubt have influenced the designed forms, but now we’re in a period in which the tools are also forcing architects to deal with more in-depth analytical design thinking and approach.

While a virtually unlimited variety of geometries can be generated with emergent computer aided design tools, they have to be evaluated and optimized with a better understanding of geometrical dependencies and limitations. The complexities we as designers are dealing with should be the derivative of the constraints of the concepts, site inputs and limitations of construction, rather than uncontrolled and unintended formal complexities.

Tools and methods

In this paper, I will present several non-standard geometry development and subdivision tools that run on different parametric modeling and scripting environments, and their evolvement by time. I will present, compare and contrast the different approaches in creation and use of these tools in different phases of design. I will refer to various large scale projects of Kohn Pedersen Fox Associates New York, including the 680 meter-tall Z3 tower in Shanghai and quarter square kilometer shelter Nanjing Train Station.

Case studies

In the following chapters, first I will present the processes performed with a blend of tools created with Rhino Script, Generative Components (GC) and GC Script, which were meant to resolve the geometrical complexities of three different towers. In these projects, the developed tools intended to rationalize and reconstruct the overall form, which are later subdivided into building components regarding the ease of construction and cost. The extension of the tool makes automatic preparation of construction documentation possible, via printing the stack joint location data for each component.

Then, I will cover the tool chain created for the design and manipulation of the roof of the Nanjing Train Station. I will present and compare the different tools developed for different phases of the specific design process, and how creation of these tools was driven by the required actions of design. While the first tool aims the generation of the complex form in a controlled environment, the second one subdivides the surface in mock-up detail for data transfer for scale models. I will go into detailed geometric and algorithmic explanations for each project.

Reverse-engineering the geometric dependencies

The main form of CSCES tower is generated by three sheared cones, whose bases are centralized on the corners of an underlying triangle (fig. 1). Unlike conventional tower forms, the tower gets wider towards top. A surface wrapping all three sheared cones creates the envelope of the building. A parametric
model of the tower was developed by Kyle Steinfeld, who also developed the initial methodology of panelization of the building.

The tower is panelized by both warped and flat panels depending on the curvature of the building surface. To succeed in creating the desired visual qualities, same number of panels is used in each floor of the building. Thus, since the length of the perimeter line for each slab is increasing from bottom to top, the width of the panels has to increase by 4 mm in each floor. The envelope of the towers was digitally panelized and the stack joint point data was written to spreadsheets and sent to construction site.

My role in the project started while the building was already under construction. Due to the misreading of the construction documents, the concrete slabs were constructed with an offset, which required an update for the stack joint data for each panel of the tower. Only input for the update was a model with current curtain wall point data, which is a point cloud, and the slab perimeter lines.

I first developed a Rhino Script, which sorted and grouped all the points as specific stack joints of each panel in each floor. To update the point data according to the new construction each point had to be translated on an axis, that axis being the projection of the normal of the surface to a horizontal plane at that specific point. While this guideline was fairly easy to determine in the flat portions of the building surface as a perpendicular to the floor slab perimeter line, the points on the curved areas were analyzed according to the central spine of the three sheared cones (fig. 2).

After determining all the translation axes the points were moved on these axes depending on the offset data sent from the site. This represented the new lower corner stack point for panels. All offset points were then used to define the top stack joint for each panel. Here I would like to highlight the complexities that are embedded in such a fairly simple geometrical construction: Since each panel has a different orientation, the top points of the panels also have to be determined by local geometrical properties. To do so, the local translation axis lines were translated to the upper floors and extended centrically to hit the slab perimeter line. The intersection gives the necessary top stack joint point data.

The four points representing the four stack joint locations for each panel are computed after creation of set of guide geometries. The output is written to spreadsheets to be used in construction.

This study can be defined as reverse engineering, in which I worked on finding the origins of the developed geometry. While not presenting explorative efforts in design, these techniques and methods created a knowledge base to be applied to the following processes.
Geometric reconstruction and rationalization

White Magnolia Tower is a 65 storey free-from building. In the original design, which was modeled by non-uniform rational b-spline (NURBS) modeling techniques, three double curved surfaces wrap the tower. The configuration of the three continuous surfaces intends to imitate the petals of the white magnolia flower. The form represents deviating curvature degrees (fig. 3) whereas the limited budget required use of flat panels instead of warped ones, which required a geometric rationalization process.

The reconstruction studies of White Magnolia Tower developed around the idea of generation and use of parametrically controlled torus patches, since a torus or a torus patch, which is a cutout from the surface of a torus, can be built by use of flat panels (fig. 4).

A set of parametrically controlled circles, which are constructed with tangential dependencies are used to produce a layout for arctangents, which are later combined into a composite curve and define the slab perimeter of the building (fig. 5). The composite curves representing the slab perimeter lines are then trimmed from both ends with trimming lines that rotate a certain amount in successive floors. This creates a twisted cut, while the complexity of the surface remains the same.

The composite slab perimeter curve is scaled while carried along a vertical arc. The manipulation of both the vertical arc and the composite slab perimeter curve defines the overall form of the building, controlling the amount of tapering and the maximum width in the middle of the building. The resulting geometry is rather complex since it’s composed of three different torus patches. However, the transitions between these patches are perfectly smooth since the composite curve was initially developed with tangential transitions of underlying circles (fig. 3).

Each of these processes was developed in Bentley’s Generative Components. The model I generated is driven by both global variables, which are numeric
inputs; and associative dependencies between the underlying geometries, which dynamically update in connection to any change in a geometric member. This parametric model was used to generate variations of the tower (fig. 6).

Generated geometries were evaluated in the means of ease and cost of construction and the proximity of the final form to the initial one. Three dimensional prints were used to compare and contrast the different visual qualities (fig. 7).

In the next phase of the study, I developed a Rhino Script, which enabled me to automate the tower panelization. The panel placement works as follows: The start point of the slab perimeter line acts as a center of a circle, whose radius is equal to desired panel width. Once drawn, this circle intersects

Figure 6
Various towers as a product of parametric model

Figure 7
3D prints of tower (02) variations
the slab perimeter line at a point, which determines the second point for the panel. The next panel is created by using this second point of the first panel. So it becomes the center or the second intersecting circle, which will determine the second point of the second panel. And the routine keeps creating the panels until it reaches to the end of the slab floor perimeter line, and then next floor is processed (fig. 8). The other two points for the panels hitting the lower floor slab were determined with similar techniques used in CSCEC tower.

Once all the panels are created, they are classified depending on size and color coded for a quick representation of the number of panel types. While determining the groups, a negligible tolerance of 10 mm is used. Ultimately with this methodology the tower can be panelized by using six different types (fig. 9).

**Cost-effective panelization**

Z3 tower is conceived with a diagrid structure and represents similar geometric qualities to White Magnolia tower. In the parametric model we developed, a subdivision factor over surface of the building updates the nodes for the diagrid, and the manipulation of the factor enables producing various densities (fig. 10).

The underlying diagrid structure implies use of parallelogram (diamond) shaped panels, which was also the original design intent. However, a quick study we made by conventional 3D modeling techniques in rhino revealed the inefficiency of use parallelogram shaped panels. A secondary study with trapezoidal panels revealed similar problems. The final conclusion for Z3 tower was using shingled panels, which would reduce the cost of panelization by enabling us to use panels of same size on most of the portions of the building.

In a shingle system, the consecutive two panels overlap a certain amount. A smaller panel, perpendicular to both panels, fills the gap between them (fig. 11). So the variation in size can be embedded to these smaller filling panels with customized fixtures. Thus, the main panels in a floor can be of same size.

Due to performance considerations and the linear logic of the panelization routine, I preferred to use Rhino Script. The scripted routine lays the panels in a similar process of Magnolia tower and then rotate them a certain amount around the vertical axis where they touch the slab edge. This angle can vary and is open to manipulation depending on some
external parameters or mathematical drivers. Further investigation can be explored with environmental and computational fluid dynamics analyses, thus the shingle angle can be manipulated either by solar orientation, or according to the prevailing winds. The panel surfaces are then rotated in the horizontal axis, and extended to create the shingle effect above the lower panes (fig. 12). This vertical shingling helps in rain water drainage by keeping most of the rainwater out of the draining system and simply letting it go down the surface of the building.

**Invisible components**

Learning from the three towers, and the motivations those studies created in the office enabled me and my colleagues to collaborate on the early design exploration studies of Nanjing Train Station. Thus, it represents a different character and work flow.

At the very beginning of the design process, we received a conceptual non-parametric CAD model that was revealing the design intentions of the train station. The 500 meter long platforms lying between
the fifteen train-tracks aligned on east-west axis are sheltered with large canopies to act as a rain screen, while the tracks themselves are left open to let sunlight in. The canopies begin to raise and respond to the north and south in the middle of the station which act as entrance and exits. While the conceptual approach intends to create an organic and fluid form, the limits of the regular modeling techniques didn’t reveal those intentions.

With references from the non-parametric CAD model, a basic parametric model was built in Generative Components to explore different topological organizations (figure 13). Unlike the initial model, the canopies were designed and modeled as responsive parts of the whole, thus any manipulation on the global form dynamically updates the form of each canopy. With certain parameters we generated several models and analyzed and evaluated their environmental performance and visual qualities (figure 14).

In Generative Components, the canopies were designed with an “S” section, which enabled us to control the reflection and penetration of sunrays, as well as the water drainage. The surfaces were...
generated as a derivative of various configurations of discrete “S” curves, which were parametrically controlled. While driving the design, these curves, as components remain invisible. The layout and organization of the “S” curves were defined with global rule-sets, and the final design configuration (figure 15) was a result of these rule-sets, rather than a “hand-crafted” geometry.

The behavior and ranges of adaptation for the “S” curves were defined before-hand, thus the geometry was already being developed under certain constraints. In other words, the design was informed and restricted by certain limitations, thus it was pre-rationalized with embedded intelligence in the parametric model. The underlying structural scheme was developed simultaneously with the surface generation, in collaboration with structural engineers.

A 1/400 scale model was built in China by using digitally transferred information. This was a rehearsal of a possible construction since all structural members and surface pieces were created and prepared for prototyping.

Conclusion

The projects I covered in this paper demonstrate different strategies developed for subdividing complex-geometry surfaces. Pointing out the problems of geometric post-rationalization, this paper highlights the importance of incorporating the information related to construction limitations into design processes.

This paper presents parametric modeling not as a solution tool for design but as a new area of inquiry for it, since every design problem now demands custom approaches, analyses and applications in the parametric modeling environments.

In the contemporary course, architecture stands as a product of hybrid processes, in which traditional analogue and developing digital design methods merge with methods developed by coding. Giving references to real large scale projects, this paper represents how concepts of architectural design can overlap with principles of scripting and how design processes can be supported with digitally generated tools.

Today, an architect capable of customizing one’s tools in digital means would be able to express and realize one’s designs intentions more solidly in the digital environment. A complete set of both analog and digital skills would put the architect to the position of the “master builder” as a “digital craftsmen”, in which all the aspects of design are determined by the designing hand and mind.

Acknowledgements

I would like to send all my gratitude to James Brogan for his support in my studies.

I would like to thank Nicholas J. Wallin, Stylianos Dritsas, Mirco Becker and Kyle Steinfeld for their collaboration in parts of the studies.

Thanks to Lars Hesselgren, Robert Whitlock, David Malott and Aman Krishan for their interest in my ongoing work.

References


Stylianos Dritsas, Renos Charitou, Lars Hesselgren: 2006, Computational Methods on Tall Buildings.


**Image credits**

All images are by the author unless otherwise noted.