Designing a Sunshade Installation for the UT Zero Energy House: An Exploration in Generative Modeling Technology

Edgar Stach\(^1\), AIA/IA (Associate Professor of Architecture)  
Casey Arthur\(^2\), CDT (Student, Bachelor of Architecture)

College of Architecture & Design  
The University of Tennessee, Knoxville  
stach@utk.edu \(^1\), carthur@utk.edu \(^2\)

Abstract
The subjective quality of architectural design requires all designers to consider an infinite set of possibilities to a project design. Logically, the faster a designer can visualize and communicate possible ideas within a given time frame, the better their opportunity to discover the best design solution. Ultimately, perfection in architectural design can never be achieved, but its pursuit does lead to a more refined solution.

Computer-aided drafting and modeling has provided designers more efficient visualization of project designs, increased productivity, and enhanced workflow; however, with the increasing complexity of contemporary designs, new generative modeling technology must be employed to sustain efficient productivity in the design process. Generative modeling is a computer software technology which allows the designer to provide a set of parameters in which a programming script can be written to generate elements within a given domain. This technology allows the designer to model highly customizable and complex elements which can be generated at the speed of a computer calculation.

What are some benefits of generative modeling technology, and does it support practicing sustainable design? This paper will discuss the implementation of Grasshopper (revised for version 0.8.0003), a generative modeling plug-in for the NURBS modeling program Rhinoceros (version 4.0 SR8), and its role in creating a sunshade installation by a research design team for the zero energy house project at the University of Tennessee. More importantly, the instructive description of the design and fabrication process will show that the same steps can be adapted to other architectural projects. Then, the paper will conclude with a brief case study, the CBD Media Tower, and its use of Grasshopper to provide enhanced modeling and visualization techniques.

The Design Process
Preliminary Design
The preliminary design process was used to determine objectives for creating the sunshade. The goals were the following: create a module which filters sunlight and could be easily folded in to shape from a single piece of a specified construction material, use a generative modeling tool to array the module over a rectangular surface which could be framed and mounted to a window, give a sculptural quality to the installation by warping the modular array across defined curves using the generative modeling tool, and successfully export the generated model for fabrication and manufacturing to complete construction of the sunshade.

The Design
In the design process, the team began by first developing a module. Initial research with paper products and scissors led to a choice in using a rectilinear volume and then subtracting shapes from it to let it filter light. At this point, it was important to consider the connection between the module pieces. Ideally, all pieces would be connected for economic efficiency; however, due to area limitations of the cutting equipment, it was determined two module pieces would be connected at top and bottom for cutting. Then, the pairs would be organized into a series of columns.

The second part of the design process was to finalize the exact shape and dimensions of the module, and then develop the algorithm for the generating the digital model. Being familiar with Rhinoceros and the Grasshopper plug-in, each team member investigated different module designs, developed generative algorithms, and inserted their results into a digital model of the zero energy house prototype for visualization. Each member also investigated various materials which could be used for construction of the module pieces. This process was repeated until a design solution was agreed upon by the design team.

The design solution for the sunshade installation module was a three-by-three inch rectilinear volume with a 1.5 inch thickness. The module was then designed to have curvilinear and triangular shapes subtracted from it to filter light and connect to its paired module. A nine foot seven inch by three foot ten inch wood frame was designed to use two-by-two inch lumber and then mounted to a
corresponding window. Then, 37 module elements would comprise each column, and 14 columns would comprise the array within the frame. The specified construction material for the modules was a synthetic triple-layer polypropylene paper manufactured by Yuko. The generative algorithm used was a modified box morphing script. Adapted from a box morphing script found in video tutorials by Jonas Bredel on the Grasshopper plug-in website (Bredel 2009), the designed algorithm was chosen because of its capability to give the installation its sculptural qualities. Unlike the others tested, it was able to generate the modular array across defined curves by allowing for division of the array. The algorithm could be used for each division of the array and then generate the divided array as a single model.

The Fabrication & Construction Process

Rhinoceros Modeling
The fabrication process for the sunshade installation began with digital fabrication in Rhinoceros. Although the initial steps of digital fabrication were performed in the design process for visualization purposes, it is important to explain the steps in detail so that it is evident that such steps can be adapted to a variety of architectural projects.

Initially, one must have a reasonable understanding of the Rhinoceros modeling software. Rhinoceros is a NURBS (non-uniform rational basis spline) modeling program. Essentially, a user defines points which subsequently generate points, lines, planes, or volumes. Commands also integrate many functions for extending modeling capabilities such as generating planes from existing lines and joining elements. Before work was started, input settings were set to the proper dimensional units; otherwise, modeling could be inaccurate. The first task was creating the module. There were a variety of ways to do this; however, the method used in this project was simply to draw lines according to the design dimensions, generate surfaces from the edge lines using the “EdgeSrf” command (Edge Surface), and finally use the “Join” command (Join) to join the surfaces as one unified piece. It was important to make sure the surfaces were joined in the proper sequence; otherwise, they would not unfold properly. In this project, it started with the top triangular surface which had a connection point to the above module piece in the same column, and then it ended with the right side rectangular surface. This was checked by using the “UnrollSrf” command (Unroll Surface). If individual surfaces were seen intersecting, or they did not unfold as desired from earlier research, then the sequence in which the module surfaces were joined was revised (see Figure 1).

Once the module was complete and observed to unfold properly, a surface plane was created so the Grasshopper plug-in could generate the modular array from its defined domain. First, lines were drawn to define the surface edges. These were dimensioned from the inner edge of the sunshade frame. Next, spline curves were drawn to provide the sculptural quality. In this project the “Curve” command (Curve) was used to draw two spline curves which ran from top to bottom. Careful consideration was taken to drawing the spline curves. Depending on the design of the module, if the slopes of the spline curves change too hastily, meaning the curve is too “curvy,” then there may be problems unfolding modules generated near the critical points. This can be tested and revised if necessary once the modular array is generated. After the spline curves were drawn, the top and bottom lines of the boundary edges were divided according to each line segment’s corresponding space. This allowed the surface segments to generate appropriately. In this project, the correct number of bounding lines was four vertical lines and six horizontal lines for the three surface segment spaces (see Figure 2).

The “NetworkSrf” command (Network Surface) was used to generate the surface segments. When creating each surface segment, make sure only line elements are selected to generate each segment. When the surface network dialogue appeared, each line was verified to use the same designated label. For example, if the top line of the surface segment network is labeled “A,” then the top line for other surface network segments should also be labeled “A.” This
and any modifications in the network surface dialogue could lead to errors in generating the modular array. Once complete, the surfaces segments created a single surface plane for the modular array. These segments were moved away from the lines used to generate them for easier use when generating the array (see Figure 3).

Next, the algorithm was created in the Grasshopper plug-in interface. Once Grasshopper was successfully installed, the “Grasshopper” command (Grasshopper) could be typed to open the interface. Grasshopper is a visual algorithm editor, and this allows the user to not need any experience in programming or scripting (Grasshopper 2011). The user simply inserts necessary algorithm elements into the workspace and connects them appropriately to generate a desired model.

A simple algorithm, the one for this project used the basic elements: parameters and components (Payne and Issa 2009). To begin, the geometry section was first configured. This section required the “Geo” (Geometry) parameter, the “Slider” (Number) parameter, the “BBox” (Bounding Box) component, and the “Morph” (Box Morph) component. The Geometry parameter was set to the designed module, and then connected to the “C” input of the Bounding Box component. This assigned the geometry to be contained by the Bounding Box component as it solved the oriented geometry for a bounding box. The Number parameter was set to 1.5 and connected to the “U” (Boolean) input of the Bounding Box component. This provided an adjustable thickness value for a unified thickness of all modules in the sunshade array. The Geometry parameter was then connected to the “G” (Base Geometry) input of the Box Morph component. This designated the geometry parameter for morphing the module into a twisted box. To complete the geometry section, the upper “B” (Box) output of the Bounding Box component was connected to the “R” (Reference Box) input of the Box Morph component. This referenced the bounded box in world coordinates for the Box Morph component (see Figure 5).

The last part of the algorithm was the surface section. This section used two “Slider” (Number) parameters, the “Dom” (Domain) component, the “Divide” (Divide Domain) component, and the “SBox” (Surface Box) component; however, it also used the Number parameter from the geometry section of the algorithm. First, the “I” output of the Domain component was connected to the “I” parameter.
input of the Divide Domain\(^2\) component. This provided a two dimensional base domain for an equally divided set of segments for the surface. The inputs for the Domain\(^2\) component, “U0”, “U1”, “V0”, and “V1,” do not need to be reset from their default values because the interval between the lower and upper limits for each dimension was set to exactly one. This generated the modular array to the exact dimensions of its associated surface; otherwise, any changes to the interval will cause it to incorrectly generate under or beyond the dimensions of the surface. One Number parameter was set to 37 and connected to the “U” input of the Divide Domain\(^2\) component. This provided the number of modules for the vertical dimension of the array. The other Number parameter was set to five and connected to the “V” input of the Divide Domain\(^2\) component. This designated the number of modules for the horizontal dimension of the array. Next, the “S” output of the Divide Domain\(^2\) component was connected to the “D” input of the Surface Box component. This gave a surface domain for creating twisted boxes on the surface. The Number parameter representing thickness for the Bounding Box component was also connected to the “H” input of Surface Box component. This matched the surface box thickness, or height, of the array with the bounding box thickness of the module. Finally, the Surface parameter was set to an outer segment of the surface, and then it was connected to the “S” input of the Surface Box component. This designated the surface segment in which the module elements would be generated (see Figure 6).

Figure 6: The surface section shown with the geometry section of the algorithm as seen in the Grasshopper workspace. A red preview will be seen in the Rhinoceros workspace when the Surface parameter is properly set.

At this time, the “B” output of the Surface Box component was connected to the “T” input of the Box Morph component. This used the resulting surface boxes as target boxes for the geometry section; thus, translating the module in to an array of modules along the associated surface segment. This completed the algorithm used for generating the sunshade; however, the entire algorithm was copied twice to complete the remaining segments of the entire surface; therefore, three separate algorithms were seen in the workspace. Because the Geometry parameter was already set to the design module, the Surface parameter only needed to be reset to one of the surface segments for each remaining algorithm. In the algorithm associated with middle surface segment, the Number parameter associated with the number of modules generated in the horizontal direction was reset to four. As specified in the design of the modular array, this created 14 columns of modules rather than 15 (see Figure 7).

Figure 7: The completed algorithms for each associated surface segment. In the middle algorithm, the Number parameter for the “U” input of the Divide Domain\(^2\) component is set to four to create 14 columns in the sunshade instead of 15. A red preview of the completed sunshade model will be seen in the Rhinoceros workspace.

The sunshade installation model was successfully generated as a red transparent preview in the Rhinoceros workspace. Once verified to be correct, the Box Morph component of each surface segment’s algorithm was highlighted, and the red model preview appeared green. Then the “Bake Selected Objects” tool was used to generate the model as actual elements in Rhinoceros. The sunshade model was officially generated in the Rhinoceros workspace; therefore, the algorithm file was saved, and the
Grasshopper plug-in could be closed. Lastly, the surface segments were moved away from the model (see Figure 8).

Next, the unfolded module pairs were exported into a computer-aided drafting program. AutoCAD was used for this project. The paired modules were highlighted, then exported as an AutoCAD file (.dwg) using the “default” setting option. At this point, all work in Rhinoceros was complete, and post-processing work in AutoCAD could begin.

Opening the exported AutoCAD file, a template for each cutting sheet was first drawn using a layer that would not be used as a cutting layer. Then, extraneous lines were deleted from each unfolded pair of modules; however it was important not to delete any lines representing folding points for the pairs. For this project, the top and bottom of each pair of modules was a triangle; therefore to reinforce the connection between separate pairs in each column, the triangles were changed into rectangles by adding two lines and deleting the extraneous line. Next, a coordinate code designating the position of the pair in the array was added for the construction stage. Once complete, all lines in the unfolded pairs designed to be cut were put on the cutting layer for the cutting equipment, and then all folding lines and element coordinates were put on a layer assigned to etching for the cutting equipment. Finally all paired unfolded modules were positioned on a cutting template (see Figure 10).

Post Processing

With the sunshade installation modeled in Rhinoceros, post-processing could begin. First, pairs of modules from each column were joined from bottom to top using the “Join” command (Join). Joining elements from top to bottom will cause incorrect unfolding of the surfaces. When all elements were joined appropriately, each pair must was unfolded using the “UnrollSrf” command (Unroll Surface). It was important to keep the pairs organized according to column so they could be labeled for construction in the proper position later (see Figure 9).

Construction

With the cutting sheets ready, manufacturing and construction of the designed sunshade installation could take place. The cutting equipment used for this project was a Universal Laser Systems Inc. X-660 laser cutter. Meticulous care was taken to insure cutting settings were identical for each sheet. Once the paired modules were cut from the material templates, they were folded to shape and assembled. A quick-adhering glue product was used to complete assembly of each module pair. Once each pair
was assembled, they could be positioned in the built wood frame according to their designated coordinates, and finally connected to the frame and other pairs using the previous glue product (see Figure 11). To complete the project, the sunshade installation was then mounted to the interior frame of the designated window in the zero energy house prototype (see Figure 12).

Figure 11: The construction process beginning with the laser-cut module pairs and ending with the arrangement of the module pairs in the wood frame.

Figure 12: The mounted sunshade installation in the UT Zero Energy House prototype.

A Case Study: CBD Media Tower

A separate design team implemented Rhinoceros and Grasshopper to create and model an energy efficient curtain wall system for a design proposal titled, “CBD Media Tower.” This design proposal was for the first “eco-friendly” high-rise building in the Central Business District in Beijing, China. Designed for one of the largest internet and cable television providers in China, VODone, this mixed-use building combined multiple telecommunication companies in a “tower-in-tower” concept interconnected by “sky gardens” and lobbies. Program space on the first six floors consisted of subterranean parking, retail, cinemas, and restaurants. Floors seven through 64 were designed to be primarily office spaces for various tenants. Green atrium spaces, designed to be meeting and dining spaces, were incorporated to separate the offices spaces. Finally, a “sky garden” was designed for the rooftop which could be open to the public (see Figure 13).

Figure 13: (Left to Right) A night and day rendering of the CBD Media Tower design proposal. The outer layer of the curtain wall system was created in Rhinoceros using Grasshopper.

Figure 14: A night and day rendering of the CBD Media Tower design proposal. The outer layer of the curtain wall system was created in Rhinoceros using Grasshopper.

Designed to be energy efficient, the proposed design explores concepts in natural ventilation, passive heating and cooling, ground cooling, stack ventilation, solar energy collection, and electrochromatic exterior glazing for shading. A dual layer exterior curtain wall system was designed to provide passive heating and cooling; however, each layer provides additional functions to the exterior design. The inner glazing envelope of the building was designed to be a social interface to the public. An LED multimedia system embedded into the curtain wall would allow the building itself to present programmed visual media to the public. The design of the outer glazing envelope was inspired from the qualities of fish scales, and it was designed to provide ventilation as well as solar energy. This was the building component generated by Grasshopper in Rhinoceros. Using a similar box morphing algorithm to the one created for the sunshade project, a triangular glazing element was arrayed across a surface representing the outer layer of the exterior building facade. This created a quick model for visualizing operable curtain wall elements that could open to provide ventilation as well as move to track the sun for optimal solar energy collection to embedded photovoltaic cells (see Figure 14).
Conclusion

These projects exhibited many benefits to using generative modeling tools; however, they also revealed improvement and innovation opportunities. The first apparent benefit is swift visualization. Any designer who has the algorithms for these projects or others designed for generative modeling can quickly visualize complicated geometries which, otherwise, would take much time and energy to draw or model. This also implies adaptability in generative modeling. In these projects, as long as a single module could be built in Rhinoceros, it could be set as the geometry and then generated as a morphed array across any designated surface using the Grasshopper plug-in. The adaptive qualities shown in each project and further research with the algorithm used in the sunshade project proposed an innovation in modeling. Could a single algorithm be used to generate the basic elements of an entire project model? In other words, could a single algorithm be used to generate floor planes, columns, curtain walls, roof planes, etc. for an entire building project? If these elements could be combined as a single module, it seems highly possible they could be generated as an array across a designated surface representing the footprint for a building model.

The post-processing in Rhinoceros for manufacturing of the sunshade installation provided opportunities for improvement with generative modeling technology. Because of the extensive time required to unfold each module pairs in Rhinoceros for the sunshade project, an improvement to the Grasshopper plug-in would be to provide algorithm elements which could provide post-processing steps. Examples could be algorithm elements that could unfold all the modules simultaneously, or one which could delete extraneous lines from the unfolded modules.

In general, generative modeling technology appears to support sustainable design due to increased speed in visualization as well as precision in computer-aided manufacturing. The ability to visualize complex geometries at a faster rate increases productivity and speed of workflow. Furthermore, the ability to manufacture complex geometric elements without needing to draw them manually or digitally beforehand decreases the opportunity for errors, wasted energy, and extended manufacturing. In general, these qualities prove that generative modeling technology gives beneficial opportunities as a new method for modeling projects in computer-aided design.

Acknowledgments

The sunshade installation project was developed by Casey Arthur with Allie Behnke (alliemb@gmail.com) and Andrew Ruff (andrew.d.ruff@gmail.com) as the other members of the design team, and their support is greatly appreciated. The CBD Media Tower design proposal was

Figure 14: The design process for the outer envelope of the dual layer curtain wall system. Beginning with the qualities found in fish scales, a box morphing algorithm was created in Grasshopper to array a triangular glazing element across a surface representing the building façade. The final product was a generated model which could investigate the characteristics desired. The model is seen here as preview in Grasshopper and as rendered image after being processed and exported from Rhinoceros.
created by Edgar Stach with Wolf Loebel (wla@rheinschiene.net) and Yuri Melnichenko (ymelnic1@gmail.com) and their contribution is also greatly appreciated.

References

