archi DOCT

The e-journal for the dissemination of doctoral research in architecture

July 2019

www.archidoct.net

ISSN 2309-0103

13

FORCES

Listed in:
Scopus®

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

Environmental aware shell design

Using solar path as a form finding force

Evangelos Pantazis // Viterbi School of Engineering, University of Southern California

Abstract

This work focuses on the development of a design methodology and the implementation of a design toolkit that can be used for the form finding of shell structures by incorporating environmental parameters. This study aims to extend traditional form finding approaches by introducing agent based modelling and simulation tecnhiques which allow for the integration of daylight as a shaping force apart from typical loads such as the gravity force. Within the developed Multi Agent Systems (MAS) framework the steering of form beyond purely form found shapes is explored by introducing behaviors which relate to the orientation of the site and the related solar path. An experimental design is developed using an existing thin shell concrete structure design by H. Isler to apply and test the proposed methodology and prototypical toolkit.

Keywords

Form Finding; Multi Agent systems; Shell design; Agent Based modeling and simulation

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

1. Introduction

Architectural design so far has been rooted into descriptive modelling which is used to produce the geometrical models that are then passed to engineering disciplines to conduct analysis before it moves to construction. Advances in information technologies and the development of custom digital design tools on different programming platforms have brought about a shift towards integrated design approaches between the fields of Architecture, Engineering and Construction (AEC). Digital fabrication and the rapidly growing research on robotic construction has presented an exciting opportunity to merge digital and physical tools and processes for constructing non-standard building structures faster and at lower costs (Gramazio and Kohler, 2014).

The realization of large-scale projects with free form geometries in the last 20 years has also presented the AEC industry with challenges. Firstly, it has shown that there is a necessity for a close collaboration between architects and engineers from early design phases in order to provide elegant and efficient design solutions(Rahman, 2010). Additionally, it has shown that despite the overall successful integration of parametric design models in practice, as can be seen in façade construction, environmental benchmarking or structural design; parametric design for complex building projects remains labor intensive and rather manual. In other words, current design methodologies could be characterized as computer based than computational (Marincic, 2016). As a result, these technological advancements have created a demand for addressing the architectural form finding in a more methodological and computational way in order to be able to provide sustainable solutions that can cater for environmental and structural parameters as well as human behavior (Rahman, 2010).

One of the most prominent and rigorous research paths on developing formal computational design methods come from the field of shell design and form finding (Block et al., 2017, Rippmann et al., 2013). Despite the increasing interest and application of computational design approaches, the capacity of computation is not fully utilized as,a lot of times, there is no direct connection between the tools that are generating designs and those that perform analysis (Grabner et al., 2013). Thus, there is a necessity for design tools which support integrated design solutions and enable the extension of forms established through conventional analytical techniques (Von Bülow, 2007). Digital tools should help retain and extend the designers' creative capacity in the early design stage but also enable them to more rigorously consider design solutions that reduce the energy footprint of both the design to construction processes and the life-cycle of buildings.

To be able to manage increasing building complexity our hypothesis is that the analogy which currently exists in digital architecture between the designer (user) and machine (digital tool) should be reversed. In this analogy the designer acts like an apprentice that uses an interface (i.e. a design software) or a specific language (python, C++, Java) to pose questions to the master (computer) and respectfully awaiting the answer. In order to promote the designers creativity, future design tools need to be conceived not as drafting aids but as the designer's collaborating partners, which are capable, when given a set of specifications, to generate proposals (design alternatives) that the user/designer can evaluate and critique. In such a situation the tool is expected to build knowledge through the interaction with the user and the processing of multiple data sets.

Along those lines, this paper discusses the problem of introducing environmental parameters such as the position of the sun in the form finding process of free form shells within the framework of multi agent systems (Figure 1). The introduction of behavioral form finding attempts to extend

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

existing digital form finding which have been based upon empirical models introduced in the 20th century such as the hanging chain and fabric (cloth) models as well as graphic statics (Rippmann et al., 2013). It also attempts to shift the focus from exploring structurally optimal solutions towards developing tools that can help designers extend their creativity by predicting behaviors and steer form finding beyond structurally optimal solutions by introducing environmental parameters.

1.1 Structure of the paper

In the following sections the proposed design methodology is described and tested in an experimental design. A short review of existing form finding methods shows the relevance for implementing a MAS framework which allows for the modelling of design objectives into agent behaviors. Our approach is based on the combination of particle physics, agent based modelling and analytical solvers. The paper is structured as follows: In section 2, a brief overview of existing design tools and form finding approaches are surveyed along with the basic concepts of multi agent systems. In section 3 the MAS framework and its related components are presented. In section 4 we present the implementation of the methodology into a case study. In section 5 we discuss our conclusions.

2. Background

Early research efforts in the field of design computing focused in developing Computer Aided Design (CAD) tools which reduced complexities relating to drafting and the automation of drawing production rather than developing new design methods and tools which transcribe fundamental formative processes, into architectural design (Scheurer, 2010). Parametric and subsequently performance based design, emerged as an integrated approach, which allows designers to consider environmental and structural parameters in the design development stage (Gerber et al., 2012, Keough et al., 2010). With it has a come a new maturity that promises to transcend the formal and geometric innovation that were mainly driving the interest in using digital technologies (Oxman, 2008). More recently, research in the field of architectural design and building engineering has largely focused on bridging the gap between physical and digital real and the integration of fabrication and material constraints in the early design stage (Oosterhuis, 2011). Examples such as the New Elephant House in the Copenhagen Zoo, or the grid shells of the Cooled Conservatories in Singapore demonstrate the potential of using computational techniques to create sustainable designs by integrating multiple disciplines such as architectural with structural and environmental design(Davey et al., 2010; Peters, 2008).Such built structures have proved the advantages of using multi objective optimization and multiple analyses in order to generate complex yet coherent and sustainable buildings (Peters et al., 2018).

By surveying the literature one can observe a dichotomy between two main paths in developing digital workflows for architectural design. The first path has led to distributed yet disconnected solutions, where multiple task specific tools are being developed based on generalized project requirements (i.e. energy modeling, structural modelling). The second path, strives for a centralized approach, which assumes a streamlined design process where all team member add their data into a central model built within one software package.

This approach can be useful for later design stages but is rather inflexible in the early design stage(Mackey et al., 2018). The former approach is manifested by the increasing number of computational design tools and communities while the latter one is manifested by the increasing adoption of Building information Modelling (BIM) in the last decade (Eastman et al., 2011).

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

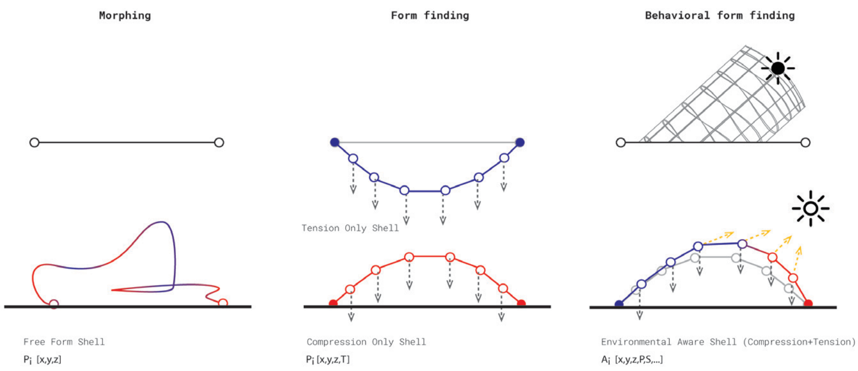

Figure 1.

Illustration showing diagrammatically three different approaches towards form finding of shells. On the left. The resulting shape is the outcome of free form morphing, in the middle the shape is the outcome of physical forces (gravity and tension) and on the right the shape is the outcome of physical and digital forces (gravity, tension and light attraction)

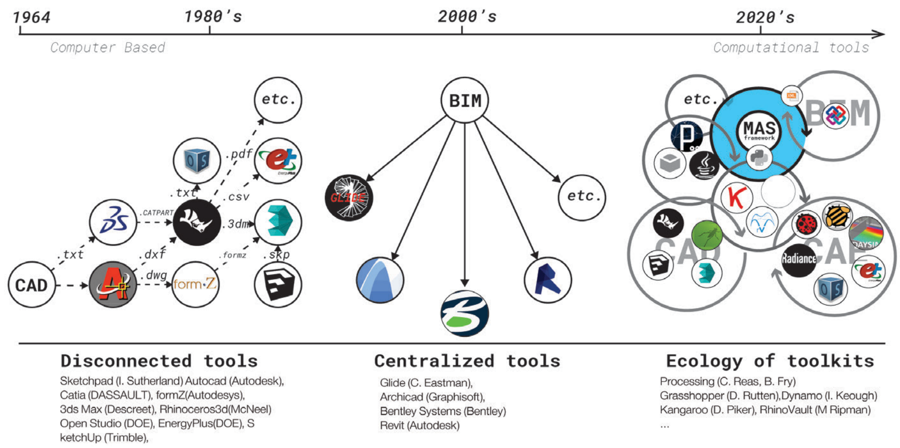

Figure 2.

Three different design software integration methods as they have evolved over time in the AEC industry

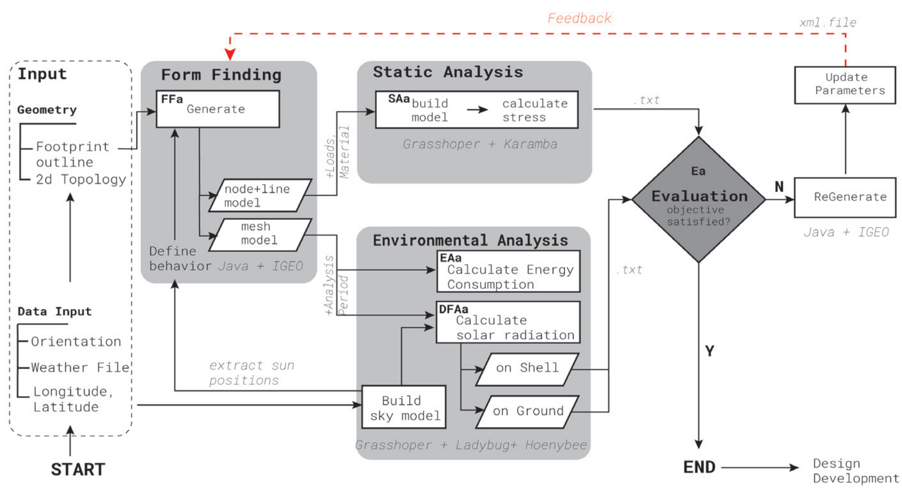

Figure 3.

Flowchart diagram of the proposed behavioral form finding workflow

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

Although both of the aforementioned approaches have their advantages and have been adopted to a big degree from the Architecture Engineering and Construction (AEC) industry, they share a common disadvantage. They both focus on the tools themselves rather than on the established design workflows and the interconnection between tools (Mackey et al., 2018). Instead of developing numerous disjointed design tools or an all-in-one tool solution, intuitive and task specific toolkits that are focused on enhancing existing workflows between software may be more effective towards managing building complexity. Moreover computational techniques can help reconsider and revive traditional design methods which were developed in the past but became obsolete due to being particularly tedious or time consuming.

The ability to generate and evaluate multiple design alternatives is essential in architectural design, as it has been shown by Woodbury and Burrow (Woodbury et al., 1999) and computational methods can extend the designers cognitive capacity by enabling more rigorous design exploration (Gero et al., 2008).

2.1 Computational Form Finding

For instance, several physical form finding techniques were developed independently in the 20th century by practitioners such as A. Gaudi, H. Isler, F. Candela and F. Otto(Adriaenssens et al., 2014). These techniques were empirical and were driven by the motivation to create open plan spaces with large spans that were conditioned by economic and material constraints. Although they opened a new set of possibilities to designers, due to their complexity they remained largely unexplored until recently. An increasing number of researchers working on the intersection of design, engineering and computing have started revisiting such methods from a computational perspective in an attempt to enable architects deal with hard design problems that include engineering and fabrication constraints in a more rigorous way(Gerber et al., 2013).

In the last two decades a number of computational based approaches have been developed for exploring architectural form based on the concepts of form finding and optimization (Kilian, 2014b), evolutionary computation and behavioral design (Menges, 2007) as well as rule based models (Fricker, Hovestadt et al., 2007). Kilian, inspired by A. Gaudi hanging chain models developed one of the first digital form finding tools (Kilian, 2006). The tool was based on the hanging chain principle which was introduced by Hooke in the 17th century and demonstrated how fabrication schemas can be linked to real time form finding simulation (Magna, 2017). Piker has introduced a particle physics engine for simulating structures based on the combination of Dynamic Relaxation and the co-rotational formulation of Finite Elements Methods(Piker, 2013). Rippmann introduced an interactive form finding tool based for compression only vault design which is based on graphic statics (Rippmann et al., 2012). The tool is based on Thrust Network Analysis (TNA), a method which generates possible 3d shell geometries by combining projective geometry, duality theory and linear optimization (Block et al, 2007).

In the field of Multi Agent Systems and Agent Based Modelling (ABM) there have been developed a number of design tools inspired by complex adaptive systems and emergent behaviors observed in nature (Bonabeau, Dorigo et al., 1999). These tools are driven by environmental conditions and allow behavioral modelling but have mostly focused on specific agent models such as the "boids" developed by C. Reynolds (Reynolds, 1987). Additionally, although in many disciplines MAS has been used for optimization processes in architectural design, they have been mainly used for generating designs.

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

More recently there has been a significant effort towards developing integrated design approaches that narrow the gap between modelling and analysis by using data to drive the design exploration process. Yet in most cases the rationalization is happening after a design is generated and thus more research is necessary to develop tools which use local relationships and analytical data that generate models that are pre-rationalized.

The aforementioned approaches have pushed the boundaries of integrated architectural design and generative design respectively. However, the former approaches have mainly emphasized in the integration of geometric and structural design (boundary condition, supports, loads etc.) but are not considering environmental parameters such as the location and/or position of the sun in the form finding process (Kilian, 2014a), while the latter ones have yet to develop agent models specific to the AEC, which are relevant in the contemporary practice (Pantazis and Gerber, 2018).

3. Methodology

The current study focuses on the early design stage and introduces a methodology which enables designers to form find different design alternatives by considering also the position of the sun and evaluate them based on their structural and environmental performance. By introducing a MAS approach where environmental and structural parameters can be modelled as behaviors the objective is to extend existing form finding approaches and enable designers augment purely form found shapes but also provide them with the capacity to evaluate them.

Another objective is to implement and test a toolkit that supports agent-based modelling, connect existing analytical solvers and enables the automated generation of design alternatives. The generation of design alternatives I based on a combination of bottom up rules that relate to design intentions and top down rules that relate to regulations and constraints.

Specifically, a Multi Agent Systems framework has been developed and a toolkit has been implemented (Gerber et al., 2017, Pantazis et al., 2018) that is built on top of Processing IDE (Reas, 2007), the 3d modelling software Rhinoceros 3d as well as a group of analytical plugins that we access via the visual programming editor Grasshopper. IGeo library is implemented in Processing in order to develop different agent classes and behaviors for generating design alternatives (Sugihara, 2014) while d3 library has been implemented for visualizing analytical result (Bostock et al., 2011). The Rhinoceros/Grasshopper software is used to generate 2d topologies, the Karamba plugin is used in order to perform Finite Element Analysis, while Ladybug and Honeybee plugins (Roudsari et al., 2014)are used to gain access to environmental analysis software such as Radiance and Energy plus. Custom python scripts are used in order to ensure communication between the different platforms and for developing the custom agent behaviors.

Within the proposed framework the modelling procedure can be summarized as follows: 1) provide a footprint of a structure in 2d (i.e. polyline), 2) specify support points (i.e. 3d Points) 3) generate a network topology for the area within the footprint (i.e. rectangular, triangular etc.), 4) generate a solar path by defining the location and orientation of the footprint 5) select specific time periods that she is interested, 6) model a form finding agent by specifying initial position (x,y,z), gravity force

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

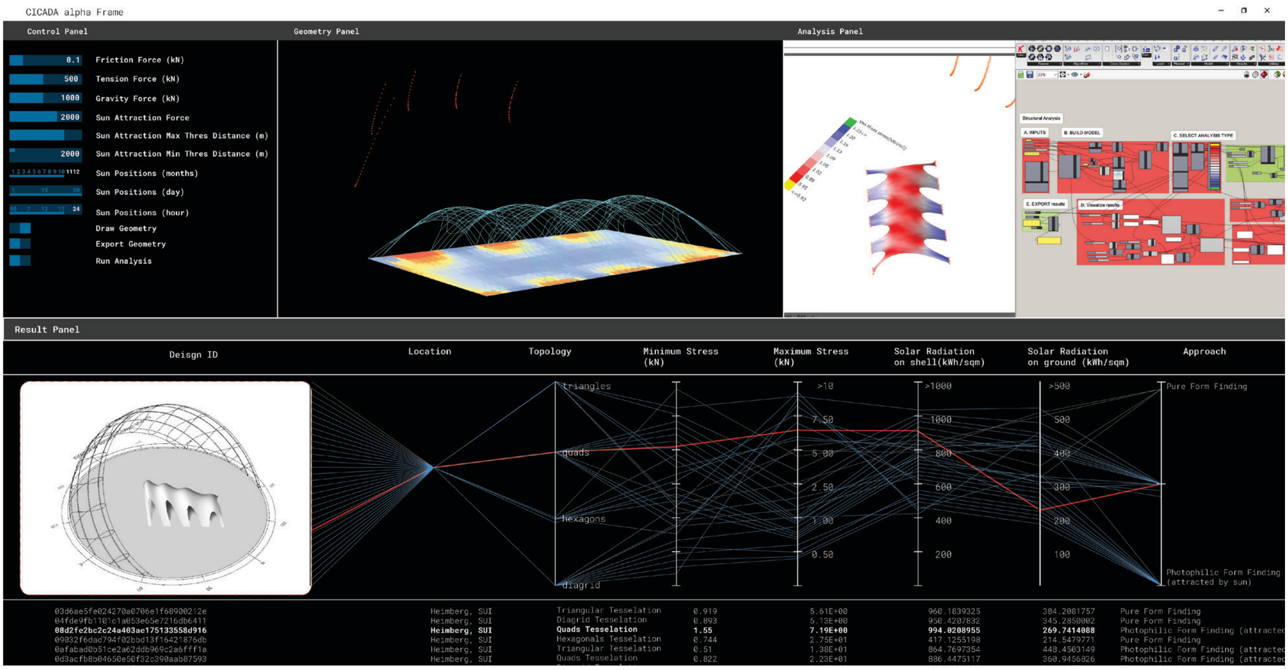

Figure 4.
Graphical User Interface of the alpha version of the tool. On the top left side (control panel) are all the input parameters, in the middle is the geometry viewport and on the left is the window where we call Rhinoceros 3d and Grasshopper. In the bottom panel the generated results are shown

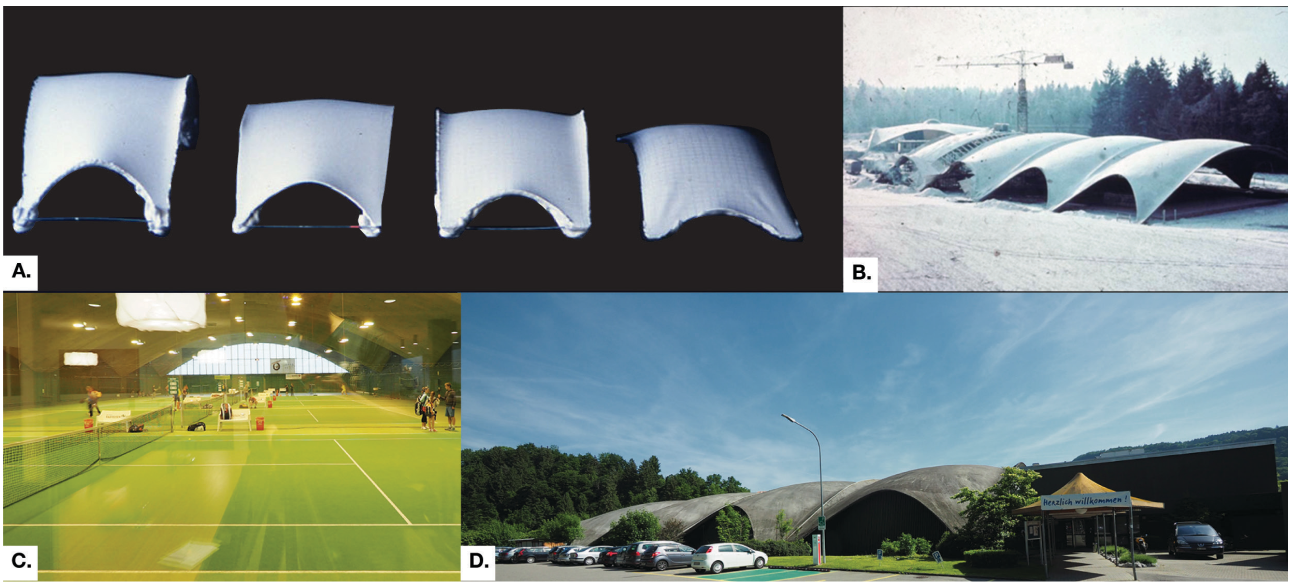

Figure 5.
Scaled fabric models (a)circa 1977, photos from the construction process of the Heimberg Tennis Hall (b) in 1973 as well as photos from the building in its current state (c,d) in 2013

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

(loads), sun attraction force, tension, friction and a probability to connect with other agents (i.e. p = 1 , agent connected with neighboring agents, p=0 agent has no connections), as well as objective targets for solar radiation and stress.

Once the above are defined the designer can specify agent behaviors i.e. photophilic or photophobic behavior depending on what they are trying to achieve (Figure 4 - Control Panel). The purpose of defining these behaviors is conceived as a way to explore shapes beyond the purely form found ones. In the case of the photophilic behavior each agent is leaning towards sun positions (specified by the user), thus resulting into shell designs that allow light to enter , while in the case of the photophobic behavior the agents tend to avoid sun positions. The designer runs the system for a specified amount of time or number of iterations by calling a java applet. At each iteration the agents interact up until they reach a state of equilibrium, depending on the forces applied to them (Figure 4 - Geometry Panel). The geometry is then exported as a .3dm file and is automatically passed to Rhinoceros 3d where a daylight analysis on and below the form found geometry is performed using Grasshopper/Ladybug tools as well as a Stress analysis using Grasshopper/Karamba (Figure 4 - Analysis Panel). Once each analysis is finished the results are saved in a .csv file along with an image of the design. The csv file is parsed in order to visualize the results using parallel line plots (Figure 4 - Results Panel). The designer can define an objective function, such as reach an X amount of daylight/area (measured in kwh/h) and run the system using a stochastic method such as hill climbing or simulated annealing or can simply interact with the tool. In short, each iteration consists of four steps: First generate a design given the input geometry and the agents' parameters. Then perform structural and environmental calculations. Next archive and visualize both geometry and analytical results and lastly evaluate if the results meet the objectives and if not adjust the sun force or initial topology and run again the simulations.

4. Experimental Design

In order to test our methodology, we apply it on an existing building which was designed in the 1970's using (physical) form finding techniques. The objective is to explore design alternatives of the purely form found shape by introducing the solar path. Specific positions of the sun throughout the year are used as attractor points which augment the purely form found shape, in order to meet desired environmental performance underneath the structure, (that is a specific amount of daylight).

4.1 Revisiting the Sports Center at Heimberg by H. Isler

The building we are using to run our pilot study is a sports center in Switzerland designed by Swiss designer H. Isler (1926-2009). The Sports center of Heimberg comprises of multiple thin-shells, which are constructed with reinforced-concrete. The structures were completed in 1978 and are still in use (Figure 5). Isler designed this structure by studying meticulously fabric models, a method which he became known for (Chilton et al., 2017). After the successful completion of the shell in Heimberg he went on and used the same model for a number of similar structures across Switzerland which were used as tennis halls. The same structural solution was uniformly applied to four sites and therefore in none of those cases the location and orientation of the structure were considered in the design. All the realized structures have a span 48.00m m and length that varies from 75 to 105 m.

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

In this study we explore how a) the topology of the line network –i.e. our digital fabric can affect the form finding, b) how we could come up with alternative designs by integrating the solar path in the form finding process and c) how can we specify agent behaviors that can create alternatives which satisfy the environmental performance targets without hurting the structural integrity of the shell.

In order to achieve the above we add a solar force to each agent apart from the basic forces used in traditional form finding approaches such as gravity loads, tension, and friction. By introducing and modulating this "virtual" force we can develop abstract behaviors that relate to the position of the sun namely a photophilic and a photophobic behavior. In terms of the topology of our line network we test four different cases (rectangular, triangular, rhomboidal/diagonal, hexagonal) and test how each one affects the stiffness of the global geometry.

Three performance metrics are used in order to evaluate the behaviorally form found shells against purely form found ones. For each generated design we calculate:

- the principal stresses (kN) on the shell surface
- the solar radiation on the shell surface
- the daylight radiation underneath the structure

As it has been described in the previous section, Grasshopper plugins Ladybug and Karamba together with some simple objective functions are used as external solvers for the evaluation of the performance values. The developed MAS toolkit functions as an interface between the different platforms and facilitates the data passing between the different softwares.

5. Results and Discussion

In Figure 6, we tabularize the different shells that have been generated using four different topologies and three different form finding approaches. On the left, we illustrate the different outcomes using pure form finding while on the middle and right rows we illustrate the result of two different behaviors namely: a photophobic and a photophilic behavior

In the tables, we compare the purely form found shells against the behavioral ones. It is shown that by adding an extra force to the agents, we can augment the purely form found shapes while by changing the topology of the initial network, we can modulate the stiffness of the global geometry. The rhomboidal/diagonal topology results into stiffer global geometry which consequently is not affected by the solar force while the hexagonal one seems to be least stiff and is the most affected. We show the extreme values for the agent behaviors after which the shells start failing structurally.

6. Conclusions

This paper introduces a behavioral form finding method and demonstrates that Multi Agent Systems and the introduction of environmental parameters can be used as a design driver to explore the design beyond pure form finding approaches. The experimental design shows that by introducing the solar path and combining agent based modelling with analytical solvers the traditional form finding techniques can be extended. By using abstract behaviors, it is demonstrated that it is possible to form find shell geometries that consider the trade of between structural constraints and environmental parameters such as solar radiation and daylight.

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

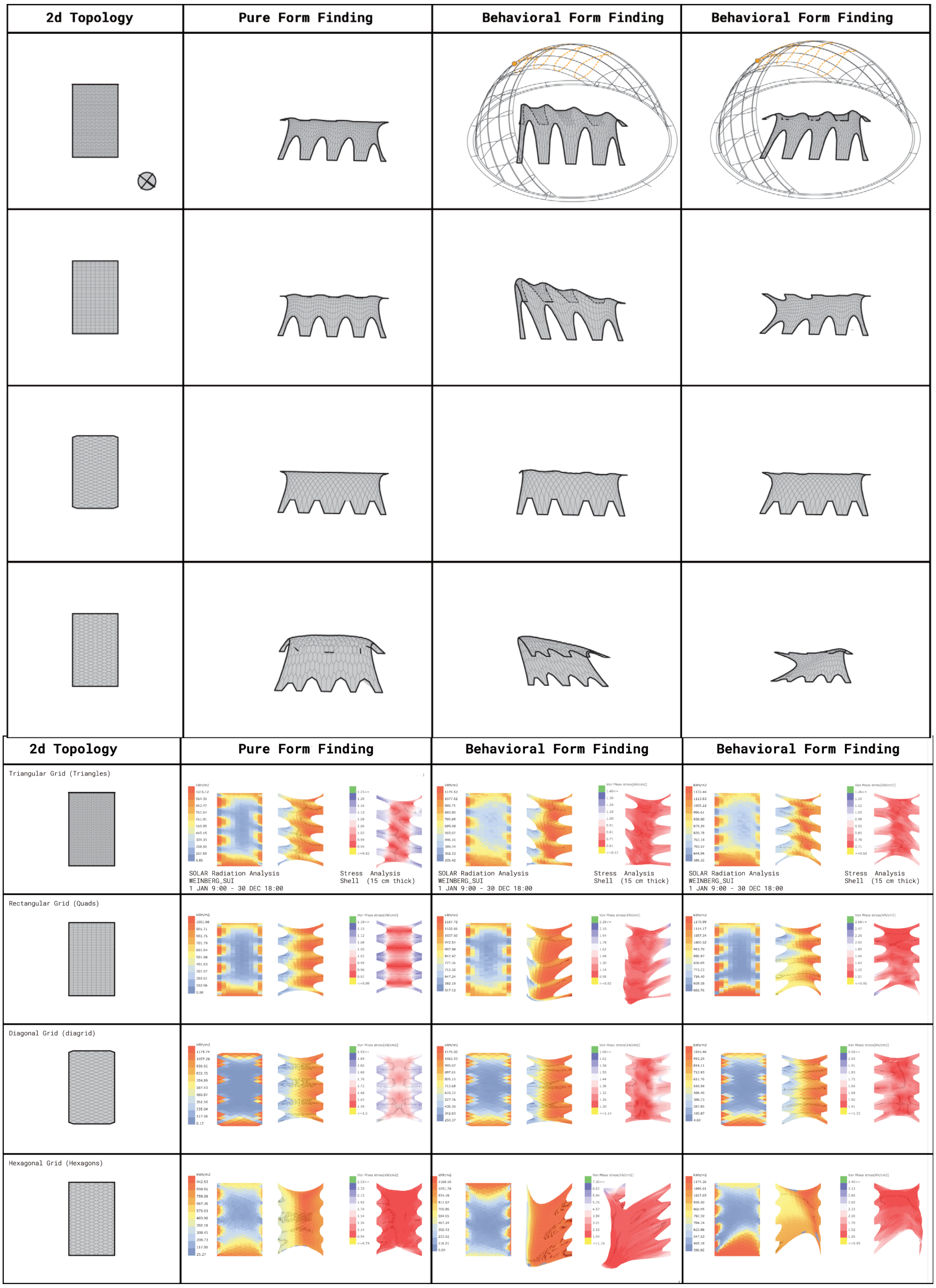

Figure 6.

(Top) Table with the different topologies that have been tested in the experimental design and (Bottom) table with combined analytical results for the purely form approach as well as two behavioral form finding approaches namely photophilic (middle) and photophobic (right)

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

The designer can modify the behaviors to generate designs and by visualizing performance metrics can rapidly asses them. By testing different values for each behavior we can map the solution space within which we can generate alternatives that satisfy both structural and environmental performance objectives. As a future step, we aim to apply the tool in multiple sites where similar structures have been places and explore how can we alter the geometry based on the different orientations in order to achieve the same environmental performance.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No.1231001. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

References

Adriaenssens, S., Block, P., Veenendaal, D. and Williams, C. 2014. *Shell Structures for Architecture: Form Finding and Optimization.* London and New york: Taylor & Francis - Routledge.

Block, P. and Ochsendorf, J. 2007. Thrust network analysis: A new methodology for three-dimensional equilibrium. I*nternational Association for Shell and Spatial Structures* 155 167.

Block, P. et al. 2017. NEST HiLo: Investigating lightweight construction and adaptive energy systems. *Journal of Building Engineering* (12) 332-341.

Bonabeau, E., Dorigo, M. and Theraulaz, G. 1999. *Swarm intelligence: from natural to artificial systems.* New York, USA: Oxford university press.

Bostock, M., Ogievetsky, V. and Heer, J. 2011. D³ data-driven documents. *IEEE Transactions on Visualization & Computer Graphics*(12) 2301-2309.

Chilton, J. and Chuang, C.-C. 2017. Rooted in nature: aesthetics, geometry and structure in the shells of Heinz Isler. *Nexus Network Journal* 19(3) 763-785.

Davey, M., Bellew, P., Er, K., Kwek, A. and Lim, J. 2010. Gardens by the Bay: High performance through design optimization and integration. *Intelligent Buildings International* 2(2) 140-157.

Eastman, C.M., Eastman, C., Teicholz, P. and Sacks, R. 2011. B*IM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*: John Wiley & Sons.

Fricker, P., Hovestadt, L., Braach, M., Dillenburger, B., Dohmen, P., Rüdenauer, K., Lemmerzahl, S. and Lehnerer, A. 2007. *Organised Complexity.* Frankfurt am Main, Germany: 25th eCAADe Conference Proceedings.

Gerber, D. and Pantazis, E. 2017. Interactive Design of Shell Structures Using Multi Agent Systems. In Gülen Çağdaş et al. eds. *CAADFutures 2017*. Istanbul, Springer. 617-629.

Gerber, D.J. and Lin, S.-H.E. 2013. Designing in complexity: Simulation, integration, and multidisciplinary design optimization for architecture. *Simulation* 1-24.

Gerber, D.J., Lin, S.-H.E., Pan, B.P. and Solmaz, A.S. 2012. Design optioneering: Multi-disciplinary design optimization through parameterization, domain integration and automation of a genetic algorithm. *SimAUD 2012,* Orlando, FL, USA.

Gero, J.S. and Sosa, R. 2008. Complexity measures as a basis for mass customization of novel designs. *Environment and Planning B: Planning and Design* 35(1) 3-15.

Grabner, T. and Frick, U. 2013. GECO™: Architectural Design Through Environmental Feedback. *Architectural design* 83(2) 142-143.

Gramazio, F. and Kohler, M. 2014. Authoring Robotic Processes. *Made by Robots* 229 136.

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

Keough, I. and Benjamin, D. 2010. Multi-objective optimization in architectural design. Proceedings of the 2010 Spring Simulation Multiconference, Society for Computer Simulation International.

Kilian, A. 2006. Design innovation through constraint modeling. *International Journal of Architectural Computing* 4(1) 87-105.

Kilian, A. 2014a. Steering Form. In S. Adriaenssens et al. eds. *Shell Structures for Architecture: Form Finding and Optimization.* London and New york, Taylor & Francis - Routledge.

Kilian, A. 2014b. Steering of form. *Shell Structures for Architecture: Form Finding and Optimization* 131.

Mackey, C. and Roudsari, M.S. 2018. The Tool (s) Versus The Toolkit. *Humanizing Digital Reality*. Springer. pp. 93-101.

Marincic, N. 2016. Towards communication in CAAD: Spectral Characterisation and Modelling with Conjugate Symbolic Domains. I*nstitute of Technology in Architecture, Chair for Computer Aided Architectural Design.* Zurich, Switzerland, ETH. 231.

Magna, R.L. 2017. Bending-Active Plates: Strategies for the Induction of Curvature through the Means of Elastic Bending of Plate-based Structures. *Institute for Structural Design.* Stuttgart, University of Stuttgart. 173.

Menges, A. 2007. Computational morphogenesis. Proceedings for 3rd International ASCAAD Conference, Alexandria, Egypt.

Oosterhuis, K. 2011. *Towards a New Kind of Building: A Designers Guide for Non-Standard Architecture*. Netherlands: NAi Uitgevers/Publishers Stichting.

Oxman, R. 2008. Performance-based design: Current practices and research issues. *International Journal of Architectural Computing* 6(1) 1-17.

Pantazis, E. and Gerber, D. 2018. A framework for generating and evaluating façade designs using a multi-agent system approach. *International Journal of Architectural Computing (IJAC)* 16(4) 248-270.

Pantazis, E. and Gerber, D.J. 2016. Design Exploring Complexity in Architectural Shells -Interactive form finding of reciprocal frames through a multi-agent system. In A. Herneoja, T. Osterlund and P. Markkanen eds. eCAADe. Oulu, Finland.

Peters, B. 2008. New Elephant House, Copenhagen. ACADIA 2008 Silicon and Skin.

Peters, B. and Peters, T. 2018. *Computing the Environment: Digital Design Tools for Simulation and Visualisation of Sustainable Architecture:* John Wiley & Sons.

Piker, D. 2013. Kangaroo: form finding with computational physics. *Architectural design* 83(2) 136-137.

Rahman, M. 2010. Complexity in building design. In I. Ruby and A. Ruby eds. *Re-inventing Construction.* Berlin, Ruby Press. pp. 440.

Reynolds, C.W. 1987. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH computer graphics* 21(4) 25-34.

Reas, C. 2007. *Processing : a programming handbook for visual designers and artists.*

ISSN 2309-0103
www.archidoct.net
Vol. 7 (1) / July 2019

Cambridge, MA: MIT Press.

Rippmann, M. and Block, P. 2013. Funicular Shell Design Exploration. ACADIA 13: Adapative Architecture, Cambridge, Ontario, Canada.

Rippmann, M., Lachauer, L. and Block, P. 2012. Interactive vault design. *International Journal of space structures* 27(4) 219-230.

Roudsari, M., Pak, M. and Smith, A. 2014. Ladybug: A Parametric Environmental Plugin for Grasshopper to Help Designers Create an Environmentally-Conscious Design.

Scheurer, F. 2010. Materialising Complexity. *Architectural design* 80(4) 86-93.

Sugihara, S. 2014. iGeo: Algorithm Development Environment for Computational Design Coders with Integration of NURBS Geometry Modeling and Agent Based Modeling. ACADIA 14: Design Agency, Los Angeles, eVolo.

Von Bülow, P. 2007. An intelligent genetic design tool (IGDT) applied to the exploration of architectural trussed structural systems. *Institute for Lightweight Structures and Conceptual Design.* University of Stuttgart.

Woodbury, R., Burrow, A., Datta, S. and Chang, T.-W. 1999. Typed feature structures and design space exploration. *AI EDAM* 13(4) 287-302.