Discrete vs. Discretized Growth

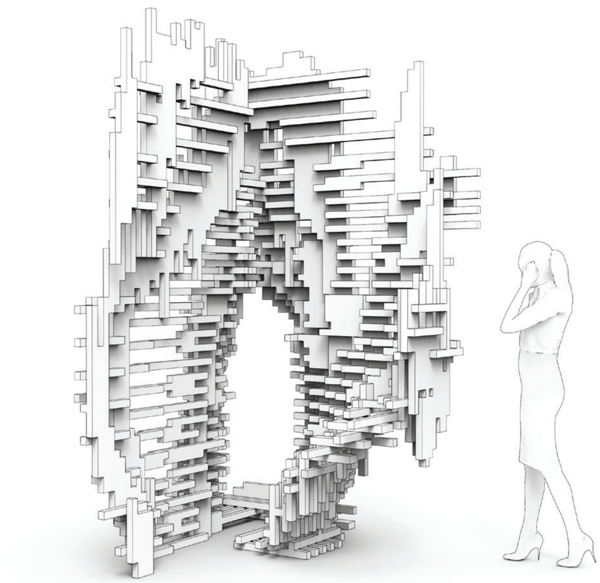
Christoph Klemmt
University of Cincinnati/
Orproject/University
of Applied Arts Vienna/
Architectural Association

Igor Pantic
Architectural Association/
UCL Bartlett

Andrei Gheorghe
University of Applied Arts
Vienna

Adam Sebestyen
University of Applied Arts
Vienna/Vienna University
of Technology/New Design
University St. Pölten

Discretized Fabrication of Geometries Generated with Cellular Growth Simulations

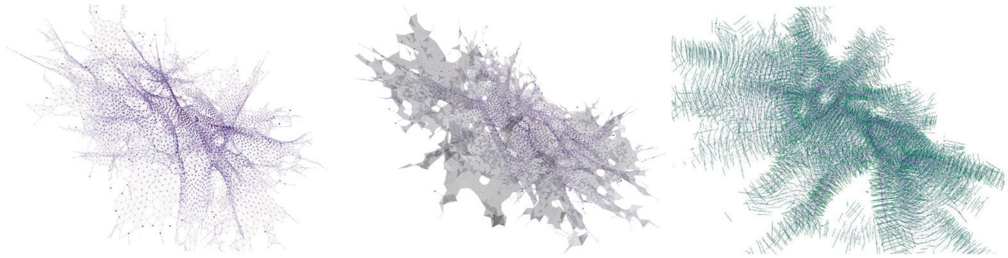


1

1 Render of Styx Prototype

ABSTRACT

Discrete computational growth simulations, such as Cellular Automata of Diffusion Limited Aggregation, appear often to be difficult to use for architectural design as their geometric outcomes tend to be difficult to control. On the contrary, free-form growth simulations such as Differential Growth or cell-based growth algorithms produce highly complex geometries that are difficult to construct at a larger scale. We, therefore, propose a methodology of discretized free-form Cellular Growth algorithms in order to utilize the emerging qualities of growth simulations for a feasible architectural design. The methodology has been tested within the framework of a workshop and resulted in the efficient construction of a large physical prototype.



2

2 Intercellular behaviors and the resulting data that are used as input parameters for the component placement: Cell Neighborhood, Local Planarity, and Cell Orientation

INTRODUCTION

The computational simulation of growth processes has been pursued by different disciplines. In computer science, especially Artificial Life, the aim is to create and understand processes that portray life-like behaviors, while in biology and medicine the focus is on creating models that simulate real-life to a high degree of accuracy.

On the contrary in design, art and architecture, the aim is often to generate new types of geometries with novel, highly complex or aesthetically pleasing characteristics, and in architecture specifically, geometries that may be able to solve specific functional requirements. A growth simulation, with its iterative development towards a larger accumulation of mass, provides the opportunity of an also iterative evaluation of the current state of the geometry, which can then influence the behaviors that guide the growth to develop towards a desirable outcome, on the global as well as local level.

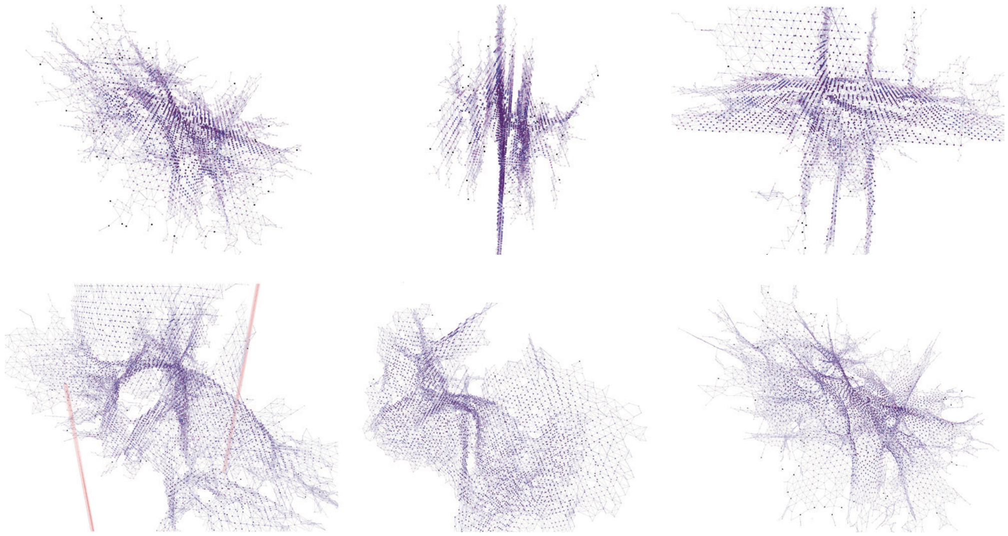
Various types of computational growth simulations have been developed. Due to the algorithmic complexities and the speed of computing, early simulations were often based on calculations within a rectangular grid, such as Diffusion Limited Aggregation (DLA) or Cellular Automata (CA) and Conway's Game of Life. On the contrary, more recent simulations make use of free-form geometry in 3D space, such as Differential Growth, Cellular Growth algorithms, or the CA-based SmoothLife.

The grid-based simulations can have very simple logics that generate a high degree of complexity and are therefore of great interest to the study of Artificial Life. However, the constraint of the grid results in very specific types of geometries that are usually not suitable for the simulation of biological behaviors, so here the free arrangement of geometry in space is usually used. Also from an architectural perspective, CA and DLA have been used for experimental research projects, but it has been found that the resulting geometries are usually impossible or very difficult to rationalize for real-world projects. The geometries that are generated by the algorithms are hard or impossible to control, and don't appear to be suitable for addressing functional architectural problems.

In respect to architectural applications, especially if intended for real-world projects, apart from the generation of a functioning geometry also the efficiency and economic buildability are of high importance. Although there have been many advances in construction methods of curved geometry, and the 3D printing of buildings is in development, economy in today's construction industry is still mostly driven by the use of repetitive elements and mostly rectangular geometries. In terms of buildability, therefore, the outcomes of the grid-based algorithms rate much better than the free-form geometries that are generated by more recently developed algorithms, which usually do not exhibit any repetition at all.

In order to develop growth simulations that can become suitable tools for architectural design, we are therefore proposing the use of discretized Cellular Growth simulations. The algorithm we propose is based on cells that can arrange free in 3D-space and that proliferate based on logics of cell division. The local as well as global behaviors of the cells can be guided through various forces and constraints that are acting on the cells. Although positioned free in space, the cells nevertheless always occupy voxels within an underlying 3D grid. The occupied voxels are then used to define the placement of repetitive components that are assembled to form the final geometry. The placement within the grid allows for further possibilities such as the cells directly reacting towards the grid locations, or the control of component alignment and patternisations.

To test the possibilities of the proposed methodology, we attempted its evaluation on three levels during a design workshop: as an architectural design tool, in an educational setting, and through the physical construction of a 1:1 prototype. During the 9-day workshop, 12 students without prior programming knowledge learned scripting and the logic of the growth algorithm. In groups the students applied the simulations as a design tool. A final 1:1 prototype of 3m height was successfully constructed within 2



3 Global and local forces influencing the resulting geometries: (top) Voxelization, Strata Force, Orthogonal Force; (bottom) Attractor Influences, Undulate Edge Condition, and Serrate Edge Condition

days, with the help of a Microsoft HoloLens, a mixed reality headset processing and overlaying construction information into the assembly process.

RELATED WORK

Diffusion Limited Aggregation, Cellular Automata, Conway's Game of Life

Different grid-based mathematical logics have been used to simulate processes of growth. DLA is today often calculated free-form in 3D space, but was originally conceived as working in 2D grids (Witten and Sander 1981).

Sarkar (2000) explains that CA as a generative design tool has been well documented since von Neuman introduced it in 1963. Especially Conway's Game of Life, a type of CA, has been used to simulate processes of growth (Wolfram 1983, Gardner 1970). Different attempts have been made at utilizing them for architecture and urban design (Al-Qattan, Yan and Galanter 2017, Shiffman 2012, Adilenidou 2015, Kuo and Zausinger 2010).

Although this process can lead to complex geometrical structures derived from simple geometry, Herr and Ford (2007) argue that the transition from a generic CA algorithm to a specific design tool is not very well understood. They argue that CA systems mostly have to undergo multiple adaptations to be able to produce useful outputs in the field architectural design.

In order to overcome the voxelized geometries that result from CA, SmoothLife was developed, a variant that takes into consideration a larger amount of neighboring cells (Rafler 2011, Carroll 2013).

Differential Growth and Cellular Growth

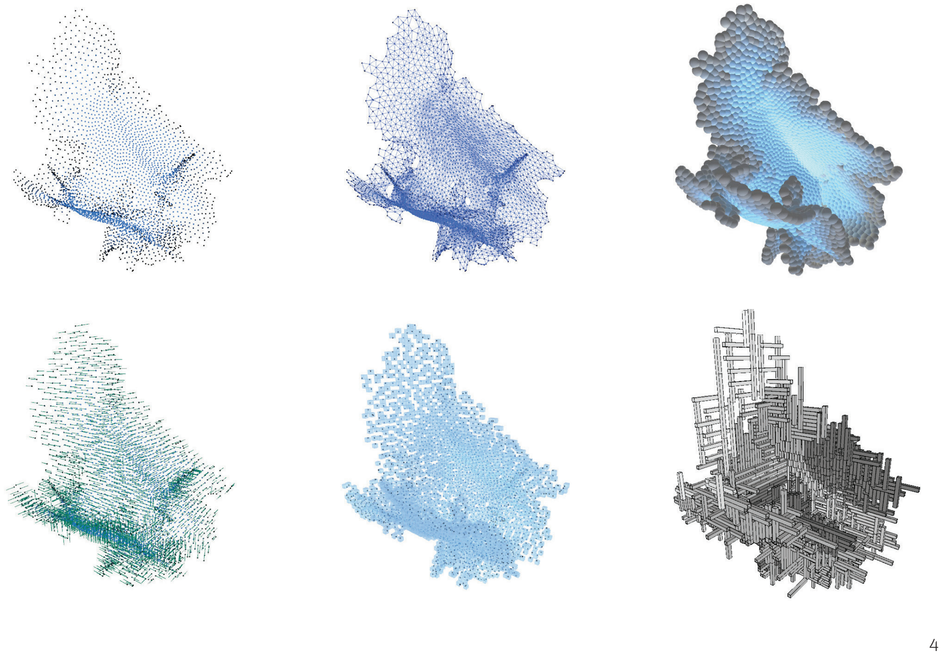
Computational growth simulations based on individual cells are used in developmental biology in order to study the processes that lead to the formation of organisms (Kaandorp et al. 2005, Kaandorp and Kübler 2001, Merks et al 2010, Merks and Glazier 2005, Palm and Merks 2014, Walpole et al. 2013, Wolpert et al. 1998), as well as in cancer research in order to study the growth of tumors (Shirinifard et al. 2009, Milde 2013, Jiao and Torquato 2012, Gevertz and Torquato 2009, Bearer et al. 2009, Neufeld et al. 2013).

In architectural design, Differential Growth simulations have been explored extensively, whereby individual cells can move in 3D space, but are usually arranged as the vertices of polylines or of mesh surfaces. The resulting simulations and free-form geometries have been used for video art or 3D-printed at the small scale (Lomas 2014, Louis-Rosenberg 2015, Bader et al. 2016). A larger installation has been 3D-printed by Alisa Andrasek (Andrasek 2016).

Previous work by the authors utilized a Cellular Growth algorithms, the base algorithm used for the work of this paper, but in a free-form state, to construct an installation from tessellated sheet material (Klemmt and Sugihara 2018).

Discretization for Constructability

The process of discretization as used for the work of this paper, aligns the project within a wider framework of combinatorial and discrete design in architecture. This has been applied to scales ranging from urban design to small scale installations (Kohler 2019, Klemmt et al. 2018).



4 Discretization process and component placement: (top) Cells, Cell Connections, and Solid Volumes; (bottom) Voxelized Normals, Occupied Voxels, and Components

Sanchez (2016) describes this process as a non-parametric approach which focuses on permutation, combination and patterning of discrete units, with part aggregating into larger assemblies, describing meaning, performance and function at different scales of aggregation, leaving the system open-ended, allowing for further expansion and placement of additional parts within.

Retsin (2016a, 2016b) further argues for serial repetition and assembly of discrete parts, based on volume and dissolution of figure, rather than surface and topology typical for parametric projects, emphasizing the importance of part-to-whole relations, and the use of elements which contain a certain kind of design agency, where elements can respond to data such as stress, vector orientation etc.

ALGORITHMIC SIMULATION

Cellular Growth Simulation

The proposed algorithm is based on a point cloud in 3D space, with each cell represented by a point of the point cloud as its center. Iteratively, the cells are evaluated and adjusted by moving their position, and possibly by dividing a cell if certain triggers are met. This leads to an increase of cells over time that take on a geometry based on the various forces that are acting on them, as described in previous papers (Klemmt and Sugihara 2018, Klemmt 2019).

Following an acceleration-velocity model, different forces as vectors are summed up as the cell's acceleration. The acceleration is then added to the cell's velocity, and the velocity is added to the cell's previous position in order to define its new position. A drag factor of commonly 0.5 is multiplied with the velocity to counteract its otherwise continuous increase from adding the acceleration.

The algorithm starts with a small set of cells that are given initially. In each iteration, a cell first evaluates which other cells it regards as its neighbors, based on proximity and a maximum amount of neighbors. This set of neighbors is then used to calculate intercellular forces between the cells (Figure 2).

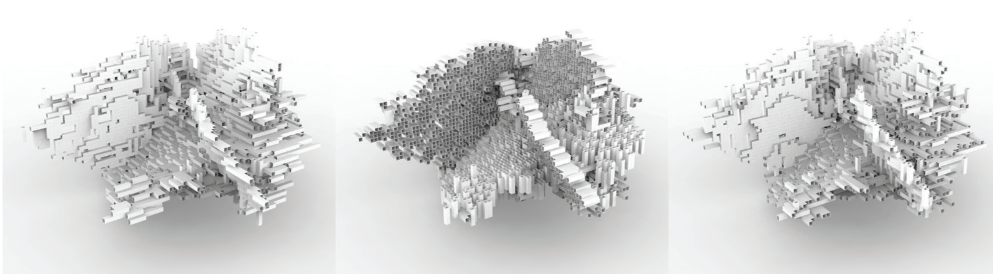
Cell Division

Cells are triggered to divide if they are positioned at the edge of the agglomeration, resulting in a marginal growth of the structure. The cells were identified by their amount of neighbors within a specified range. The cells on the edge will have fewer neighbors than those at the center of the agglomeration. Additionally, cells need to have a minimum age in order to divide.

When a cell divides, a new child cell is inserted close to the parent cell. The child cell inherits the parent's values for various settings, as well as the parent's current neighbors. The age of both parent and child cells are set to 0. Their velocity after the division may be set to move towards or away from their neighbors, resulting in either a smooth or a lobed edge condition.

Cell Forces

In addition to intercellular behaviors, different global forces are acting on the cells causing them to reposition according to their local cell neighborhood. The settings of those



5

5 Different possibilities for component orientation in 1, 2 or 3 axes

6,7 Elevation and close up renders of the proposal, illustrating patterning and sparsification of the components

forces can be used to control the geometric behaviors as well as the overall arrangement of the cell accumulations (Figure 3).

A point force is calculated between direct neighbors, with the aim of keeping those cells at a defined distance from each other. If the cells are closer than their intended distance, the cells will push each other away; if they are further than this distance they will attract each other. The further two cells are from each other, the attraction between them will diminish.

A planarity force pushes neighboring cells to locally align adjacent to each other in planes. The force is calculated by evaluating the plane that passes through the closest neighbors of a cell, and by then pulling the cell towards this plane.

A strata force causes cells to align in planes that have a globally defined orientation. The normal vector of the planes is given, and a cell is pulled along this vector onto the plane that has its origin at the center of the cell's neighbors.

Globally acting forces are based on attractors and imported geometry that influence the behavior of the cells.

Attractors such as points, lines and surfaces can attract or repel the cells. Different strength / distance profiles can be programmed, and the active range of the attractors can be defined.

Imported mesh geometry can act as attractors. The cells will react to the closest point on the mesh, with similar control over the attractor strength as for the fixed point attractors.

Imported mesh geometries can also act as constraints for the cells. The cells in their movement may be constrained to the inside of a solid mesh, they may be prevented from entering the inside of a solid mesh, or they may be constrained to move on the surface of a geometry.

Structural Analysis Integration

Structural Analysis was performed using the Karamba tool plugin for Grasshopper, as presented by Preisinger and Heimrath (2014). The advantage of Karamba is to be directly embedded within a parametric environment, and can therefore calculate and evaluate the respective structural setup in real time.

The code was exported in form of a .txt file consisting of the coordinates of the start and end points of the beams in x, y, z direction. This .txt file was read in Rhino Grasshopper and transformed into a structural beam network in Karamba using native components. After adding the material and cross-section properties (40 x 40 mm spruce beams of square shape), a gravity load was applied and the structural calculations were performed. For the scope of the workshop, an optimized geometrical setup was created by taking into account the utilization and deflection values rendered by Karamba.

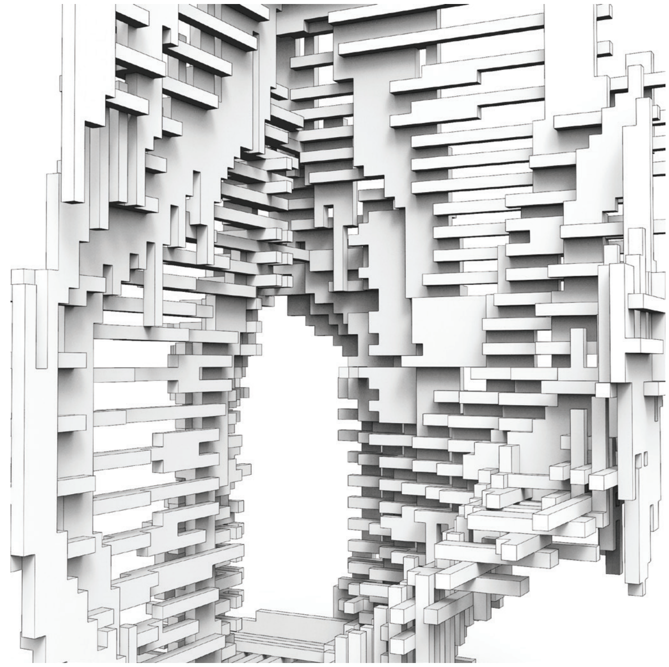
Discretization and Component Placement

The algorithm calculates with cell locations that are free in space, however, the graphic representation of every cell is then discretized onto a regular square grid. In doing so, the underlying algorithm still has the emergent abilities of the freeform calculations, but its output geometry is voxelized.

The voxelization can be applied to the point representation of the cells, whereby those points are repositioned to their nearest location on the grid. However, it is now also possible to draw those voxels as a solid that are occupied by a cell, thereby indicating the volumetric qualities in space that are associated with the cells.

The physical prototype was constructed out of 40 mm x 40 mm timber beams. A set of components used lengths of this material of 320 mm, 480 mm and 640 mm. Therefore, the grid used had an edge length of 40 mm, corresponding to the cross section of the material, and accordingly the set of components occupied either 8, 12, or 16 voxels in a row adjacent to each other.

6



7

For every cell, a component was placed within the grid through the voxel that was occupied by the cell. During the placement, the voxels of the grid that were occupied by the component were marked, and further components were prevented from being placed if they attempted to use an already occupied voxel (Figure 4). Different possibilities were programmed for the orientation of the components, which can align with the X, the Y or the Z axis of the grid. The components can be placed globally to align with one axis or with two alternating axes, or the components can be aligned according to the local normal direction of each cell, to either most closely align with this normal, or to be placed in line within the surface of the adjacent cells (Figure 5).

Lastly, material densities were controlled through a removal of components, dependent on the proximity of a cell and its component from attractor points in space. This results in a rich pattern design language, able to respond to architectural criteria of porosity, lightness, etc. (Figures 6, 7).

PROTOTYPE DESIGN AND CONSTRUCTION

Using Fixed Length as a Measure for Construction Efficiency

The entire construction consisted of 720 overall pieces all equal in width and height (40 mm x 40 mm). Overall there were nine variations in length (320 mm, 480 mm, 640 mm, 800 mm, 960 mm, 1280 mm, 600 mm, 2080 mm, 2240 mm). Wooden beams of the correct cross section were acquired and marked for their intended length. An ordinary jigsaw was used to cut the pieces along their markings. Due to the geometrical simplicity of the individual segments, no additional tools such as CNC machinery was required, and all necessary pieces could be produced in a short time.

The square cross section meant that during the construction process, pieces could not be oriented the wrong way around their length axis. Furthermore, since the offset of two adjacent pieces was always a multiple of the beams cross-section (320 mm, 480 mm, 640 mm), offsets could be easily marked along a segment by any square pieces left over from the initial cutting process.

Materials, Assembly, and Construction Process

Before construction, the overall geometry was digitally split into four main pieces and given to student groups for assembly. The solid-wood components were sorted and distributed accordingly. The first step for each group was to create a panel-like part using leftover cutting blocks as separators for the correct offsets between the individual beams. Beams were connected using ordinary SPAX screws and a regular power drill, with the screw length being just under the thickness of two beams. After the construction of the flat panel, the missing pieces were added on the bottom and top accordingly. Finally after the completion of all four pieces, all parts were erected and held in place using bar clamps before being fixated using SPAX screws.

Microsoft HoloLens and Fologram—Augmented Reality plug-ins for Rhino/Grasshopper—were used to enable students to track and supervise the construction process


8

8 Physical 1:1 Prototype

9, 10 Details of 1:1 prototype

of the four main pieces inside augmented reality (Jahn et al. 2018). Inside the HoloLens the entire part was displayed having beams of different length being represent by different colors. This should ensure that the HoloLens wearing student could quickly instruct his/her peers which beam type had to be placed at what position.

EVALUATION

Architectural Design Possibilities

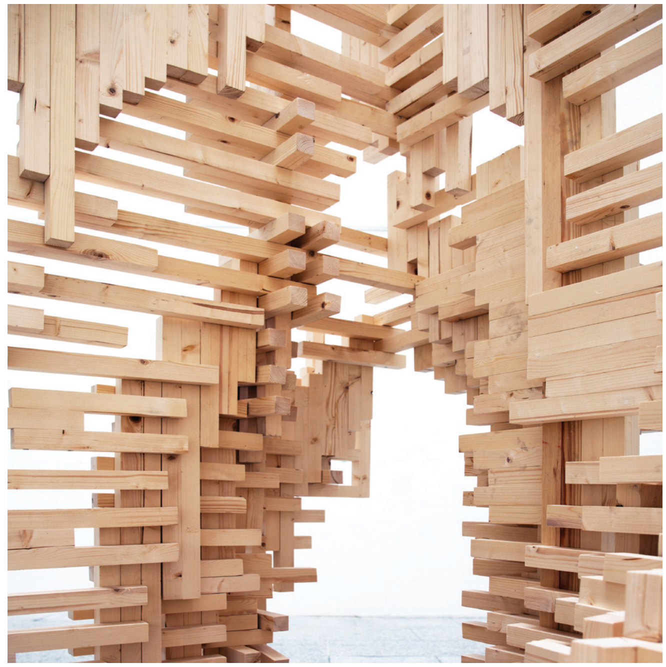
According to Carroll (2013), "Conway's original game supports an impressive variety of structures, but it's not really robust; if you start with a random configuration, chances are good that it will settle down to something boring before too long." Herr and Ford argue that despite earlier studies that portrait CA as generic generative design tools, the transformation from generic CA to specific design tool is not yet well understood, and further that a lack of detailed CA documentation in the field of architectural design has led to a pervasive lack of awareness of such processes among those aiming to apply CA as design tools in architecture (Herr and Ford 2015). While CA and other grid-based growth simulations generate complex geometry based on simple rules, it appears that their outcomes are often unexpected and difficult to control and are therefore mostly unusable for architectural design. The free-form algorithms on the contrary, while still being able to produce complex geometry, provide much more detailed controls over their outcomes.

The presented project and built installation can be understood as a part of a larger architectural system. The process of form generation is essentially two-fold: The overall form is generated through the cellular growth algorithm, which is then discretized for purpose of fabrication. The methodology therefore draws on the strengths of these two systems.

The Cellular Growth of the underlying surface allows for the generation of spatial enclosures of varying levels of complexities, macro-scale porosity and inter-connectivity. The algorithmic setup, which generates surfaces of different orientations through the previously described cellular forces, can be understood as a tool for the generation of primary architectural elements such as slabs or walls.

The process of discretization acts as a second layer of articulation, assigning meaning to function to enclosed spaces, through the creation of different levels of porosity, openness and patterning. As these aggregations are

9



10

understood on a volumetric level, and can to a certain degree depart from the underlying surfaces, the architectural qualities of the generated spaces are not strictly bound to the surface topology. If the introduction of porosity might affect the structural performance of a surface, this can be compensated through the introduction of additional elements.

Finally, the system allows for multi-material assemblies in order to create fully functional spaces.

Educational Possibilities

The educational setup was chosen to follow two aims. On one hand, the workshop aimed to educate architecture students towards a critical use of digital technologies. On the other hand, it intended to introduce fresh architecture graduates to teaching experiences. According to Menges (2011) and Oxman (2006), prospective architecture professionals are required not only to understand the contemporary digital skill set, but also need to be taught the creative potential and design development mechanism of computational design. A specific design development educational environment, as discussed by Gheorghe (2019) is adapted for this workshop, whereas the participants are introduced in a tutorial style format to state-of-the-art digital tools, while simultaneously having the opportunity to develop in groups their own project ideas under the supervision of the instructors. These projects are discussed within the whole group, and one project is selected for further development and full scale fabrication by the complete audience. The students learned the programming of the algorithm and applied it to various design case studies, which were then critically evaluated.

As suggested by Al-Qattan, Yan, and Galanter, computer programming is challenging for users because of their difficulty to comprehend its abstract notion, construct algorithms, and envision of the algorithm application in the real world. In architectural education, programming applications present additional level of complexity for designers. (Al-Qattan, Yan, and Galanter 2017). Further, Herr and Ford encountered difficulties among students to apply CA as design tools in architecture (Herr and Ford 2015). While the students of our workshop did not all learn programming to a degree where they could develop their own algorithms, all students did receive an in depth introduction, and all students were able to adjust the code and apply it successfully to architectural design.

It was the aim to find the right balance between educational freedom and creativity, user controlled input, and semi-random automated process to created sound architectural structure and space. Here, our educational model as applied and described achieves a critical level of production during the week long workshop sufficient for the fabrication of the full scale prototype.

Construction Process

Geometrical features resulting from the use of discrete components led to a successful prototype built within two days. The voxelization process allowed the use of standard wooden beams as building material, thus saving costs and time since no specialized construction tools or methods were necessary. Designing with repetitive elements led to a small amount in the variation of the component length, therefore speeding up the preparation time when it came to cutting the beams to correct size. The square cross-section of the building material, as well as the fact that components could only be placed along the X, Y or Z direction, made it less likely that during construction elements could be placed or rotated along the wrong axis. Due to the underlying grid, the offset between two adjacent pieces was always a multiple of the cross-section, furthermore preventing the construction team from a wrongful assembly of the prototype.

CONCLUSION AND FUTURE WORK

We regard the combination of the free-form Cellular Growth algorithm with the discretization as a successful tool in order to generate highly complex and reactive geometry that is still easily buildable. The free-form surfaces that a growth algorithm generates are still readable at the large scale, but curvature or geometric details at the voxel or component scale are lost. However, the discretization allows for its own further detailing possibilities through the local positioning and orientation of the individual segments.

In future, an aim is to react in real-time to structural input and adapt the geometry directly within the construction process. Hence, the design process can be optimized to allow a setup with a structural feedback loop—whereby the constructor performs the placement action, the digital Karamba interface recalculates the structural condition in real-time, and suggests the next placement position—possibly communicated through an augmented reality environment. In such a manner, a direct link between design development and product construction is enabled via means of digital technologies.

ACKNOWLEDGEMENTS

The project was constructed as part of the AA Visiting School at the Angewandte Vienna 2018. Teaching Assistants: Alexandra Moisi, Nasim Nabavi, Saba Nabavi, Adam Sebestyen.

REFERENCES

Adilenidou, Y. 2015. "Error as Optimization; Using Cellular Automata Systems to Introduce Bias in Aggregation Models through Multigrids." In Real Time; Proceedings of the 33rd eCAADe Conference, Volume 2, edited by B. Martens, G. Wurzer, T. Grasl, W.E. Lorenz, and R. Schaffranek, 601-610.

Al-Qattan, E, W. Yan, and P. Galanter. 2017. "Tangible Computing for Establishing Generative Algorithms; A Case Study with Cellular Automata." In Proceedings of the 35th eCAADe Conference, Volume 1, editedy by A. Fioravanti, S. Cursi, S. Elahmar, S. Gargaro, G. Loffreda, G. Novembri, and A. Trento, 347-354.

Andrasek, A. 2016. "Xenocells: In the Mood for the Unseen." Architural Design 86: 90-95. doi:10.1002/ad.2116.

Bader, C., W.G. Patrick, D. Kolb, S.G. Hays, S. Keating, S. Sharma, D. Dikovsky, B. Belocon, J.C. Weaver, P.A. Silver, and N. Oxman. 2016. "Grown, printed, and biologically augmented: An additively manufactured microfluidic wearable, functionally templated for synthetic microbes." 3D Printing and Additive Manufacturing 3(2): 79-89.

Bearer E.L., J.S. Lowengrub, H.B. Frieboes, Y.L. Chuang, F. Jin, S.M. Wise, M. Ferrari, D.B. Agus, and V. Cristini. 2009. "Multiparameter Computational Modeling of Tumor Invasion." Cancer Research 69 (10): 4493.

Carroll, S. 2013. "Smooth Life." Accessed April 3, 2019. https://www.preposterousuniverse.com/blog/2013/02/20/smooth-life/.

Gardner, M. 1970. "Mathematical Games; The fantastic combinations of John Conway's new solitaire game 'life'." Scientific American 223(4): 120-123.

Gevertz J., and S. Torquato. 2009. "Growing Heterogeneous Tumors in Silico." Phys Rev E Stat Nonlin Soft Matter Phys 80(5 Pt 1): 051910.

Gheorghe, A. 2019. "Digital Design Thinking in Architectural Education; Testing idea-driven and science-driven design processes towards researching polymer-wood composite structures." In Digital Wood Design, Lecture Notes in Civil Engineering 24, edited by F. Bianconi, M. Filippucci. Cham: Springer.

Herr, C.M. and R.C. Ford. 2015. "Adapting cellular automata as architectural design tools." In Emerging Experiences in the Past, Present and Future of Digital Architecture: Proceedings of the 20th CAADRIA Conference, 169-178.

Herr, C. and T. Kvan. 2007. "Adapting cellular automata to support the architectural design process." Automation in Construction 16(1): 61-69.

Jahn, G., C. Newnham, and M. Beanland. 2018. "Making in Mixed Reality; Holographic design, fabrication, assembly and analysis of woven steel structures." In ACADIA 18; Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture, 88-97.

Jiao Y., and S. Torquato. 2012. "Diversity of Dynamics and Morphologies of Invasive Solid Tumors." AIP Advances 2(1): 11003.

Kaandorp, J.A. and J.E. Kübler, eds. 2001. The Algorithmic Beauty of Seaweeds, Sponges and Corals. Heidelberg: Springer.

Kaandorp J.A., P.M.A. Sloot, R.M.H. Merks, R.P.M. Bak, M.J.A. Vermeij, and C. Maier. 2005. "Morphogenesis of the branching reef coral Madracis mirabilis." In Proceedings of the Royal Society B 272(1559): 127-33.

Klemmt, C. and S. Sugihara. 2018. "Growth Structures." In 2016 TxA Emerging Design + Technology Conference Proceedings, edited by Kory Bieg, 108-123.

Klemmt, C., A. Gheorghe, I. Pantic, P. Hornung, and R. Sodhi. 2018. "Engineering Design Tropisms; Utilization of a bamboo-resin joint for voxelized network geometries." In ACADIA 18; Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture, 322-327.

Kohler, D. 2019. "Mereological Thinking: Figuring Realities within Urban Form." Architectural Design 89: 30-37. doi:10.1002/ad.2409.

Kuo, J., and D. Zausinger. 2010. "Scale and Complexity: Multi-layered, multi-scalar agent networks in time-based urban design." In Future Cities: 28th eCAADe Conference Proceedings, 651-657.

Lomas A. 2014. "Cellular Forms: An Artistic Exploration of Morphogenesis." In ACM SIGGRAPH 2014 Studio. ACM, New York.

Louis-Rosenberg, J. 2015. "Floraform; An Exploration Of Differential Growth." Accessed May 17, 2017. http://n-e-r-v-o-u-s.com/blog/?p=6721.

Menges, A., and S. Ahlquist, eds. 2011. Computational Design Thinking. Chichester: Wiley.

Merks R.M.H. and J.A. Glazier. 2005. "A cell-centered approach to developmental biology." Physica A: Statistical Mechanics and Its Applications 352(1): 113-130.

Merks R.M.H., M. Guravage, D. Inzé, and G.T.S. Beemster. 2010. "VirtualLeaf: An Open-Source Framework for Cell-Based Modeling of Plant Tissue Growth and Development." Plant Physiology 155: 656-666.

Milde J.F. 2013. Computational modeling and simulation of cellular growth and migration in angiogenesis. ETH. doi:10.3929/ethz-a-009756864.

Neufeld, E., D. Szczerba, N. Chavannes, and N. Kuster. 2013. "A novel medical image data-based multi-physics simulation platform for computational life sciences." Interface Focus 3: 20120058.

Oxman, R. 2006. "Theory and design in the first digital age." Design Studies 27(3): 229-265.

Palm M.M. and R.M.H. Merks. 2014. "Large-Scale Parameter Studies of Cell-Based Models of Tissue Morphogenesis Using CompuCell3D or VirtualLeaf." In Tissue Morphogenesis Methods in Molecular Biology 1189, 301-322. New York: Humana Press.

Preisinger, C. and M. Heimrath. 2014. "Karamba; A Toolkit for Parametric Structural Design." Structural Engineering International 24(2): 217-221.

Rafler, S. 2011. "Generalization of Conway's 'Game of Life' to a continuous domain, SmoothLife." arXiv:1111.1567 (preprint).

Retsin, G. 2016a. "Discrete Assembly and Digital Materials in Architecture." In Proceedings of eCAADe 2016, 143-151.

Retsin, G. 2016b. "Discrete and Digital; A Discrete paradigm for Design and Production." In 2016 TxA Emerging Design + Technology Conference Proceedings, edited by Kory Bieg.

Sarkar, P. 2000. "A Brief History of Cellular Automata." ACM Computing Surveys 32(1): 80-107.

Sanchez, J. 2016. "Combinatorial design: Non-parametric Computational Design Strategies." In ACADIA 16: Posthuman Frontiers: Data, Designers, and Cognitive Machines; Proceedings of the 36th Annual Conference of the Association for Computer Aided Design in Architecture, 1-13.

Shiffman, D. 2012. The Nature of the Code, 324-330. [S.l.]: D. Shiffman.

Shirinifard A., J.S. Gens, B.L. Zaitlen, N.J. Popławski, M. Swat M, et al. 2009. 3D Multi-Cell Simulation of Tumor Growth and Angiogenesis. PLoS ONE 4(10): e7190. doi:10.1371/journal.pone.0007190

Walpole J., J.A. Papin, and S.M. Peirce. 2013. "Multiscale Computational Models of Complex Biological Systems." Annual Review of Biomedical Engineering 15: 137-154. doi:10.1146/annurev-bioeng-071811-150104.

Witten Jr., T.A. and L.M. Sander. 1981. "Diffusion-Limited Aggregation, A Kinetic Critical Phenomenon." Phys. Rev. Lett. 47: 1400.

Wolfram, S. 1983. "Statistical Mechanics of Cellular Automata." Reviews of Modern Physics 55(3): 601–644. doi:10.1103/RevModPhys.55.601.

Wolpert L., R. Beddington, T. Jessell, P. Lawrence, E. Meyerowitz, and J. Smith. 1998. Principles of Development. Oxford: Oxford University Press. p. 43

IMAGE CREDITS

Christoph Klemmt received his diploma from the Architectural Association in London in 2004. He co-directed the AA Visiting School Vienna, and he is partner at Orproject and Assistant Professor at the University of Cincinnati, where he founded the Architectural Robotics Lab.

Igor Pantic received a Master's Degree from the AA DRL. He is a Teaching Fellow at the UCL Bartlett School of Architecture and was co-director of the AA Visiting School Vienna.

Andrei Gheorghe graduated with distinction from the Harvard Graduate School of Design and received his doctorate from the University of Applied Arts Vienna. He is teaching as an Assistant Professor at the University of Applied Arts Vienna and directs the Angewandte Architecture Challenge.

Adam Sebestyen received his diploma from the University of Applied Arts Vienna, and he teaches at Vienna University of Technology and the New Design University St. Pölten.