



sim AUD 2011

Boston MA USA

2011 Proceedings of the
**Symposium on Simulation for
Architecture and Urban Design**

Edited by
Ramtin Attar

2011 Proceedings of the
**Symposium on Simulation for
Architecture and Urban Design**

Edited by
Ramtin Attar

Cover by **Justin Matejka**
Layout by **Michael Glueck**

Contents

01 **Preface**

03 **Session 1: Materials**

05 **Designing with Deformation - Sketching material and aggregate behaviour of actively deforming structures**

ANDERS HOLDEN DELEURAN, MARTIN TAMKE and METTE RAMSGARD THOMSEN
Center for Information Technology and Architecture, Royal Danish Academy of Fine Arts, School of Architecture

13 **Analysis of Sustainable Manufacturing Using Simulation for Integration of Production and Building Service**

JOHN MICHALOSKI, GUODONG SHAO, JORGE ARINEZ, KEVIN LYONS, SWEE LEONG and FRANK RIDDICK
National Institute of Standards and Technology and General Motors

21 **Session 2: Data Sensing**

23 **Real-Time Occupancy Detection using Decision Trees with Multiple Sensor Types**

EBENEZER HAILEMARIAM, RHYS GOLDSTEIN, RAMTIN ATTAR and AZAM KHAN
Autodesk Research

31 **Sensor Placement Tool for Rapid Development of Video Sensor Layouts**

TYLER GARAAS
Mitsubishi Electronics Research Laboratories

35 **A New System Dynamics Framework for Modeling Behavior of Vehicle Sharing Systems**

DIMITRIS PAPANIKOLAOU
Media Laboratory, Massachusetts Institute of Technology

43 **Session 3: Implementation**

45 **GRAPE: Using graph grammars to implement shape grammars**

THOMAS GRASL and ATHANASSIOS ECONOMOU
SWAP [+] and Georgia Institute of Technology

53 **Automated Energy Model Creation for Conceptual Design**

LILLIAN SMITH, KYLE BERNHARDT and MATTHEW JEZYK
Autodesk

61 **An Integrated Approach to Algorithmic Design and Environmental Analysis**

ROBERT AISH and ANDREW MARSH
Autodesk

69 **Leveraging Cloud Computing and High Performance Computing Advances for Next-generation Architecture and Urban Design Projects**

FRANCESCO IORIO and JANE L. SNOWDON
Autodesk Research and IBM T. J. Watson Research Center

77 **Session 4: Generative Design**

79 **Generative Fluid Dynamics: Integration of Fast Fluid Dynamics and Genetic Algorithms for wind loading optimization of a free form surface**

OUTSTANDING PAPER AWARD

ANGELOS CHRONIS, ALASDAIR TURNER and MARTHA TSIGKARI
University College London

87 **Use of Sub-division Surfaces Architectural Form-Finding and Procedural Modelling**

SHAJAY BHOOSHAN and MOSTAFA EL SAYED
Computation and Design Group and Zaha Hadid Architects

95 **Irregular Vertex Editing for Architectural Geometry Design**

YOSHIHIRO KOBAYASHI and PETER WONKA
Arizona State University

103 **Session 5: Design & Analysis**

105 **Solar Zoning and Energy in Detached Residential Dwellings**

JEFFREY NIEMASZ, JON SARGENT and CHRISTOPH REINHART
Graduate School of Design, Harvard University



BEST PAPER AWARD

115 **A simple method to consider energy balance in the architectural design of residential buildings**

LAËTITIA ARANTES, OLIVIER BAVEREL, PASCAL ROLLET and DANIEL QUENARD
Ecole Nationale Supérieure d'Architecture de Grenoble, Centre Scientifique et Technique du Bâtiment and École des Ponts Paris-Tech

123 **A methodological study of environmental simulation in architecture and engineering. Integrating daylight and thermal performance across the urban and building scales.**

PETER ANDREAS SATTRUP and JAKOB STRØMANN-ANDERSEN
School of Architecture, Royal Danish Academy of Fine Arts and Technical University of Denmark

133 **Session 6: Augmented Reality**

135 **Lifecycle Building Card: Toward Paperless and Visual Lifecycle Management Tools**

HOLGER GRAF, SOUHEIL SOUBRA, GUILLAUME PICINBONO, IAN KEOUGH, ALEX TESSIER and AZAM KHAN
Fraunhofer IGD, CSTB, Buro Happold and Autodesk Research

143 **3D Scans of As-Built Street Scenes for Virtual Environments**

NAAI-JUNG SHIH, CHIA-YU LEE and TZU-YING CHAN
National Taiwan University of Science and Technology

149 **Visualizing Urban Systems: Revealing City Infrastructures**

CHRIS KRONER, PHU DUONG, LIZ BARRY and MIKE SZIVOS
Graduate School of Architecture, Preservation and Planning, Columbia University

153 **Session 7: Parametric Urbanism**

155 **City of Love and Hate**

ADNAN IHSAN, AMIRALI MERATI, EVANGELIA POULOPOULOU and FOTEINOS SOULOS
Columbia University

163 **Components for Parametric Urban Design in Grasshopper. From Street Network to Building Geometry**

CHRISTIAN SCHNEIDER, ANASTASIA KOLTSOVA and GERHARD SCHMITT
ETH Zurich

171 **Multi-Objective Optimization in Urban Design**

MICHELE BRUNO, KERRI HENDERSON and HONG MIN KIM
Columbia University

179 **Presenting Author Biographies**

184 **Organizers**

185 **Sponsors**

186 **Cover Image Credits**

187 **Index of Authors**

Preface

Ramtin Attar

Autodesk Research

Our built environment is the site where our incremental understanding of the world and natural phenomenon leads to techniques of creation and modals of construction; a site where our creative capacity in utilizing our tools and techniques manifest itself into a world we inhabit. In an era driven by our ambition for environmental reform and green design, then we have to question the capacity of our existing methods in helping us to achieve deeper insight into the properties of complex systems such as our built environment. Using simulation methods to study complex processes and interactions is fairly well positioned in the history of technological innovation. However, at this moment of history, the unforeseen behavior of our built environment is grabbing more forcefully our attention. The role of simulation either as a predictive model, providing us a slice of reality, or as a prescriptive model, allowing us to tweak and explore alternative realities, is already widely explored in physics, biology, economics and engineering. Simulation is also a “game changer” in how we will fundamentally rationalize and conceptualize the design of our built environment. The future availability of massive data simulated through a sea of distributed processors combined with the data collected from an instrumented environment are only few exemplary paradigms that will challenge our conventional assumptions about the built environment in the near future.

Moving forward, the key is to push our future developments beyond single-pointed technological solutions by conceiving complex causal relationships related to the architecture and methods of thinking and collaboration. Our future success is deeply rooted in our predisposition and capacity to shape an integrative thinking and approach towards the design of our built environment. Modern technological systems are so complex and interdisciplinary groups are needed to share perspective and information in order to create and control these systems.

Unlike other research conferences and symposia that merely provide a venue for publishing research on a particular topic, SimAUD will work to build a closer community for collaborative work and co-authoring of research that envisions the future of the built environment. This year marks the second year of SimAUD and it is indicative of a growing interest in this exciting venue. I would like to thank Azam Khan for his ground work in establishing SimAUD and his continuous support during this year’s event. I am grateful to Michael Glueck for all his work in designing and building the website and the proceedings, and Rhys Goldstein for his continuous input throughout the process. I would also like to thank Gord Kurtenbach and Jeff Kowalski of Autodesk for their continuous support and encouragement. At last but not the least, I thank our reviewing committee for their diligent work and dedication to assure that SimAUD will continue to publish high quality research.

Ramtin Attar

SimAUD 2011 Chair

All accepted papers will be published in the ACM Digital Library at the SpringSim Archive.
Sponsored by The Society for Modeling and Simulation International.

Session 1: Materials

- 05 **Designing with Deformation - Sketching material and aggregate behaviour of actively deforming structures**

ANDERS HOLDEN DELEURAN

Royal Danish Academy of Fine Arts, School of Architecture

- 13 **Analysis of Sustainable Manufacturing Using Simulation for Integration of Production and Building Service**

JOHN MICHALOSKI, JORGE ARINEZ, GUODONG SHAO, SWEE LEONG, KEVIN LYONS and FRANK RIDDICK

National Institute of Standards and Technology (NIST)

Designing with Deformation - Sketching material and aggregate behaviour of actively deforming structures

Anders Holden Deleuran, Martin Tamke and Mette Ramsgard Thomsen

CITA - Center for Information Technology and Architecture,
Royal Danish Academy of Fine Arts, School of Architecture
Philip de Langes Allé 10, 1435 Copenhagen, Denmark
andersholden.deleuran@karch.dk, martin.tamke@karch.dk, mette.thomsen@karch.dk

Keywords: Architecture, Complex Modelling, Dynamics Simulation, Material Behaviour, Active Deformation, Aggregate Structures.

Abstract

The recent development of material performance as a key driver of architectural design is currently challenging the role of representation and prototyping. This paper shares findings from a research project exploring the potential of a *digital-material prototype* capable of addressing this challenge. The project examines the possibility of incorporating material properties into digital models using respectively an *analytical* and a *dynamics-based approach*. The paper will present three design experiments with different material properties all attempting to deliberately embrace deformation as a key principle of design. This exploration of *actively deforming structures* is carried out using *light weight dynamics simulation* producing flexible and intuitive models for sketching material behaviour in the early design stages.

1. INTRODUCTION

Architectural constructions are traditionally conceived as rigid structures that are static in their behaviour. When considering the material performance of a brick wall or a concrete slab we imagine these as inherently fixed in time and space. However, as the project passes from design to production the parallel knowledge field of structural engineering is focused on the calculation and control of the dynamic qualities of these materials. Here, the performative role the built environment is foregrounded aiming to optimise for maximum stiffness with least deformation and to limit the bending, buckling or twisting of materials. Traditionally these two knowledge fields have been clearly separated, one succeeding the other, but as the tools of architectural design change new opportunities for designing for and with material behaviour appear.

The introduction of complex 3D modelling allows architects to engage with material simulation. Traditional architectural tools such as CAD based drafting support the static perception of our surroundings. Architectural drawing prioritises the geometry of the building proposal, delineating extension while giving little significance to the structural tensions and material behaviours of torsion and bending that take place within the structure. With the introduction of computation, 3D modelling and the ability to program models parametrically the direct simulation of material behaviour becomes possible. Working deliberately with material deformation, architects are able to investigate material performance as an integrated part of design.

This paper reports on three projects undertaken as part of a research focus on integrating material performance in architectural design. The projects examine a range of different materials moving from the mono-material to the composite. Our aim with the work presented in this paper is to investigate how material simulation can be used to explore material behaviour within the early design phase both as an analytical tool as well as a design tool. Exploring “light weight” tools for dynamic simulation and their interfacing with architectural design tools our objective has been to understand how simulation can lead to the design of new material structures.

2. RESEARCH INQUIRY

The architectural design space is traditionally defined by the hand drafting and modelling of non-associative geometry. The integration of parametric tools has advanced this by enabling data-driven “smart” geometry generated through a process of defining relationships in an algorithmic manner (Kolarevic and Malkawi 2005). Although being highly flexible these new tools still prioritise the perception of structures as being materially static and rigid. In this

design space deformation is therefore typically considered as a *passive* parameter. Geometric description and structural constraints are therefore in general manageable. This is not the case with *actively deforming structures*. These rely on a more complex set of spatial logics which fundamentally challenge the design space. When embracing *active deformation* the physical constraints which govern material reality need to be applied throughout the design process and therefore geometry may no longer be idealized as rigid.

As a consequence of this design involving actively deforming structures in an architectural context has very little precedent. Research within this field is therefore often informed by other fields such as garment making (McQuaid 2005), biology (Lienhard et al. 2010) and ship building (Lindsey 2001). In contemporary architectural research this field is rapidly maturing through an increased focus on the relationship between material strategy, digital representation and digital fabrication techniques. A recent example of this can be seen in the ICD/ITKE pavilion developed by a team directed by Achim Menges at the University of Stuttgart (Menges 2010). It convincingly displays how computational design, material simulation and robotic production processes can result in a novel bending-active structure made entirely of elastically bent plywood strips (Kaltenbach 2010). Other contemporary examples of this thinking can be seen in the work and literature of Michael Hensel (Hensel and Menges 2006), Mark and Jane Burry (Burry and Burry 2010) and Gramazio and Kohler (Gramazio and Kohler 2008)

Although such projects are proving highly innovative most work is generally still being carried out through the making of physical prototypes. Often the problem in projects that aim to integrate material behaviour such as bending or stretching is that the digital representations are limited in their scope for abstracting or simulating such behaviour. In these projects physical prototyping becomes a way of empirically testing concepts without directly employing digital representation (Tamke et al. 2009). The material prototype as a mode of design therefore engages deformation through the medium of physical representation i.e. scale models and full size mock-ups. This approach suffers several drawbacks being costly and time-consuming, lacking sufficient scalability and being very limited in their adjustability. The described shortcomings suggests that a medium is needed which bridges the gap between representation and reality more efficiently in order to better engage the domain of actively deforming structures.

3. THE DIGITAL-MATERIAL PROTOTYPE

The following describes our initial efforts to define and elaborate such a medium through the concept of a *digital-material prototype*. This prototype concept embeds material behaviour into the digital model and may potentially overcome the drawbacks of physical modelling. In this work the initial research is focused on conceptual modelling, looking at simulation as an early design tool and not only a tool for validation and analysis. Our primary goal is to develop a framework capable of producing flexible and intuitive models which may be employed in a process of *sketching material behaviour* in complex deforming structures. This goal necessitates the possibility of generating models capable of simulating various different materials in an interactive and intuitive manner, so as to quickly iterate ideas and concepts. In the following we will present and discuss two distinct approaches we have developed for digital-material prototyping. In the first approach we use empirical data as a basis for informing a parametric model simulating bending deformation. In the second approach we address the use of dynamics based physics solvers for simulating a more broad range of material behaviours, this approach is exemplified through three specific design experiments.

3.1. An Analytical Approach

Abstracting and embedding material properties is inherently problematic. We have conducted experiments aimed at informing digital models by analysing the behaviour of different materials and plotting the resulting data into the model using parametric software such as Grasshopper and Generative Components. An applied case of this approach is the installation Thaw. Thaw investigates the use of textile concepts for tectonic structures of an architectural scale. It is designed by CITA's Mette Ramsgard Thomsen and Karin Bech for the "Digital Material" exhibition at the ROM gallery in Oslo. Thaw is a five meter tall woven wall membrane constructed of ash slats braced by steel joints. As a material structure it explores active deformation through the elastic bending and twisting of its wooden members as well as the friction based interlocking of the weave. For the design and construction of Thaw a geometrically relational parametric model was built. Following this the bending deformation of varying member lengths were analysed under a range of load conditions (figure 1. right). Subsequently the produced data set was abstracted into formulas which could simulate the "bend" of members using a law curve in the parametric

model (figure 1. left). This gave us a general idea of how the wood would deform in the structure, generating sufficiently accurate geometric data to use for the fabrication of a “geometry dependent” skin system attached to the wooden slats. The design model also supported real-time feedback which was valuable in quickly iterating proposals and making design decisions.

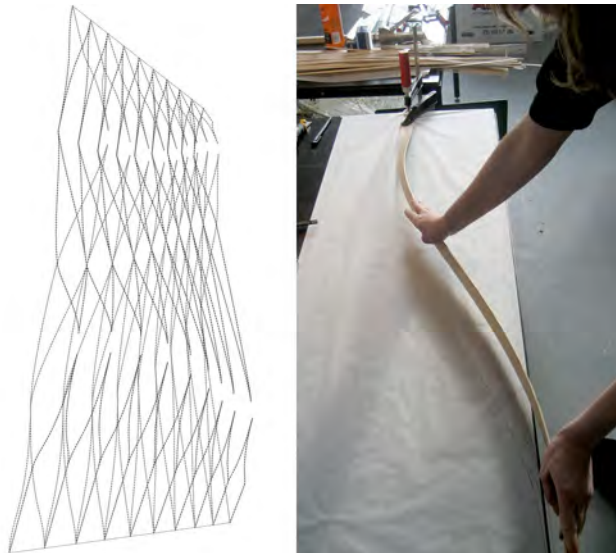


Figure 1. Left: The analytical bend simulation. Right: Plotting the bending deformation which produces the data set driving the law curve.

From this experimentation we learned that this analytical and empirical approach provides highly valuable data in understanding the material behaviour. The direct translation of this data into a function-based simulation has however exposed several drawbacks in representing material structures beyond that of the singular member. A model based on the isolated deformation of a single member does not account for the collective physical interdependency in the overall structure. The load-conditions applied in the process of analysing the singular component will furthermore not correspond to the actual distributed load condition for any given member in the structure. By comparing the analytical bend simulation of Thaw to the behaviour of the actual built structure (figure 3. right) these drawbacks are clearly seen.

This approach of embedding material behaviour thus result in models where the part-to-whole relationship does not include the distributed material behaviour and deformation of the system as a whole, what we refer to as the *aggregate behaviour* of a material structure. Due to the intrinsic nature of the analytical model it furthermore lacks

the ability to incorporate external forces acting upon the structure such as gravity, wind or impact. This ability is crucial in order to meaningfully embed a property such as mass, which is essential, as the compound mass comprising the deadload of a structure greatly influences its aggregate behaviour. When working with active deformation and employing highly deformable materials this holds especially true as these display less resistance towards deforming forces than their rigid counterparts.

From these findings it can be concluded that a more holistic, exhaustive and flexible method is required in order to qualify as an apt digital-material prototype. The shortcomings of the analytical approach suggest that a decentralized approach - directly embedding material properties and physics based constraints into digital matter and the environment which it inhabits - might prove better suited. In the following this assumption is explored and elaborated upon.

3.2. A Dynamics-Based Approach

The dynamics-based approach explores the potential of integrating physics simulation in our digital models. This involves implementing a physics solver which can provide an approximate simulation by explicitly defining material properties. This provides a decentralized approach with full collective physical interdependency and the possibility of integrating participating external forces. In current explorations on simulation for generative design this approach is supported by research at Autodesk which suggest that “...physics-based generative design represents a paradigm shift from the traditional primacy of object to an exploratory approach of investigating interacting elements, interdependencies and systems. The integration of simulation opens up the possibilities for a more dynamic framework in the early stages of design” (Attar et al. 2009). Consequently such systems provide better support for the needed requirements of flexibility, scalability and distributed behaviour. The following presents an investigation into the involved factors of identifying specific simulation tools and techniques applicable to this agenda.

4. IDENTIFYING A DYNAMIC DESIGN SPACE

The description of material behaviour is an extensive subject of study with many distinct sub-fields ranging from elastic to plastic deformation characteristics, to damage and fracture mechanics, to fluid dynamics of gaseous and liquid media. As a consequence numerous specific approaches for

simulating material behaviour exist within many fields. These can be divided into four overall fields of employment: gaming, film/animation, engineering and natural science.

Film and animation is concerned with the simulation of physical phenomena with the main issues being stability, robustness and speed. Gaming has similar goals but with an increased focus on real-time performance and interaction. These fields are in contrast to engineering and the natural sciences which operate at a wider scope of simulation where the main focus is on accuracy, extendibility and reliability (Müller et al. 2006). Generally speaking this means that the described spectrum of applications can be perceived as going from *light weight* to *heavy weight* in terms of computational overhead and ease of use. The sum of these applications furthermore spans a vast amount of specialized singular tools as well as more flexible and sophisticated multi-physics solutions. This makes the subject extensive and more or less inaccessible for designers and architects. Determining an appropriate dynamics solver may therefore be problematic without specified requirements combined with sufficient research on the subject.

4.1. Light Weight Dynamics Simulation

Our goals for the use of simulation as a digital-material prototype disqualifies computationally intensive and less intuitive applications commonly used for engineering purposes and analysis such as Ansys, Sofistik or Comsol. Such applications do furthermore not interface particularly smoothly with 3D modellers commonly used in the early design stages by designers and architects such as Rhino 3D, 3ds Max, SketchUp or Maya. These factors point towards the branch of dynamics simulation related to computer graphics and the gaming industry. Researchers within these fields have historically developed dynamics simulation solvers for structured deformable objects aimed at interactive frame rates (Desbrun et al 1999, Terzopoulos 1987). Many of these integrated into 3D packages already familiar to designers and architects. Consequently we have focused on exploring the potential and applicability of such solvers which may generally be described as *light weight dynamics simulation*.

Precedent in the field shows that architects and designers have previously explored this subject in the pursuit of more intuitive models for structural and environmental analysis as well as the aforementioned generative design processes (Zarzycki 2010). This exploration has been driven by

dynamics solvers utilized “as is” in commercial 3D animation packages or by writing bespoke software. Common to these dynamics solvers is the approach to model objects as a system of point masses (particles) connected by elastic springs or constraints (Ahlquist and Menges 2010). This method provides a relatively simple method for simulating deformable objects exhibiting realistic physics at interactive frame rates on personal computers.

Although lacking accuracy and being geared towards visual results “...particle systems are related to computer-based structural analysis methods used in the design of buildings, where the springs of the particle system correspond to finite elements used to model structural members” (Martini 2001). Despite this promising parallel the implementation of light weight dynamics simulation should not yet be adopted as a substitute for accurate finite element analysis. Instead it should be used as a complementary method for initial conceptualization, shortcutting the process by embedding similar tools in the design phase where the simulated structure can respond to the designers input in real time. This enables intuitive and immediate changes to the system, thereby increasing both structural understanding and the ability to iterate design concepts.

4.2. A Unified Design Space

Our design development of new material structures is based on experimentation with a diverse and often interacting range of distinct materials. The usability of our digital-material prototypes is consequently dependent on the implementation of dynamics solvers capable of simulating multiple interacting materials simultaneously in a unified manor. Traditionally light weight dynamics solvers have been designed to simulate particular types of objects such as rigid bodies, cloth or hair, thereby making combinations of such effects problematic due the transference of data from one solver to the other (Stam 2009). To resolve this and advance the flexibility of physics simulation promising developments towards unified and extendable solvers are currently being pursued. Based on unified generalizations of matter and forces, these solvers are capable of computing multiple physical phenomena interacting simultaneously, thereby supporting the identified requirements for a digital-material prototype.

Within computer graphics and animation these include the Nucleus solver, implemented in Autodesk Maya since

version 8.5, and the Lagoa Multi-Physics solver, released for Autodesk Softimage 2011. Interestingly, similar physics solvers are being developed for both parametric CAD modellers and text-based programming environments, including the Kangaroo Physics plug-in, available for Grasshopper in Rhino and Generative Components, and the Traer Physics Library, available for the open source programming environment Processing. Due to familiarity with Maya and the relative maturity and production proven nature of the Nucleus solver we have so far employed these as the initial toolset for design experimentation and elaboration of the proposed framework. This experimentation will now be exemplified through distinct applications of the proposed approach in the context of three concrete projects.

5. DESIGN EXPERIMENTS

5.1. Thaw: Weave and friction based tectonics

The first experiment presents a revisit to the installation Thaw. In order to confirm the assumption that the dynamics-based approach can provide a more accurate simulation of the actual behaviour than the analytical we built a new digital model using Maya/Nucleus. To inform this model we set up a calibration rig. This consisted of ten slats of varying lengths locked in one end deforming under their own weight due to gravity. By measuring and correlating this deformation with corresponding nMeshes (a polygon mesh with embedded material properties in the digital model) we tuned the parameters of the system and mesh resolution to a level where a relatively approximate behaviour was simulated (Figure 2).

From here the existing relational Grasshopper model was modified to generate meshes corresponding to the slats used in the construction while simultaneously adhering to the established mesh resolution. These were exported to Maya and converted to nMeshes. To create the meshes for the steel joints a MEL (Maya Embedded Language) script was written which can generate a joint based on selecting the two opposing faces of the slats to join and the angle at which they meet.



Figure 2. Sideview of the simulated calibration rig in Maya.



Figure 3. Left: Equilibrium state of the dynamics-based model in Maya. Right: Photo of the built structure with fabric skin.

With the nMeshes in place the question was how to bend, twist and attach the pieces together. The first step here was to lock the end conditions at the top and bottom part of installation. Following this a considerable number of “attraction” constraints, linking one mesh vertex to another, were created and key-framed to pull the slats and joints together over time. To weave the slats in the correct manor each joint was loosely constrained to its world position thereby forcing the slats to attract outwards. By key-framing this order of events the process of weaving could be managed very precisely. In this key-framed timeline the slats and joints are connected in three primary stages, mirroring the real process, before finally reaching a stage of equilibrium in the final step shown here (Figure 4). By comparing the results of the dynamic model with the actual built structure (figure 3) it is seen that the dynamic model successfully manages to approximate the complex aggregate behaviour of the built structure very closely.

From the experiment we learned that the specific order of physical events in time is a highly important factor when simulating a complex structure such as Thaw. Due to the non-parameterized nature of polygon meshes the constraint creation could not be automated. The process was therefore remarkably similar to the tedious task of manually doing it by hand in real life. Beyond the problematic nature of setting up the constraints we also encountered issues with

making the slats stiff enough without having to use settings that seriously affected the frame rate. The behaviour of the model is therefore less rigid than the real installation.

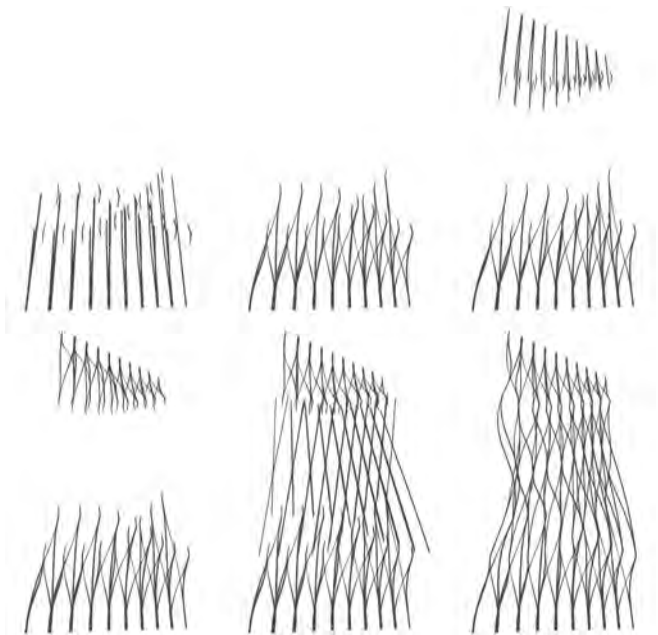


Figure 4. Screenshots from the dynamic model over time. The process of weaving the slats and connecting them to the joints.

5.2. Reef: Minimal surfaces as a method of actuation

The second experiment explores the simulation of smart composite materials. In the responsive ceiling installation Reef CITA’s Aurélie Mossé and physicist Dr. Gugli Kofod are collaborating on the use of “electro-mechanically active polymers” to create self-organizing responsive components with embedded actuation. As a material structure these components are essentially minimal surfaces bounded by an elastically deformable frame. A pre-stretched elastomer sheet is attached to a thin laser cut plastic frame resulting in complex out-of-plane composite structures. The shape change occurs due to the elastomer attempting to minimize its surface, thereby pulling the frame and forcing it to bend out of plane. The elastomer has the ability to reverse this by running an electrical current through it. This enables the shape change to go from one state to the other based on sensor input. The component can thereby become responsive.

A primary challenge in this work is how to anticipate the self-organized shape of the component and thereby the design. A secondary challenge is how the overall performance of distributed components in the Reef may behave as a responsive environment. Through simulation we

have developed a workflow which can anticipate behaviour in both these challenges.



Figure 5. Top: Mock-up of a component for the Reef ceiling installation. Bottom: A corresponding dynamics-based model with the elastomer undergoing surface minimization.

To simulate the material behaviour we inform the model using a similar approach of calibration as with Thaw. From here a CAD-file describing the planar shape of the frame is translated into a polygonal mesh. To ensure good mesh topology this is done using conventional poly-modelling (figure 6). The mesh is then duplicated and modified so that one mesh represents the frame and the other the elastomer. After converting to nMeshes the two are aligned and a constraint is applied which essentially “glues” the two together at their nearest vertices. The model is now in the planar state where the elastomer is active. To turn it off an attribute controlling the “rest length scale” of the mesh is used. By sliding this below its initial value the elastomer will begin to contract thereby simulating the behaviour of surface minimization and changing the overall shape of the component (figure 5. bottom).

Despite the rather liberal mesh interpretation of the frame shape it can be concluded that the dynamic model can anticipate the self-organized shape of the real component to a satisfactory degree. This has been further confirmed by simulating the behaviour of other components with dissimilar frame shapes. As with Thaw we have encountered problems with making the plastic frame stiff enough at an

acceptable frame rate. Due to the complex shapes used the biggest hurdle in the workflow has however been the translation from CAD-file to polygon mesh.

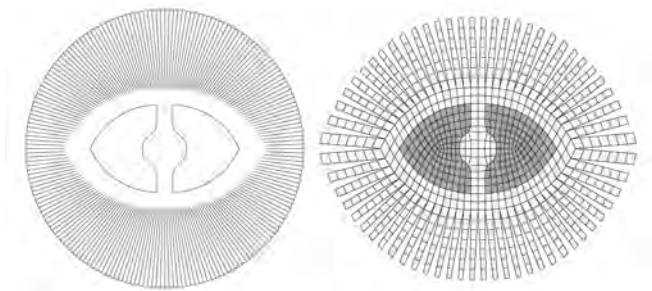


Figure 6. Left: CAD-file of laser-cut shape. Right: The translated polygon mesh divided into the frame (white) and the elastomer (grey).

To simulate the behaviour of the installation as a responsive environment we have distributed components into a “Reef”. From here the individual component actuation and resulting collective behaviour can be simulated using various parametric animation techniques such as “driven keys” and “expressions” using parameters such as proximity or wind direction as input to which the installation responds. This has proven a valuable sketching tool in the design process of the installation.

5.3. Project Distortion: Dynamics as a design tool

The final experiment explores the use of dynamics as a form finding tool in the design of a pavilion for the 2010 Distortion street festival in Copenhagen. It is the result of collaboration between CITA’s Martin Tamke and Brady Peters, students and supervisors from the Royal Academy of Fine Arts School of Architecture in Copenhagen and the Rensselaer Polytechnic Institute in New York. The pavilion is a configuration of connected equilateral cones constructed from acoustic foam and plywood joined using cable ties. Experiments with physical scale models revealed that the surface of connected equilateral cones acted like hinged triangles. This resulted in an overall textile behaviour where the surface is pliable, finding its form through pleating and wrinkling. Where these scale models helped to figure out a design we needed a more precise model to provide the data needed for the fabrication of the pavilion on CNC machinery. At this point we turned to dynamics-based simulation as a potential solution.

The input for the dynamic model was an array of cones generated through a Grasshopper definition which can propagate eight types of cones into an array based on

parameters such as aperture, height, colour and the outline the array. This array is converted to polygons and exported to Maya (figure 8). Here it is split in two and aligned to face each other. Since this form finding method is a geometric exercise informed solely by mass, gravity and collision the cones can in this case be idealized as rigid. To attach the cones to each other a constraint locking the overlapping vertices of the meshes is created. Two adjustable guide rails defining the desired footprint of the pavilion are created using inverse kinematic chains. To pull the array back together and simultaneously attract the base cones to targets on the guide rails a number of constraints using the same “attraction” technique as with Thaw is created. Finally the gravity on the solver is key-framed. While the geometry is being pulled into place gravity is set to zero. As the cones connect gravity is put in reverse for a further form finding process following the Gaudi inspired paradigm of inverted catenary chains. Once the system reaches equilibrium the form finding process is complete with a structure that inherits mainly compressive forces.

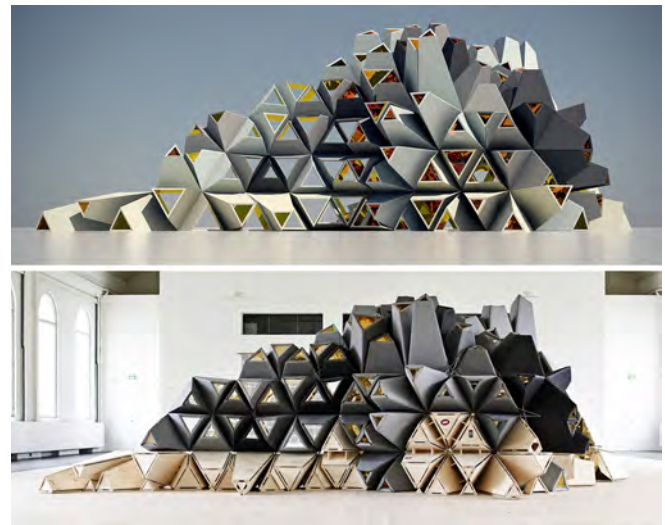


Figure 7. Distortion Pavilion. Top: Render of final form found configuration. Bottom: Photo of the built structure.

In the built structure the aggregation of cones ultimately becomes structural through bespoke steel braces fixing the cones in a staggered hexagon pattern across the arch of the pavilion. Therefore the form found geometry was imported back into Rhino where a second Grasshopper definition computes the angles between the cones and drives the fabrication of the braces. The purpose of the model thus becomes twofold: part form finding and part fabrication data. As figure 7 demonstrates this experimental design and fabrication method was successfully confirmed.

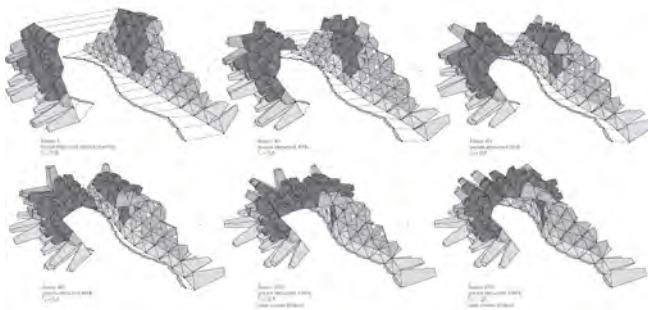


Figure 8. The array of cones from the Grasshopper definition translated to a dynamic Maya model and the process of form finding.

6. CONCLUSION:

We have established that the design of material structures based on active deformation may benefit from new design tools capable of embedded material behaviour. Through the concept of the digital-material prototype we have identified *unified light weight dynamics solvers* interfacing with existing CAD tools as a potential design space for achieving this goal. Finally we have validated this assumption through three successful design experiments using dynamics-based simulation.

The presented work has mainly focused on predicting the behaviour of a material system before it is built. In this process it has been the design intent which has informed the tool. The Distortion project conversely hints at a parallel application for dynamics-based simulation in which it is the tool that informs the design in a process of generative form finding. It can be imagined that combining these two levels of information by feeding them into each other might result in novel self-designing systems. In our future work we aim to explore this aspect more extensively.

The design experiments also exposed specific problems with physics-based simulation. The computational overhead can quickly result in the loss of interactivity and the polygon based workflow requires high-quality meshes and lacks sufficient parametric control. It can therefore be concluded that neither the analytical nor the dynamics-based approach can fully satisfy the requirements for a digital-material prototype as defined in this paper. In our further research we therefore aim to experiment with the development of new tools and workflows which can combine the advantages of both.

References

Ahlquist, S. and Menges, A., 2010. Realizing Formal and Functional Complexity for Structurally Dynamic Systems in Rapid Computational

Means. Proceedings of the Advances in Architectural Geometry conference 2010

Attar, R., Aish, R., Stam, J., Brinsmead, D., Tessier, A., Glueck, M., and Khan, A. 2009. Physics-Based Generative Design. CAAD Futures Conference 2009.

Burphy, M. and Burphy, J. 2010. The New Mathematics of Architecture. Thames & Hudson 2010

Desbrun, M., Schröder, P., and Barr, A. 1999. Interactive animation of structured deformable objects. GI 1999

Gramazio, F. and Kohler, M. 2008. Digital Materiality in Architecture. Lars Müller Publishers 2008

Hensel, M. and Menges, A. 2006. Morpho-Ecologies. AA Publications 2008

Kaltenbach, F. 2010. Teaching by Design - A Research Pavilion in Stuttgart, Detail Magazine - Review of Architecture, Serie 2010, 10 Bauen mit Holz. 2010

Kolarevic, B. and Malkawi, A. 2005. Performative Architecture – Beyond Instrumentality. Spoon Press 2005

Lienhard, J., Schleicher, S., and Knippers, J. 2010. Form-finding of nature inspired kinematics for pliable structures. IASS symposium 2010.

Lindsey, B. 2001. Digital Gehry: Material Resistance Digital Construction. Birkhäuser Basel 2001

Martini, K. 2001. Non-linear Structural Analysis as Real Time Animation - Borrowing from the Arcade. Acadia 2001

McQuaid, M. 2005. Extreme Textiles: Designing for High Performance. Princeton Architectural Press

Menges, A. 2010. Material Information: Integrating Material Characteristics and Behaviour in Computational Design for Performative Wood Construction. ACADIA 2010.

Müller, M., Heidelberger, B., Hennix, M., and Ratcliff, J. 2006. Position Based Dynamics. 3rd Workshop in Virtual Reality Interactions and Physical Simulation 2006

Stam, J. 2009. Nucleus: Towards a Unified Dynamics Solver for Computer Graphics. IEEE International Conference on Computer-Aided Design and Computer Graphics 2009

Tamke, M., Thomsen, M.R., Asut, S., and Joseffson, K. 2009. Translating Material and Design Space - Strategies to Design with Curved Creased Surfaces. ECAAD27.

Terzopoulos, D., Platt, J., Barr, A., and Fleischer, K. 1987. Elastically Deformable Models. Computer Graphics, Volume 21, Number 4 1987

Zarzycki, Andrzej. 2010. Intuitive Structures: Applications of Dynamic Simulations in Early Design Stages. SimAUD 2010.

Analysis of Sustainable Manufacturing Using Simulation for Integration of Production and Building Service

John L. Michaloski¹, Guodong Shao¹, Jorge Arinez², Kevin Lyons¹, Swee Leong¹, Frank Riddick¹

¹National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, Maryland, USA, 20899
john.michaloski@nist.gov

²General Motors
30500 Mound Road
Warren, Michigan, USA, Postal
jorge.arinez@gm.com

Keywords: Discrete event simulation (DES), energy management system (EMS), manufacturing execution systems (MES), production, sustainability.

Abstract

The desire to be environmentally sustainable gives manufacturers the necessary impetus to implement “green” technology that previously may have been regarded as less important. Traditionally, Energy Management Systems (EMS), which handle energy-related activities within building services, and Manufacturing Execution Systems (MES), which handle production activities, have been isolated from one another. Clearly, the integration of EMS-MES offers a compelling opportunity to make important energy-efficient contributions toward manufacturing sustainability. Discrete Event Simulation (DES) has been very valuable for manufacturing applications as an efficient analysis tool to aid problem solving and decision-making. This paper analyzes the requirements of EMS-MES system integration within the framework of DES. A case study of the EMS-MES system integration for precision sand casting production will be explored.

1. INTRODUCTION

Sustainable manufacturing is achieved by many strategies, including improving process efficiency, reducing waste, and conserving energy and resources. Positive sustainable manufacturing outcomes translate into an improved “triple bottom line” (Elkington 1997), that is, financial profitability, environmental integrity, and social equity. Manufacturing sustainability has many goals, but a primary objective is to reduce energy consumption in order to reduce greenhouse gases and prevent natural resource depletion. It is estimated that manufacturing in the United States uses one-third of all the energy consumed (DOE 2004). Considering that buildings constitute about

39 % of all energy use, a substantial portion of this manufacturing energy is consumed directly by the building facility apart from any manufacturing activities. To achieve sustainable manufacturing, adequate steps must be taken to accurately measure the total costs and impact of production operations and the associated building infrastructure.

Manufacturing facilities are complex buildings where many of the production and energy relationships are modeled in a suboptimal way. Part of the problem can be attributed to the functionality schism between energy management system (EMS), which handle energy-related activities within building services, and manufacturing execution systems (MES), which handle production activities. In effect, MES and EMS are “silo” operations, that is, isolated subsystems from each other. Production, quality, and other MES data reside in a separate database from that of EMS energy and environmental data. The sharing of data between the EMS-MES subsystems is limited, often manually exchanged, and impedes the ability to make sustainability improvements. Clearly, the computer integration of EMS-MES offers a compelling opportunity to make important energy-efficient contributions toward manufacturing sustainability.

This paper will investigate the sustainability opportunities related to the integration of MES control of industrial automated production and EMS which is part of building services. This EMS-MES integration poses unique system and data modeling challenges whose resolution has the potential to improve both productivity and energy performance for manufacturing firms. Discrete Event Simulation (DES) offers an advantageous framework in which to formally model EMS-MES and understand the integration issues.

The rest of the paper is organized as follows. Section 2 introduces a “silo” production system architecture to highlight the EMS and MES functional requirements. Section 3 describes a Unified Modeling Language (UML) component analysis of EMS and MES and then proposes an initial schedule interface for EMS-MES integrated communication. Section 4 performs a sustainability analysis of EMS-MES to develop a DES model based on key performance indicators. Section 5 discusses a case study of EMS-MES integration using DES to model an automotive precision casting production facility. Section 6 contains a brief summary of the EMS-MES analysis.

2. SYSTEM ARCHITECTURE

In manufacturing, numerous system technologies and the integration of such technologies are necessary to make products. The modern manufacturing plant has a large set of interrelated systems but the scope of this paper will be on Enterprise Resource Planning (ERP), MES, building services, EMS, industrial automation systems found on the plant floor and external factors. Figure 1 shows the manufacturing technology of interest in this paper related to EMS-MES sustainable integration. These systems, which serve to manage some aspect of a manufacturing plant’s operations, are not always well connected nor integrated despite their common sustainability objectives, which include improving energy efficiency.

Within building services, EMS is a general term given to the software subsystem dedicated to energy-related building functions (Maghsoodlou 2004). EMS monitors and controls a building’s energy related systems, including heating, ventilating and air conditioning (HVAC), hot water, interior lighting, exterior lighting, on-site power transmission, and other energy-related building devices. EMS has a subcomponent that manages the local power distribution, a HVAC subcomponent to manage factory environment based on the production requirements, a component to manage the lights in the facility and a component for distribution of compressed air.

The basic EMS objective is to accurately capture overall plant energy consumption in order to validate the manufacturing plant’s utility bills. EMS that is highly integrated with production on the plant floor has the potential to perform advanced functionality. Integrated EMS can adapt production schedules based on real-time pricing to reduce energy consumption for energy-intensive

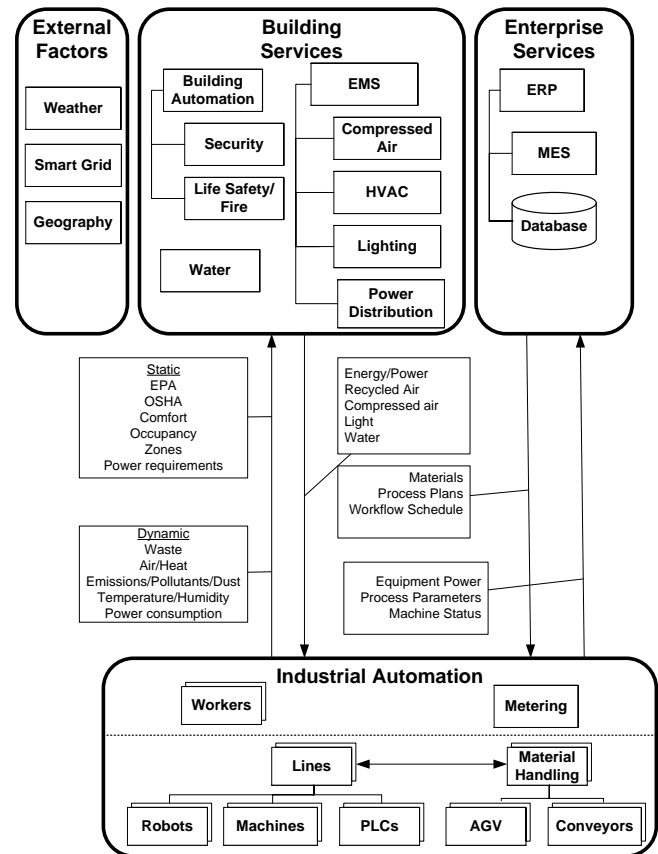


Figure 1 EMS-MES Sustainable Relevant Architecture

yet non-time-critical processes (Steininger 1998). Integrated EMS can perform demand side management by managing energy demand through direct control of equipment operation. (Parker et. al. 2010) Demand side management includes load shedding, peak shaving, and load shifting. If on-site energy generation is available, such as solar panels, peak shaving replaces some of the utility electricity with the on-site electrical power. Load shedding is the removal of unnecessary or low-priority loads, typically to reduce peak demand when the facility is under heavy load. This requires real-time data about the energy needs of the production in order to properly reduce the load. Load shifting involves moving load from on-peak to off-peak electrical tariff rates.

The external factors of climate, geography, and Smart Grid impact production energy consumption. Clearly, more energy resources are required when outside weather conditions are at temperature extremes due to heat conduction losses/gains transmitted through the walls, windows, and ceilings. Consequently, geography is important in understanding the EMS models. EMS simulation would be required to employ some means of representing local climate and geography in order to predict

the EMS behavior. In the near future, EMS must consider the implications of Smart Grid utility power, which is capable of real-time energy pricing. With the Smart Grid, manufacturers could vary energy usage in response to dynamic price fluctuations and manage their energy costs by shifting usage to a lower cost period.

ERP is the broad term for the set of manufacturer activities that include marketing and sales, field service, production planning, inventory control, procurement, logistics and distribution, human resources, finance and accounting. One of the ERP production components is MES, which is used to control processes and infrastructure. MES (Qui 2004) are single or multiple pieces of software that together perform such roles as machine scheduling and monitoring, inventory and product tracking, quality monitoring, material movement management, maintenance dispatching, and operating allocation. MES provides the information and control link between the process planning system (Rehg 2005) at higher ERP level and the industrial automation control systems at the lower level. MES can perform real-time data acquisition and archiving of the operational behavior of the control automation and other auxiliary plant equipment. MES data analysis is useful for understanding equipment utilization, production scheduling, and yield.

In this paper, industrial automation will be defined as the computer programmable control of resources and processes that are used to make products. Industrial automation has many production requirements, such as high utilization, reliability, and scalability. Industrial automation equipment includes computers, machines, robots, Programmable Logic Controllers (PLC), and material handling devices such as conveyors and Automatic Guided Vehicles (AGV). Industrial automation that is completely automated requiring no workers is termed “Lights out” manufacturing, and essentially eliminates the need for many comfort-related Building Services. However, in this paper the presence of workers on the shop floor is assumed and consequently the need to address worker comfort, safety, and health issues. In this instance the manufacturers’ EMS must deal with managing comfort level and indoor air quality (IAQ), which is important not only for workers' comfort but also for their health as mandated by government regulation.

The industrial automation requires a healthy

environment for workers and local power distribution for equipment to operate. HVAC must account for appropriate IAQ that complies with Occupational Safety and Health Administration (OSHA), Environmental Protection Agency (EPA), Center for Disease Control (CDC) safety, and Department of Health and Human Services (DHHS) National Institute for Occupational Safety and Health (NIOSH) health guidelines. For example, OSHA standards prohibit worker exposure to an average of more than 50 parts of carbon dioxide gas per million parts of air during an 8-hour time period. Taken in the context of industrial process that have chemical emissions, compliance to IAQ safety, and health regulations can be quite complex and require considerable effort.

3. ANALYSIS EMS-MES SYSTEM INTEGRATION

ISO 50001 (ISO 2010) is an energy management standard that specifies requirements to achieve continual improvement of energy performance, energy efficiency and energy conservation. ISO 50001 specifies requirements of an EMS for an organization to develop and implement an energy policy, establish objectives, targets, and action plans. To demonstrate conformance to the ISO 50001 standard, a company must have a sustainable EMS in place, complete a baseline of energy use, and commit to continuous assessment and improvement of their energy use.

Currently, prescribed energy reduction methods in industry are often related to lean manufacturing concepts and include energy treasure hunts, value stream mapping, Six Sigma, and Kaizen events (EPA 2007). Most of these methods rely on empirical observation and basic analysis. The continuing lower cost of networks and computers makes energy data acquisition from production systems easier. Increasingly, companies collect process, energy, and facility data from the various control and supervisory systems on the plant floor.

Although data collection is routinely done, it often uses many (often unconnected and incompatible) data collection subsystems, especially when dealing with MES production data and EMS energy data (Arinez 2010a). Given separate systems and databases, this complicates the effort for EMS-MES information sharing and achieving sustainability. To overcome this dilemma, an analysis of EMS-MES system integration will be developed.

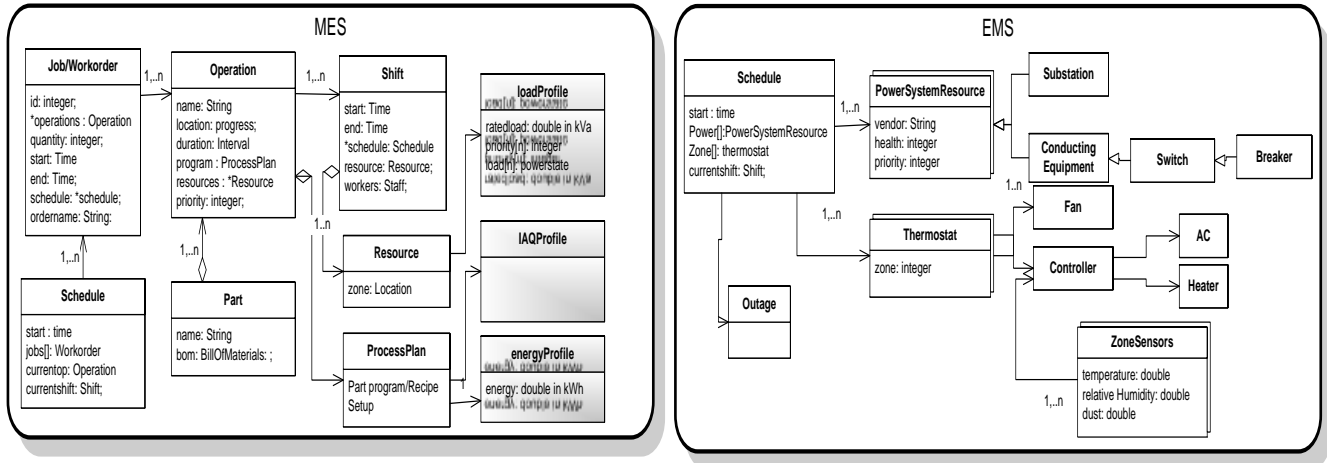


Figure 2 UML Class Diagram of MES and EMS Subsystems

First an analysis of the EMS and MES components will be performed to find areas of functional overlap where information exchange is important. UML class diagrams are used to organize and describe the subcomponent data and functionality (OMG 2005a and OMG 2005b). Figure 2 shows the UML class diagram analysis for MES and EMS. MES uses production planning (not in scope) to generate a schedule that assigns factory Resource(s) including equipment, personnel, process plans, and facilities, to perform a series of Jobs (also known as Workorders) containing Operation(s) to make Part(s) as constrained by production Shifts. Each Part contains a bill of materials (BOM), which is a list of the raw materials, parts, sub-assemblies, and the quantities needed to manufacture a Part. The MES dispatches and monitors the schedule as it is executed.

Figure 2 shows unique MES analysis concepts as shaded entities, including the incorporation of a load profile associated with a Resource and the energy intensity and the IAQ associated with a process. Included in the load profile is the new concept of power states, (e.g., off, sleep, idle, busy), that refine characterization of energy usage and a priority parameter defining the ability of a Resource to shed load.

The EMS subsystem described in Figure 2 is based on the IEC 61970-301 Common Information Model (CIM) standard (IEC 2003) developed by the electric power industry to provide a comprehensive, logical view of electric power transmission and distribution. The EMS subsystem contains a Schedule component that provides the capability

to schedule power transactions. A PowerSystemResource is defined as a generic power system superclass of various derived power system equipment entities, such as lines, breakers, switches, and substations. Note, the lighting and measurement aspect of the EMS component is out of scope.

With considerable potential benefit, the initial focus of the integration will be aimed at scheduling. Loose coupling between the EMS and MES will be considered the important modularity requirement for our analysis (Vinoski 2005). Essentially, any lower level data will be duplicated and then copied into EMS-MES communication, as opposed to allowing the EMS full navigability through the MES database. Thus, the MES creates an equivalent schedule of top level operation sequences with additional data on energy and IAQ related activity: operations, resource and loads, process energy intensity and waste/emission specifics, and down time activity such as maintenance. Figure 3 shows the class interface to integrate EMS-MES scheduling activity.

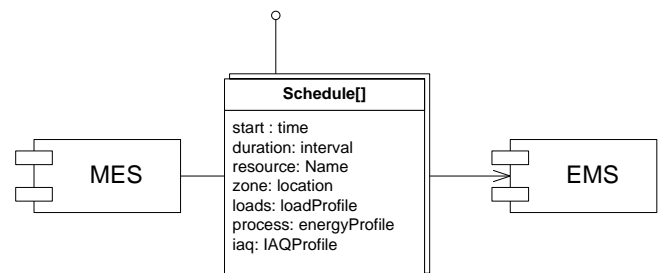


Figure 3 UML Class Diagram of EMS-MES Schedule Interface

The rate of EMS-MES schedule exchange could range from once at the beginning of a shift to more frequent real-time exchange. Real-time schedule exchange would be important where dynamical disturbances, such as equipment or process breakdowns, would mean that less heat is being generated on the factory floor, and would imply the need for a revised lower-intensity EMS schedule.

The scope of data contained within an EMS-MES schedule exchange could vary from basic coverage of actual production, to then adding maintenance operations, to a more complete activity summary including all resource uptime. The inclusion of up and down transitions is important if more dynamical EMS response to production fluctuation is required, so that various HVAC, light, or power can be adjusted in real-time.

4. EMS-MES SIMULATION MODELING

DES is a computer model of a real or proposed system, which is well suited for EMS-MES integration analysis. DES is particularly effective because it is a controlled environment that allows study of Key Performance Indicators (KPI) and the sustainability metrics under different conditions. Further, these prediction models can be developed without disturbing production operations. Compared to ad hoc analysis, DES offers a practical formalism for system modeling that provides holistic understanding and dynamical system analysis.

The prevailing use of simulation in building-only technology has been to refine design, construction and operation parameters yielding substantial sustainability benefits (Ellis 2008, Issa 2009). The use of simulation in EMS analysis is extensive (Contreras 2001), but focuses on energy-specific issues, mostly ignoring MES aspects. For manufacturing, the use of simulation to study sustainability is an emerging analysis technique. Solding and Thollander (Solding 2005) consider design issues related to energy and power utilization inside an iron foundry. Heilala et. al. (Heilala 2008) propose an integrated factory simulation tool for the design phase to help maximize production efficiency and balance environmental constraints and present methods for calculating energy efficiency, carbon dioxide emissions and other environmental impacts. Kuhl and Zhou (Kuhl 2009) show sustainability modeling and simulation of logistics and transportation systems, but does not incorporate real-time data collection. Research into industrial process and sustainability analysis using Discrete

Event Simulation (DES) has been illustrated, but does not offer much insight into the integration of EMS and MES functionality.

DES data integration involves a number of steps: data collection, cleaning and filtering, state and event correlation, and finally data fitting to statistical distributions. From the production process side, the relevant DES data is idle time, part cycle times and disturbance data (i.e., system breakdowns). These variables are primarily characterized by time, but can have energy data elements (e.g., machine loads, process energy intensity) associated to a given state.

The data is then filtered into KPI by fitting the data to statistical distributions. For MES and factory floor production, throughput, utilization, and cycle time are considered KPI for DES manufacturing analysis. DES analysis for energy KPI yields: cost and energy consumption and waste. Table 1 summarizes the EMS-MES functional and KPI performance metric data requirements.

Table 1 Functional and Performance Metrics' Requirements

System	Functional Requirements	Performance Metrics - KPI
ERP	Maximize return on investment (ROI) and quality	job, product and inventory costs
MES	Manage and control production operation	throughput, yield, machine and process efficiencies, waste(scrap cost), maintenance, scheduling
EMS	Supply water, power, compressed air, Maintain healthy IAQ	energy consumption, energy losses, Indoor environment (temperature, humidity, emissions, heat generation, and dust)
HVAC		Return: air temperature, air relative humidity, air volumetric flow rate, Supply: air temperature, air static pressure, air volumetric flow rate.

Industrial Automation	Produce quality parts with high throughput	cycle time, idle time, scrap, part quality
Equipment	High overall equipment effectiveness (OEE) and reliable operation	uptime, faults, energy consumption,
Outdoor Weather	Predict weather	air temperature, relative humidity, sky radiant temperature, solar radiation, wind speed, and wind direction

Given the production and energy (for both MES and EMS) data and statistical characterization, the factory is modeled in DES so that potential benchmarks can be run to project different operational outcomes. The first DES performance metric to measure is an EMS-MES baseline using historical energy consumption as well as production performance data. The EMS-MES baseline is defined as a measurement of the initial period of metered energy and production data used as a point of reference for comparison purposes. With the EMS-MES baseline, performance benchmarking can be done. Performance benchmarking is the process of evaluating current EMS-MES performance against the EMS-MES baseline (DOE 2010). Benchmark comparison of performance metrics can also be normalized and then used for comparing EMS-MES performance of one plant to that of other similar plants. Thus, benchmarking can be used to compare performance over time, within and between equivalent facilities, or to document top performers.

5. CASE STUDY

The DES EMS-MES analysis was applied to a case study of a precision casting production facility. Figure 4 shows a high-level overview of the precision casting process. The molten aluminum process is responsible for melting the aluminum, refining the melt, and adjusting the molten chemistry. Once molten, the aluminum is degassed, leveled, and laundered to remove deleterious gases before being tapped to flow into cores. Cores are made of sand that is poured into molding machines to create the contours of the casting, then pressed and heated to bind the sand. Since the sand casting process is an expendable mold metal casting process, the core process builds a new sand core for each casting. Overall, core parts are molded from sand and binding elements, assembled into the engine block core, and

then dried before casting. The casting and finishing process is where the molten aluminum flows into the sand cast core, after which, the casting is cooled and then casting sand is removed from around the now solidified aluminum engine block by shakeout, trim, and degating operations

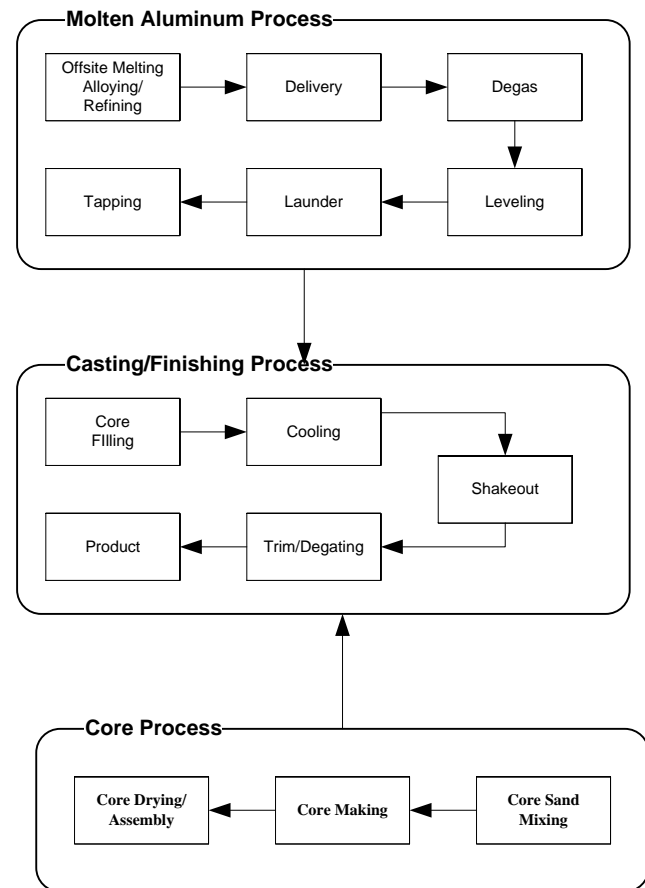


Figure 4 Overview Precision Casting

Some observations are in order. Because of intellectual property issues, representative data will be given, not actual performance data. The precision sand casting operation involves hundreds of electrical equipment being controlled - robots, conveyors, elevators, sand core making machines, saws, etc. The extent of the casting production size necessitated narrowing the initial analysis scope to one of the finishing lines. The analysis was also limited to data already being collected by the plant's production system. The granularity of the actual precision casting data was limited to process steps, which grouped related equipment into a cell. Within a process step, raw cycle time casting and equipment fault data was collected and easily adapted into

process KPI parameters.

The Finishing process was simulated with three steps: Spiral Cooling, Blast Robot, and Degating. These operations have resources that were simulated with the Seize-Delay process model (Kelton2009). Once the simulated delay is completed, the casting is moved forward in the simulation lines. Input and output queues are associated with each operation, but queue states (blocked/starved), conveyor times, buffering, were not addressed in this initial analysis. The casting production facility has a target castings yield per hour that served as the baseline DES performance benchmark.

Previous work on integrating process and energy data discusses the difficulties in integrating and simulating process energy (Arinez 2010b). In this case, the commercial simulation package only allowed one cost function to be associated to a given process state, specifically, burn rate. To overcome this limitation, simulation of process energy was performed in an entirely parallel sequence that summated the energy consumption based on machine state duration (i.e., BUSY, IDLE, or DOWN).

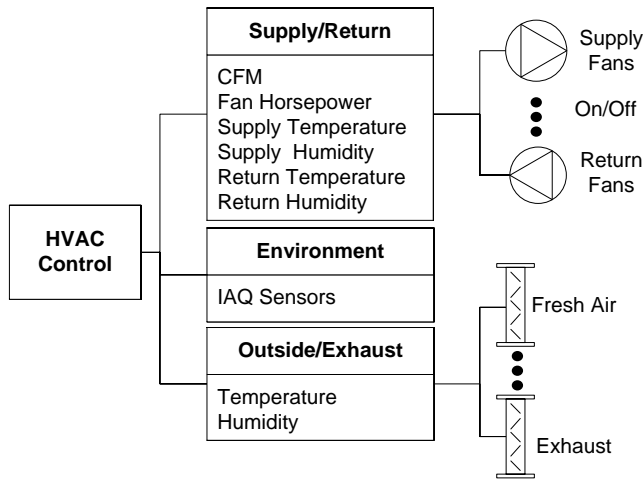


Figure 5 DES Model of HVAC Control

To allow easier addition of more cost functions, the EMS-MES simulation was based on DEVS (Zeigler 2000), an open source C++ DES software package. Figure 5 models a factory HVAC control that turns on or off fans on the plant floor based on a schedule. The schedule provides a description for the supply/return fans and fresh air/exhaust dampers. For the fans, the schedule contains desired airflow as measured in cubic feet per minute (CFM), with a

corresponding fan horsepower (HP). Typically, the EMS control is seasonal, one schedule for winter and a higher air flow schedule for summer. Table 2 shows a portion of the HVAC schedule, which depending on production energy intensity (i.e., on or off) and outdoor weather (e.g., hot, mild, cold) would add or remove active fans and either raise or lower temperature setpoints.

Table 2 Illustrative Portion of HVAC Schedule

HVAC	On/Off	CFM	HP	Fresh Air (CFM)	Temp Setpoint
Finish Supply	On	55,000	60	--	78
Finish Fresh Air Damper	--	--	--	55,000	--
Degate Return	On	47,000	150	--	--
Cooling Return	On	50,000	50	--	--

6. SUMMARY

It is clear that a manufacturing plant is a rather complex building where many of the process and energy relationships are modeled in a non-optimal way. Often the building services satisfy just the nominal worst-case HVAC production requirements because it can be too costly to implement more optimal facility services due to the lack of collection of equipment and floor sensor data and the lack of data integration standards. Some issues of concern related to EMS-MES integration as needed for modeling and computer simulation are identified.

In this paper, the potential for energy reduction was analyzed through the sharing of an MES process schedule with an EMS energy schedule to understand the issues relevant to making this beneficial. In terms of EMS-MES integration there is a need for additional MES data not typically available (i.e., equipment loads and energy losses, and expected process energy use and heat generation).

References

- ARINEZ, J. AND BILLER, S., JAN 2010A. INTEGRATION REQUIREMENTS FOR MANUFACTURING-BASED ENERGY MANAGEMENT SYSTEMS. IN PROCEEDINGS OF IEEE CONFERENCE ON INNOVATIVE SMART GRID TECHNOLOGIES (ISGT).
- ARINEZ, J., BILLER, S., LYONS, K., LEONG, S., SHAO, G., LEE, B. E., AND

- MICHALOSKI, J. 2010B. BENCHMARKING PRODUCTION SYSTEM, PROCESS ENERGY, AND FACILITY ENERGY PERFORMANCE USING A SYSTEMS APPROACH. IN PERFORMANCE METRICS FOR INTELLIGENT SYSTEMS WORKSHOP (PERMIS '10) BALTIMORE, MARYLAND, USA.
- CONTRERAS, J., LOSI, A., RUSSO, M., AND WU, F. F. 2001. SIMULATION AND EVALUATION OF OPTIMIZATION PROBLEM SOLUTIONS IN DISTRIBUTED ENERGY MANAGEMENT SYSTEMS. IEEE POWER ENGINEERING REVIEW, VOL.21, NO.11, PP.57-57.
- DEPARTMENT OF ENERGY (DOE), 2004. ENERGY USE, LOSS AND OPPORTUNITIES ANALYSIS: U.S. MANUFACTURING AND MINING, U.S. DEPARTMENT OF ENERGY, OFFICE OF INDUSTRIAL TECHNOLOGIES, INDUSTRIAL TECHNOLOGY PROGRAM, WASHINGTON, DC.
- DEPARTMENT OF ENERGY (DOE), 2010. FEDERAL ENERGY MANAGEMENT PROGRAM (FEMP) - BUILDING ENERGY USE BENCHMARKING GUIDANCE.
- ELKINGTON, J., 1997. CANNIBALS WITH FORKS: THE TRIPLE BOTTOM LINE OF 21ST CENTURY BUSINESS. NEW SOCIETY PUBLISHERS
- ELLIS, P.G., TORCELLINI, P.A., AND CRAWLEY, D.A. 2008. SIMULATION OF ENERGY MANAGEMENT SYSTEMS IN ENERGYPLUS, PROC. OF THE 10-TH CONFERENCE. INTERNATIONAL BUILDING PERFORMANCE SIMULATION ASSOCIATION (IBPSA).
- ENVIRONMENTAL PROTECTION AGENCY (EPA) 2007. LEAN ENERGY TOOLKIT EPA-100-K-07-003.
- HEILALA, J., VATANEN, S., TONTERI, H., MONTONEN, J., LIND, S., JOHANSSON, B., AND STAHRE, J. 2008. SIMULATION-BASED SUSTAINABLE MANUFACTURING SYSTEM DESIGN. IN WSC '08: PROCEEDINGS OF THE 40TH CONFERENCE ON WINTER SIMULATION. PP. 1922 -1930.
- INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC) 2003. IEC 61970-301 ENERGY MANAGEMENT SYSTEM APPLICATION PROGRAM INTERFACE (EMS-API) - PART 301: COMMON INFORMATION MODEL (CIM) BASE.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO), 2010. DRAFT INTERNATIONAL STANDARD ISO/DIS 50001 - ENERGY MANAGEMENT SYSTEMS — REQUIREMENTS WITH GUIDANCE FOR USE.
- ISSA, R.R.A., SUERMANN, P.C., OLBINA, S. 2009 , "USE OF BUILDING INFORMATION MODELS IN SIMULATIONS," *WINTER SIMULATION CONFERENCE (WSC), PROCEEDINGS OF THE 2009* , VOL., NO., PP.2664-2671, 13-16.
- KELTON, R. S. W. AND SWETS, N. 2009. SIMULATION WITH ARENA. MCGRAW-HILL, INC., NEW YORK, NY, USA.
- KUHL M., AND ZHOU, X. 2009. SUSTAINABILITY TOOLKIT FOR SIMULATION-BASED LOGISTICS DECISIONS. IN PROCEEDINGS OF THE 2009 WINTER SIMULATION CONFERENCE (WSC), PP 1466-1473.
- MAGHSOODLOU, F., MASIELLO, R., AND T. RAY, SEPT.-OCT. 2004 . ENERGY MANAGEMENT SYSTEMS. IEEE POWER AND ENERGY MAGAZINE. VOL. 2, NO. 5, PP. 49–57.
- MAYER, A.L. 2008. STRENGTHS AND WEAKNESS OF COMMON SUSTAINABILITY INDICES FOR MULTIDIMENSIONAL SYSTEMS, ENVIRONMENT INTERNATIONAL, VOL. 34, NO. 2, 277-291.
- OBJECT MODELING GROUP (OMG) 2005A. UNIFIED MODELING LANGUAGE SPECIFICATION (VERSION 2.0): INFRASTRUCTURE.
- OBJECT MODELING GROUP (OMG) 2005B. UNIFIED MODELING LANGUAGE SPECIFICATION (VERSION 2.0): SUPERSTRUCTURE.
- PARKER, J., LINDSAY, H., AND BROWN, B. WHITE PAPER - TACKLING TODAY'S DATA CENTER ENERGY EFFICIENCY CHALLENGES – A SOFTWARE-ORIENTED APPROACH. SCHNEIDER ELECTRIC. [HTTP://WWW.GRAYBAR.COM/DATA-CENTERS/WHITEPAPERS/DATA-CENTER-ENERGY-EFFICIENCY-CHALLENGES.PDF](http://www.graybar.com/data-centers/whitepapers/data-center-energy-efficiency-challenges.pdf) VISITED: 8 Nov. 2010.
- QUI, R. G., AND RUSSELL, D. W. , JANUARY 2004 . A FORMAL MODEL FOR INCORPORATING SHOP FLOOR CONTROLS INTO PLANT INFORMATION SYSTEMS, THE INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY, VOL. 23, NO. 1-2, PP. 47–57.
- REHG, J. A. AND KRAEBBER, H. W. 2005. COMPUTER-INTEGRATED MANUFACTURING. PRENTICE-HALL: ENGLEWOOD CLIFFS, N.J.
- SOLDING, P. AND THOLLANDER, P., 2006. INCREASED ENERGY EFFICIENCY IN A SWEDISH IRON FOUNDRY THROUGH USE OF DISCRETE EVENT SIMULATION. IN PROCEEDINGS OF THE 2006 WINTER SIMULATION CONFERENCE, PAGES, PP. 1971-1976.
- STEININGER, R. 1988. ENERGY MANAGEMENT SYSTEM-AN INTEGRATED PART OF PLANT AUTOMATION. IEEE/PCA 40TH CEMENT INDUSTRY TECHNICAL CONFERENCE.
- VINOSKI, S. 2005. "OLD MEASURES FOR NEW SERVICES," INTERNET COMPUTING, IEEE , VOL.9, NO.6, PP. 72- 74.
- ZEIGLER, B. P., PRAEHOFFER, H., AND KIM, T. G. 2000. THEORY OF MODELING AND SIMULATION: INTEGRATING DISCRETE EVENT AND CONTINUOUS COMPLEX DYNAMIC SYSTEMS, 2ND ED. ACADEMIC PRESS.

Session 2: Data Sensing

23 **Real-Time Occupancy Detection using Decision Trees with Multiple Sensor Types**

EBENEZER HAILEMARIAM, RHYS GOLDSTEIN, RAMTIN ATTAR and AZAM KHAN
Autodesk Research

31 **Sensor Placement Tool for Rapid Development of Video Sensor Layouts**

TYLER GARAAS
Mitsubishi Electronics Research Laboratories (MERL)

35 **A New System Dynamics Framework for Modeling Behavior of Vehicle Sharing Systems**

DIMITRIS PAPANIKOLAOU
Massachusetts Institute of Technology (MIT)

Real-Time Occupancy Detection using Decision Trees with Multiple Sensor Types

Ebenezer Hailemariam, Rhys Goldstein, Ramtin Attar, Azam Khan

Autodesk Research
210 King St. East, Toronto, Ontario, Canada, M5A 1J7
{firstname.lastname}@autodesk.com

Keywords: Occupancy detection, sensor data, classification, decision tree, building performance

Abstract

The ability to accurately determine localized building occupancy in real time enables several compelling applications, including intelligent control of building systems to minimize energy use and real-time building visualization. Having equipped an office workspace with a heterogeneous sensor array, our goal was to use the sensors in tandem to produce a real-time occupancy detector. We used Decision Trees to perform the classification and to explore the relationship between different types of sensors, features derived from sensor data, and occupancy.

We found that the individual feature which best distinguished presence from absence was the root mean square error of a passive infrared motion sensor, calculated over a two-minute period. When used with a simple threshold, this individual feature detected occupancy with 97.9% accuracy. Combining multiple motion sensor features with a decision tree, the accuracy improved to 98.4%. Counterintuitively, the addition of other types of sensors, such as sound, CO₂, and power use, worsened the classification results. The implication is that, while Decision Trees may improve occupancy detection systems based on motion sensors alone, one risks overfitting if multiple types of sensors are combined.

1 INTRODUCTION

In addressing building performance, we face a complicated balance between occupant comfort and energy consumption, as comfort and energy savings are often inversely related [Yilmaz, 2007]. If we are to achieve this balance, intelligent building systems are needed that are aware of not only their occupants' environment, but also the number of occupants within that environment. In most office buildings, the office cubicle is what demarcates an occupant's personal workspace. It is within these semi-open spaces that we attempt to detect occupancy.

The ability to accurately determine localized building occupancy in real time enables several compelling applications; examples include the intelligent control of building systems, fine-grained space utilization data collection, real-time building visualization [Hailemariam et al., 2010], and security systems. An occupant detector dedicated to a specific cubicle

could be used to inform a Personal Environmental Module, a device that controls the environment around a single office worker [Mendler et al., 2006]. With an occupant detector at nearly every workspace, a central building control system could minimize energy wasted in the heating, air conditioning, and lighting of unoccupied spaces.

Currently, most commercial systems which perform occupant detection solely utilize passive infrared (PIR) motion detectors. These systems apply relatively simple analysis to the sensor signal to infer occupancy from the degree of apparent motion. One problem with these systems is that PIR sensors fail to detect subjects which remain relatively still. Furthermore, distant passersby and wafts of warm or cold air are interpreted as motion leading to false positives. Traditional approaches also rely on camera-based methods [Ramoser et al., 2003]. However, these methods require complex video processing and introduce privacy concerns.

In our approach we embed a number of low-cost sensors of different types into the cubicle furniture. Using these sensors we measure several attributes of the occupant's environment: light, sound, CO₂ level, power use, and motion. Through Decision Tree analysis we develop a method to deduce the occupancy of the workspace at any given time.

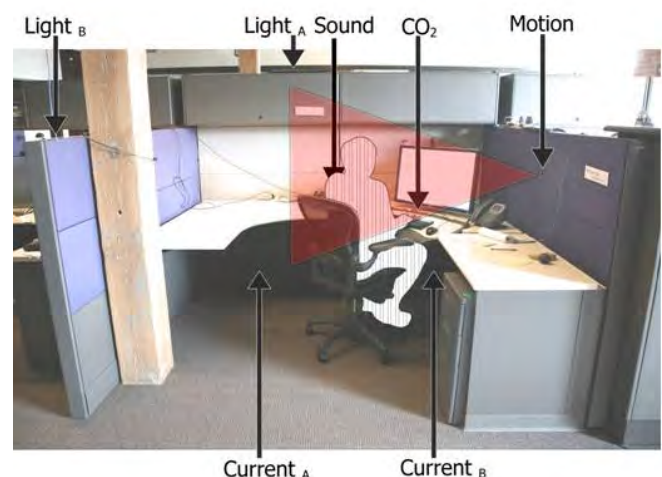


Figure 1. Physical configuration of sensors installed in a cubicle to help detect occupancy. The red cone denotes the sensing region of the motion sensor.

Our approach is not without precedent. The authors of [Tarzia et al., 2009] use sonar along with a simple threshold-based classifier to develop a computer which can detect the physical presence of a user. Similarly, the authors in [Dodier et al., 2006] deployed multiple passive infrared sensors into enclosed offices and used Bayesian probability theory to develop an occupancy classifier.

In [Lam et al., 2009a] and [Lam et al., 2009b], the authors deployed a wider array of sensor types (various gas sensors, sound pressure level, illuminance, PIR, temperature and humidity) into an open office environment to count the number of occupants. The classification methods used by Lam et al. were *Support Vector Machines (SVM)*, *Neural Networks*, and *Hidden Markov Models (HMM)*. Features were selected prior to classification based on *information gain*, a statistical quantity commonly used to measure the degree to which uncertainty is reduced by exploiting additional data.

We also treat occupancy detection as a classification problem, but use *Decision Trees* instead of the methods above. The simplicity of the Decision Tree method allows us to explore relationships in the sensor data. Our approach builds on previous work, focusing on the following:

- Occupant detection, as opposed to occupant counting.
- Occupant detection within a semi-open space (an office cubicle).
- The use of multiple low-cost sensor types to improve the detection quality over a single sensor alone.
- The use of Decision Trees to automatically select the features with the highest information gain and exploit those features to construct an occupancy classifier.
- The use of Decision Trees to explore the relationship between different types of sensors, features derived from sensor data, and occupancy.
- The use of various performance metrics to rank sensor types and features.

It is important to note that different applications of an occupant detector require different levels of accuracy. Whereas an HVAC system may be able to maintain appropriate temperature levels given relatively coarse real-time occupancy information, a security system or a power usage visualization tool may need to detect even short periods of presence and absence. Our interest lies in comparing the relative accuracies of various occupancy detection options, and so we do not seek any particular level of accuracy.

2 APPARATUS

We equipped a cubicle in our office with sensors to measure various attributes of the occupant's environment. Specifically, the sensors we chose measure those attributes which

we believed could be directly correlated with occupancy and at the same time provide information about the environment which may be useful for other applications. The selected sensors measured carbon dioxide (CO₂), electrical current (current), illuminance (light), PIR (motion), and sound pressure level (sound). Table 1 describes the make, model and quantity of each sensor we deployed in our test bed. The physical placement of each of these sensors is illustrated in Figure 1.

Type	Make	Model	Qty.
CO ₂	Sensair AB	k30	1
Current	Elkor Technologies Inc.	iSnail-VC-25	2
Light	Phidgets Inc.	1127	2
Motion	Phidgets Inc.	1111	1
Sound	Phidgets Inc.	1133	1
Total			7

Table 1. Sensors used in cubicle during the study.

Current sensor A (Current_A) measures the current drawn by a computer monitor with a 15 minute power saving timer enabled. Current Sensor B (Current_B) measures the current drawn by a desktop computer. Light sensor A (Light_A) and light sensor B (Light_B) are positioned to capture the conditions of the shared overhead lighting system. The motion sensor is placed such that its sensing region cuts the enclosed space of the cubicle thereby eliminating unwanted interference from passersby. The CO₂ sensor is located atop the occupant's desk between their keyboard and monitor so as to optimally capture exhaled CO₂. Lastly, the sound sensor is positioned so as to capture the voice of the occupant and their visitors from multiple directions while minimizing occlusion.

3 DATA COLLECTION

For our experiment we collected data for a single cubicle in our office over a contiguous seven day period, including weekends and business off hours. Ground truth occupancy data was obtained by recording the occupant using a conventional camera. A human operator would periodically review the video and produce a schedule, accurate to the minute, of when the occupant was present in their cubicle.

Sensor data was represented by sequences of time-value pairs. For two consecutive pairs $\langle t_n, v_n \rangle, \langle t_{n+1}, v_{n+1} \rangle$, value v_n spans the time interval $[t_n, t_{n+1})$. A new time-value pair would only be output if the value changed by a fixed sensor-dependent amount. Thus the sampling rate varied on a per sample basis depending on the variability of the sensor data. Noisy data, such as that produced by motion or sound sensors, would typically be acquired at the maximum sample rate of 2 Hz. The CO₂ sensor, light sensors, and computer current meter were captured at moderate sample rates. The relatively stable computer monitor current meter output rel-

atively few time-value pairs. Over the testing period we collected roughly 41 million data points, as indicated in Table 2.

Type	Number of Data Points
CO ₂	2, 233, 542
Current _A	132, 358
Current _B	3, 980, 179
Light _A	1, 137, 060
Light _B	1, 361, 186
Motion	14, 066, 006
Sound	18, 889, 586
Total	41, 799, 917

Table 2. Number of raw sensor data samples collected over seven consecutive days.

4 OCCUPANCY DETECTION FEATURES

Occupancy detection can be formulated as a classification problem in which a set of features must be automatically associated with one of several classes. In this context, a *feature* is a function of sensor data over a period of time, and a *class* is a representation of occupancy at a single point in time.

The simplest classifiers for occupancy detection use only a single feature and two classes. In [Padmanabh et al., 2009], for example, the state of a conference room is classified based on the number of times a microphone value exceeds a threshold in a 5-minute time interval (the feature). If the threshold is exceeded more than twice, a meeting is assumed to be in progress (one class); otherwise, the conference room is assumed to be in the “no meeting” state (the other class).

Our approach is similar to [Lam et al., 2009a] in that it exploits numerous features derived from multiple sensor types. However, because the classifiers in [Lam et al., 2009a] aim to count the number of occupants in a space, separate classes are used for vacancy, 1 person, 2 people, 3 people, etc. We use only two classes: one which represents the absence of any occupants and one which represents the presence of at least one occupant.

Expressions of the following form denote the value of a feature at time t :

$$\langle \text{type of feature} \rangle_{2j}(\langle \text{type of sensor} \rangle)[t]$$

The integer j indicates that the feature value is calculated over a time period starting at time $(t - 2^j \cdot \Delta t)$ and ending at time t , where Δt is an arbitrary duration. With this notation and a Δt value of 1 minute, for example, $AVG_{25}(CO_2)[11:58]$ represents the average value of a carbon dioxide sensor from 11:26 to 11:58. Note that the time period over which a feature value is calculated precedes the time label t . Our focus on real-time applications motivates the use of only past information to detect occupancy.

Certain types of features provide more information when applied to certain types of sensors. That said, our approach is to apply every type of feature to every type of sensor. The task of identifying the most informative sensor-type/feature-type combinations is left to the classification method, as described later.

The equations that follow define our feature types, with X representing any sensor and $X(t)$ denoting the sensor reading at time t . The first feature we calculate is the average value over a duration of Δt .

$$AVG_{2^0}(X)[t] = \frac{1}{\Delta t} \cdot \int_{t-\Delta t}^t X(t) dt$$

As a sensor’s raw data can often be interpreted as a step function, the integral above can generally be evaluated as a sum of duration-weighted sensor readings. Similarly, the integral in the $(j = 0)$ root mean square (RMS) error feature below can in most cases be treated as a weighted sum of squares.

$$RMS_{2^0}(X)[t] = \sqrt{\frac{1}{\Delta t} \cdot \int_{t-\Delta t}^t (X(t))^2 dt - (AVG_{2^0}(X)[t])^2}$$

The rationale for using time interval widths based on powers of 2 is that the $(j + 1)^{\text{th}}$ set of features can be obtained efficiently from the j^{th} set. We define an upper limit j_{\max} to constrain the number of feature values. Below are the *AVG* and *RMS* feature calculations for time intervals of duration $2 \cdot \Delta t, 4 \cdot \Delta t, 8 \cdot \Delta t, \dots, 2^{j_{\max}} \cdot \Delta t$.

$$AVG_{2^{j+1}}(X)[t] = \frac{AVG_{2^j}(X)[t - 2^j \cdot \Delta t] + AVG_{2^j}(X)[t]}{2}$$

$$RMS_{2^{j+1}}(X)[t] = \sqrt{\frac{MS + MSX}{2} - (AVG_{2^{j+1}}(X)[t])^2}$$

$$\text{where } MS = (RMS_{2^j}(X)[t - 2^j \cdot \Delta t])^2 + (RMS_{2^j}(X)[t])^2$$

$$\text{and } MSX = (AVG_{2^j}(X)[t - 2^j \cdot \Delta t])^2 + (AVG_{2^j}(X)[t])^2$$

For $j \geq 1$, we use four additional types of features per sensor. The two below capture the change in the average sensor value and root mean square error over time.

$$\Delta AVG_{2^{j+1}}(X)[t] = AVG_{2^j}(X)[t] - AVG_{2^j}(X)[t - 2^j \cdot \Delta t]$$

$$\Delta RMS_{2^{j+1}}(X)[t] = RMS_{2^j}(X)[t] - RMS_{2^j}(X)[t - 2^j \cdot \Delta t]$$

The final two types of features are $|\Delta AVG|$ and $|\Delta RMS|$, the absolute values of the feature types defined above.

The set of features defined above can be used with a wide range of different classification methods. A good overview of various methods can be found in [Kotsiantis, 2007]. We chose Decision Trees, which have received little attention for cubicle-based occupancy detection.

5 DECISION TREE METHOD

True to its name, the *Decision Tree* classification method selects a class by descending a tree of possible decisions. At each internal node in the tree, a single feature value is compared with a single threshold value. One of the two child nodes is then selected based on the result of the comparison. When a leaf node is reached, the single class associated with that node is the final prediction. Correlations between features can be exploited, as a particular node's feature will only be used if the features of upper nodes produce a certain path through the tree. For an example of a decision tree, see Figure 4.

A decision tree is generated automatically from *training data*, sets of features coupled with known classes. As explained in [Quinlan, 1996], the feature selected for the root node of the tree is the one with the highest information gain, and the threshold is selected based on a quantity known as the *gain ratio*. For a lower node, the feature and threshold are selected in the same fashion using the subset of the training data that would reach that node. Nodes are only created if they are reached by at least K points in the training data, for some arbitrary K . The tree is also pruned according to the Minimum Description Length principle, as described in [Mehta et al., 1995].

The Decision Tree method is compelling in part because the trees themselves are intuitive and informative. Each path through a tree consists of a combination of features which tend to work together to distinguish between classes. This simplicity gives useful insights into the inner workings of the method. In contrast, the outputs from alternative methods such as SVM, Neural Networks and HMM, as used in [Lam et al., 2009a], are very difficult to interpret and respond to in an exploratory manner. Also, these methods leave feature selection to the user, whereas the Decision Tree method has this process built-in. Lastly, unlike SVM, Decision Tree analysis are not sensitive to the scale of the input data, so no conditioning of the data is necessary.

One weakness of Decision Trees is the danger of overfitting, where statistically insignificant patterns end up negatively influencing classification results. The pruning mentioned above reduces the risk of overfitting, but results in smaller decision trees that exploit fewer features.

6 EXPERIMENTAL SETUP

We used the open source application KNIME (Konstanz Information Miner) 2.1.2.0024559 [Berthold et al., 2007] as the environment to conduct our data analysis. KNIME is an application which provides a number of data mining and statistical analysis algorithms. Specifically, we used its Decision Tree implementation to conduct our study.

The data supplied to KNIME was prepared outside of the application using Python scripts which would calculate the

feature values and align them based on time. We chose $\Delta t = 1$ minute; thus, for every sensor and every minute, dozens of raw samples were reduced into an average value and a noise level (the AVG_1 and RMS_1 features in Section 4). From these 1 minute values, additional features were calculated for time intervals of widths 2, 4, 8, 16, 32, and 64 minutes. The 64-minute maximum time width corresponds to $j_{max} = 6$ ($64 = 2^{j_{max}} \cdot \Delta t$). No single feature would use data from more than 64 minutes in the past.

In the end we had 7 individual sensors, 2 types of features calculated for a time width of 1 minute, and 6 types of features calculated for each of 6 longer time widths. This yielded a vector of 266 feature values for each minute of acquired data. The vectors were used in conjunction with the occupancy data to train and test Decision Trees classifiers. We limited the size of the decision trees by selecting $K = 10$.

7 RESULTS

The experiment consisted of several trial conditions, each involving different combinations of features. All conditions were evaluated using seven-fold cross validation. This means that for a given day in our data set, we used the data from that day as the validation set and used the data from the six remaining days to train the classifier. A result for each of the seven days was produced by calculating the percentage of correctly classified validation points. A single measure of accuracy was produced for the experiment by averaging all seven results. Occupancy detection accuracies are reported in Table 3.

CO ₂	Current	Light	Motion	Sound	Accuracy
		×			81.019%
				×	90.787%
×	×	×		×	94.242%
×					94.679%
	×				96.267%
×	×	×	×	×	96.267%
			○		97.943%
×			×	×	98.213%
			×		98.441%

Table 3. Occupancy detection accuracies achieved using various combinations of sensor types. Each × indicates that all features of the associated sensor type were made available for the associated trial condition. The ○ indicates that only a single motion feature was used.

It is important to recognize that these classification accuracies are sensitive to the time interval Δt , the behavior of the observed occupant, and other experimental conditions. In a different setting, for example, a CO₂-based classifier may yield an accuracy significantly higher or lower than the 94.7% we obtained. However, based on these results, it is reasonable to suspect a CO₂-based classifier to outperform a light-based

detector in an office setting, and a motion-based detector to outperform both. Our analysis focuses not on the overall magnitude of the classification accuracies, but rather on the relative differences in accuracy obtained for each set of conditions.

7.1 Results by Sensor Type

As a baseline measurement, we ran trials using all features of only one sensor type at a time. Light features performed the worst at determining occupancy, followed by sound, CO₂ and current. The best performing group of features of a single sensor type were those derived from motion.

We then ran trials in which the classifier was trained with combinations of features derived from multiple sensor types. By combining features of all sensor types except the best performing sensor, motion, we achieved a classification accuracy greater than that of light and sound features alone. Adding motion features to the mix yielded an accuracy greater than that of CO₂ and current features alone. Under the assumption that overhead lighting conditions and electricity consumption were leading to dubious classifications, we removed features of those sensor types. Removing light and current improved accuracy by 2%.

Surprisingly, no classifier trained with features of multiple sensor types was able to outperform the classifier trained with motion features alone. In fact, a decision tree trained with all available data from all sensors performed worse than a single-node decision tree exploiting only one motion feature. The use of a single-node decision tree is equivalent to applying a threshold to a single feature. A multi-node decision tree still produced the best results, but only when using motion features at all nodes.

Each of the experiments we described so far exhibited different characteristics on the weekend when the occupant was not present for 48 straight hours. While the classifier that was trained using only motion features gave us the best accuracy overall, it was only able to achieve 98.3% accuracy over the weekend. However, classifiers trained with either current features or sound features were able to achieve 100% classification accuracy during this period of extended absence.

7.2 Results by Feature Type

Here we attempt to quantify which features in particular proved to be the most useful in determining occupancy. The metric below computes a non-dimensional score value for each feature. High scores indicate high importance, and vice versa. The scoring formula favors features which are frequently used by decision nodes and are used at decision nodes which are close to the root of the tree.

$$Score(feature) = Occurrences(feature) \cdot \frac{1}{2^{AvgTreeDepth(feature)}}$$

For example, suppose that the feature $\Delta RMS_{64}(Sound)$ occurs in three decision trees at levels 1, 2, and 4. The score for this feature is then calculated as follows.

$$Score(\Delta RMS_{64}(Sound)) = 3 \cdot \frac{1}{2^{(1+2+4)/3}} = 0.595$$

We applied the formula to the set of seven decision trees produced for the trial condition in which all features of all sensor types were used. A summary of the top 16 scoring features is depicted in Figure 2. Overall, the feature most highly favored by the method was $RMS_2(Motion)$, as this feature was chosen as the root decision in most trees. The relative strength of this feature is further emphasized by the fact that in all trees, decisions which were children of decisions based on $RMS_2(Motion)$ very rarely chose a different classification than was already chosen by $RMS_2(Motion)$. As indicated in Table 3, this feature alone produced a classification accuracy of 97.9%. The addition of other motion features only increased the accuracy to 98.4%.

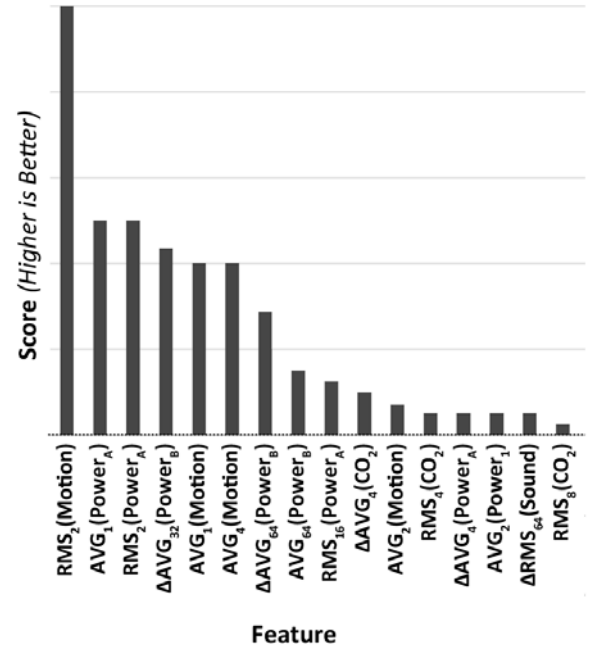


Figure 2. Simple feature scoring based on weighted feature occurrences in decision trees.

8 DISCUSSION

Simultaneously using features derived from multiple sensor types was no better than using features derived from a motion sensor alone. Our initial intuition was that the presence of additional sensor types would allow us to capture occupancy trends which a motion sensor alone is unable to detect. However, it seems that the presence of additional sensor types did

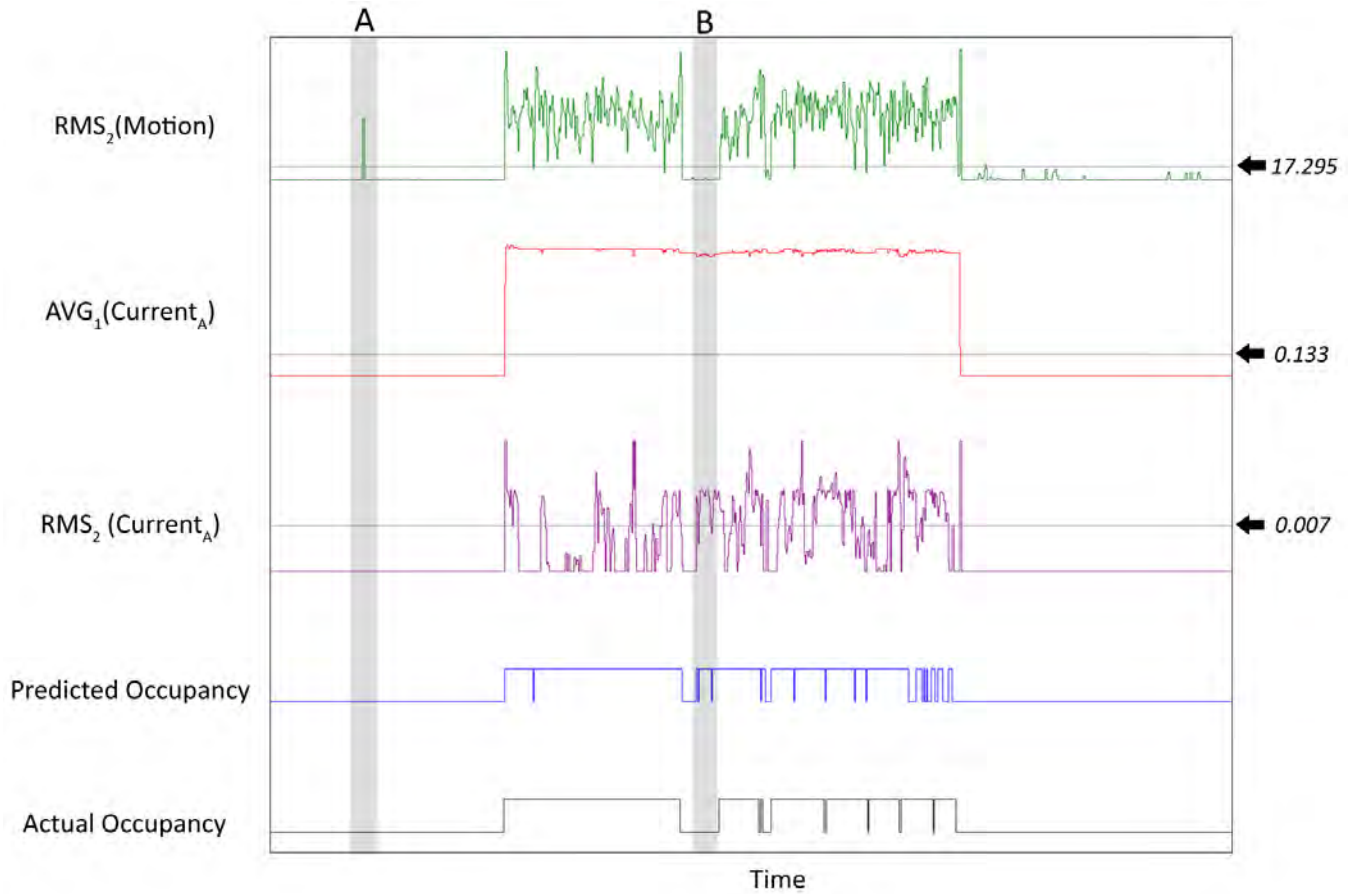


Figure 3. Plot of three features and both predicted and actual occupancy for a single day. For occupancy signals, the upper level indicates presence and the lower level indicates absence. The time periods A and B are discussed in Section 8.

as much harm as good. Here we analyze one of the classifiers to illustrate how additional sensor types sometimes improve occupancy detection, yet sometimes undermine it.

Figure 3 provides plots of the root feature of the classifier, $RMS_2(Motion)$, the two second-level features, $AVG_1(Current_A)$ and $RMS_2(Current_A)$, the predicted occupancy, and the actual occupancy. The predicted occupancy was produced by the decision tree trained using all features of all sensor types. The tree itself is shown in Figure 4.

Over time period A, $RMS_2(Motion)$ produces a spike that crosses the threshold 17.295 when the occupant is not present. In the absence of additional features, this would result in a classification error. However, since $AVG_1(Current_A)$ is below its own threshold of 0.133 over the same time period, it correctly classifies the occupant as being absent. The additional sensor type, in this case a current meter on the computer monitor, improves the classification based on motion data alone.

Over time period B in Figure 3, we can see that $RMS_2(Motion)$ is well below the threshold 17.295, correctly classifying the occupant as absent. However, when we de-

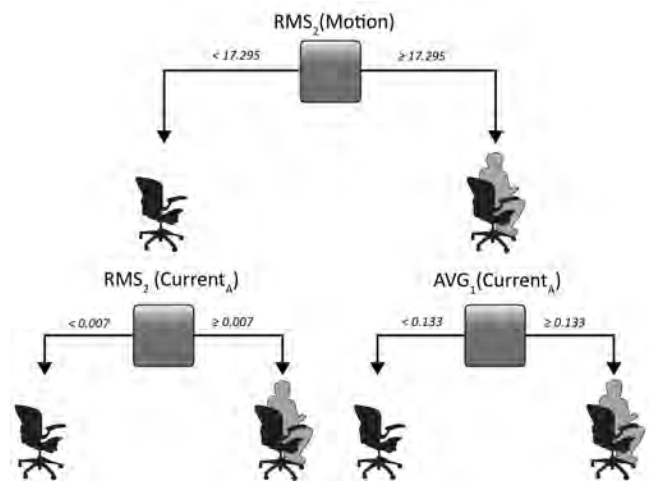


Figure 4. Top two levels of a decision tree trained using all features of all sensor types.

scend one level in the tree, $RMS_2(Current_A)$ undoes the correct classification produced by motion. Here the presence of an additional sensor type leads to a misclassification that would not have otherwise occurred. This is an example of how overfitting may occur in a decision tree, where a low-level node exploits a misleading pattern in the training data.

In our experiment, cases like that of time period B outweighed cases like that of time period A. In some instances the additional sensors of various types corrected decisions based on the motion sensor alone. But surprisingly, in a greater number of instances the motion-based classification ought not to have been changed.

Classification errors can be loosely classified into two categories: *transition errors* and *spurious errors*. Transition errors occur over time periods when the occupant has just arrived or has just left their workspace. When these types of errors occur the classifier is typically late to predict the transition change by an average of one to two minutes. This type of error is largely due to the fact that we only use past data to inform our classifications, generally making the predictor unable to output a state change until past information of the transition is available. Transition errors may also be introduced by slight inaccuracies in the ground truth occupancy data. For instance, the video observer may have marked the occupant absent before they were entirely out of sensor range.

We describe all other classification errors as spurious. Spurious errors can be credited to a multitude of factors including limited training data, inadequate selection of features, overfitting, or anomalies in sensor values or occupancy data.

We found that the majority of classification errors were transition errors. A close examination of Figure 3 reveals several transition errors where the vertical edges of the two occupancy plots do not perfectly align. There are also a few noticeable spurious errors in the plots, the first appearing as a brief predicted absence about halfway between time periods A and B. In building automation applications, relatively short misclassification errors of either type may be tolerable. However, spurious errors are less tolerable when collecting data to calibrate simulations and perform real-time monitoring.

One should be aware that the make and model of sensors used for training must be the same as what is used for classification. To illustrate why, consider a motion sensor that outputs a non-standard non-dimensional value. In this case, one make of motion sensor is not necessarily able to use the decision thresholds arrived at via training with a different motion sensor. Even sensors which output dimensional values such as current (Amperes) and sound (Decibels) are subject to the same limitations, as sensing ranges, resolutions, and sampling frequencies vary by sensor make and model.

9 CONCLUSION

Having improved occupant detection accuracy from 97.9% to 98.4% in our study, Decision Trees may well outperform simple thresholds when applied to multiple features derived from a single motion sensor. Interestingly, the inclusion of other types of sensor data did not improve overall accuracy when combined using a decision tree. The implication is that, while Decision Trees may improve occupancy detection systems based on motion sensors alone, one risks overfitting if multiple types of sensors are combined.

It remains intuitive that the use of other types of sensors in conjunction with motions sensors could improve an occupant detection system. However, in light of our results, it might be best to explore alternative classification methods known to be less prone to overfitting.

Despite mixed results in terms of classification accuracy, we demonstrated how Decision Trees can be used to explore the relationship between sensor types, features, and occupancy. The method is compelling in large part because one can understand how each sample was classified, and often determine the cause of a classification error.

If deployed, our occupancy detection method might have to be adjusted to suit the particular application. The objective of our study was to detect present occupancy at a one-minute resolution, which could be useful for visualization and security applications. For a building control system, one might modify the method to predict occupancy between the present time and an hour or two in the future. In some applications, the real-time aspect of our classifier may not be priority. For example, if one is collecting occupancy data to calibrate a simulation of occupant behavior, then it should be reasonable to use future sensor data as well as past information to produce the feature values.

Future work will need to focus on capturing data from not only different individuals, but individuals who exhibit considerably different behavior in a wide variety of settings. For example, an occupant who is primarily a computer user would produce different patterns in the sensor data and occupancy profiles than one who spends more time on the phone. Data from a diverse set of occupants may yield a generic classifier that is able to detect the presence and absence of many different individuals.

10 ACKNOWLEDGMENTS

Thanks to our colleagues Michael Glueck, Conan Huang, Michael Lee, Wei Li and Alex Tessier for their invaluable support. Also thanks to John King, Regional Facilities Manager and Zulfikar Texiwala, Building Operator, at Autodesk Office in Toronto for their continuous support throughout our ongoing research.

REFERENCES

- BERTHOLD, M. R., CEBRON, N., DILL, F., GABRIEL, T. R., KÖTTER, T., MEINL, T., OHL, P., SIEB, C., THIEL, K. AND WISWEDEL, B., 2007. KNIME: The Konstanz Information Miner. In *Studies in Classification, Data Analysis, and Knowledge Organization (GfKL 2007)*. Springer.
- DODIER, R. H., HENZE, G. P., TILLER, D. K. AND GUO, X., 2006. Building occupancy detection through sensor belief networks. *Energy and Buildings*, volume 38, 9, 1033–1043.
- HAILEMARIAM, E., GLUECK, M., ATTAR, R., TESSIER, A., MCCRAE, J. AND KHAN, A., 2010. Toward a Unified Representation System of Performance-Related Data. In the 6th IBPSA Canada Conference, 117–124.
- KOTSIANTIS, S. B., 2007. Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, volume 31, 249–268.
- LAM, K. P., HÖYNCK, M., DONG, B., ANDREWS, B., CHIOU, Y.-S., ZHANG, R., BENITEZ, D. AND CHOI, J., 2009a. Occupancy Detection Through an Extensive Environmental Sensor Network in an Open-Plan Office Building. In *Proceedings of the 11th International IBPSA Conference*, 1452–1459. Glasgow, Scotland.
- LAM, K. P., HÖYNCK, M., ZHANG, R., ANDREWS, B., CHIOU, Y.-S., DONG, B. AND BENITEZ, D., 2009b. Information-Theoretic Environmental Features Selection for Occupancy Detection in Open Offices. In *Proceedings of the 11th International IBPSA Conference*, 1452–1459. Glasgow, Scotland.
- MEHTA, M., RISSANEN, J. AND AGRAWAL, R., 1995. MDL-based Decision Tree Pruning. In *Proceedings of the First International Conference on Knowledge Discovery*, 216–221. Montreal, Canada.
- MENDLER, S., ODELL, W. AND LAZARUS, M. A., 2006. *The Guidebook to Sustainable Design*. John Wiley & Sons Press, second edition.
- PADMANABH, K., V, A. M., SEN, S., KATRU, S. P., KUMAR, A., C, S. P., VUPPALA, S. K. AND PAUL, S., 2009. *iSense: A Wireless Sensor Network Based Conference Room Management System*. In *BuildSys '09: Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, 37–42. Berkeley, CA, USA.
- QUINLAN, J. R., 1996. Improved Use of Continuous Attributes in C4.5. *Journal of Artificial Intelligence*, volume 4, 77–90.
- RAMOSER, H., SCHLÖGL, T., BELEZNAI, C., WINTER, M. AND BISCHOF, H., 2003. H.bischof. shape-based detection of humans for video surveillance. In *In Proc. of IEEE Int. Conf. on Image Processing*, 1013–1016.
- TARZIA, S. P., DICK, R. P., DINDA, P. A. AND MEMIK, G., 2009. Sonar-based measurement of user presence and attention. In *UbiComp '09: Proceedings of the 11th international conference on Ubiquitous computing*, 89–92. ACM, New York, NY, USA.
- YILMAZ, Z., 2007. Evaluation of energy efficient design strategies for different climatic zones: Comparison of thermal performance of buildings in temperate-humid and hot-dry climate. *Energy and Buildings*, volume 39, 3, 306–316.

Sensor Placement Tool for Rapid Development of Video Sensor Layouts

Tyler W. Garaas

Mitsubishi Electric Research Laboratories
201 Broadway, Cambridge, MA, USA, 02139
garaas@merl.com

Keywords: sensor placement, coverage problem, security monitoring, visibility analysis, ray casting

Abstract

The arrangement of video sensors – in closed-circuit television (CCTV) systems, for instance – can have drastic effects on the efficiency and cost of the final system. In the present work, I describe a tool designed for rapid construction of simulated video sensor layouts that allows quantification of sensor coverage and cost estimation to be determined prior to installation; thus, avoiding costly changes during or after the installation. Most previous work in this area either considers sensor coverage only in a 2D space or requires significant preparation to achieve accurate results in 3D. In the present work, I describe an implementation of a novel coverage-analysis algorithm that uses the geometry of image formation to cast rays from simulated video sensors through the monitored area to estimate sensor coverage at every 3D location. Visualization techniques of the acquired sensor coverage data are additionally presented.

1. INTRODUCTION

The use of closed-circuit television (CCTV) systems, on which I focus in the current work, has seen a rapid increase in the past decade. For instance, an estimated 4.2 million CCTV cameras are deployed in the UK alone as of 2004 (McCahill et al. 2004). Aside from the privacy and effectiveness conversations that are now in the spotlight due to the massive deployment of CCTV systems in public spaces, novel research regarding sensor placement, networking, and event detection are rapidly becoming more commonplace.

When designing a CCTV system, or indeed any video sensor layout, many factors need to be considered. As the monitoring region expands to cover greater and greater area

– sometimes reaching neighborhood- or city-wide coverage – these factors can become critical in controlling cost and ensuring the effectiveness of the CCTV system. For instance, issues such as spatial resolution and angular coverage of certain 3D locations can mean the difference between successful deployment of a face-recognition algorithm, and egregious overlap in sensor coverage can increase system cost dramatically.

1.1. Previous Work

Research related to coverage analysis has roots in a computational geometry problem commonly referred to as *the art gallery problem*. In the art gallery problem, the goal is to place a minimum number of guards such that all wall-positions are observable by at least one guard; see O'Rourke (1987) for a review.

Most of the research regarding the art gallery problem and its variants has ignored certain, significant real-world constraints inherent in the placement of video sensors. As such, some researchers have extended the art gallery problem to include such considerations as limited field of view, visibility, and spatial resolution. Erdem and Scarloff (2004) present an algorithm for the placement of static and active video sensors that incorporates such considerations.

While the large majority of work related to video sensor coverage and placement has focused on 2D spaces (e.g., Erdem and Scarloff, 2004; Yabuta and Kitazawa, 2008), some have recently considered the problem in 3D spaces. Murray et al. (2007) describe a process for placing sensors while considering occlusion in 3D, but base their work on the assumed existence of a visibility algorithm. Becker et al. (2009) describe a method for placing cameras in a 3D environment based on the visibility of individual 3D locations. However, this work does not consider the quantification of visibility, but treats coverage of the

locations binarily as either visible or not visible. Finally, van den Hengel (2009) present a method for placement of video sensors in a 3D environment based on the visibility of ‘marker locations’ that meet a minimum criterion – e.g., number of pixels that observe the marker. This work also does not describe quantitative coverage analysis – aside from the definition of a minimum criterion – and, furthermore, the 3D model must be manually annotated with meta-data, which appears to be a slow process.

Perhaps the previous work most closely related to the present work, is a commercial product known as VideoCAD (<http://www.cctvcad.com>), which includes many functionalities to aid in the creation of CCTV systems. However, while VideoCAD does provide an interface for designing CCTV layouts, it does not allow quantitative analysis of sensor coverage.

1.2. Overview

In the present work, I describe *Sensor Placement Tool* (SPtool, a non-commercial, internal tool used by Mitsubishi Electric), which facilitates rapid construction of 3D environments and effective video sensor layouts. To start, users construct a 3D representation of the environment they wish to monitor by adding and manipulating 3D models using GUI controls common in 3D modeling and CAD software. Since 3D modeling software has become commonplace through products such as Google Sketchup, Blender, and 3DS Max among many others, details regarding the construction of 3D environments are omitted here. Figure 1 illustrates the standard layout and GUI components of SPtool as well as a sample environment constructed with the software.

Once constructed, users add and orient virtual video sensors within the 3D environment. Analysis and visualization of the sensor coverage – a measure indicating how much or how well the environment is observed by the sensor(s) – is automatically performed to aid users in determining the most effective sensor layout.

In the following section, I describe the coverage analysis and visualization algorithms implemented in SPtool, and in Section 3, I provide a discussion of SPtool and potential extensions for making the techniques discussed here more applicable to sensor placement tasks in general.

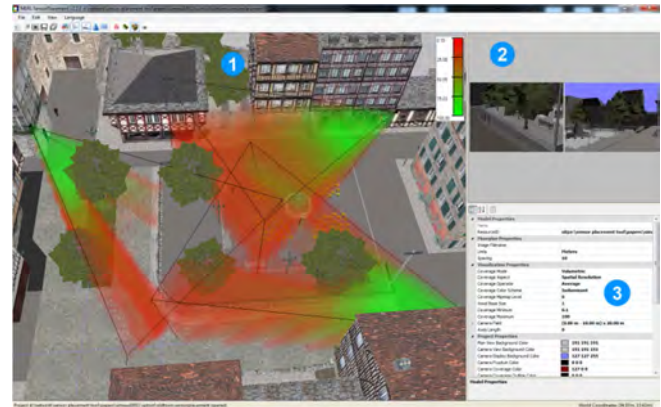


Figure 1. Illustration of the SPtool GUI and the components that comprise it. Component 1 is the primary view window, where users interact with the environment and video sensors. Component 2 provides a virtual view from sensors within the environment. Finally, component 3 is a properties grid, where users can customize the parameters of selected 3D models, selected video sensors, environment parameters, or the sensor coverage analysis and visualization.

2. CONSTRUCTING THE SENSOR LAYOUT

Following construction of the environment, users place and orient video sensors in the environment using simple GUI operations. Parameters of the camera(s) (e.g., sensor width or image dimensions) and lens (e.g., focal lengths) are manipulated using the properties grid.

2.1. Coverage Computation

Previous approaches to computing coverage by a set of video sensors have largely computed coverage as the percentage of a 2D floorplan observed by one or more sensors. This can lead to drastic errors between the computed coverage and the intuitive sense of coverage due to such considerations as partial occlusions (e.g., cubicles) or monitoring height (e.g., it may be critical that the sensors selectively observe the faces of people in the monitored area). As such, I have developed a method to compute 3D sensor coverage that automatically handles such considerations, which I briefly describe in the following paragraphs.

Figure 2 illustrates the rough steps of the algorithm used in the present work to compute sensor coverage. The first step in coverage analysis, is to compute a bounding box that entirely encloses the target environment (Figure 2a; environment geometry is omitted for clarity). The bounding box is then divided into a regular grid of voxels (Figure 2b; voxel sizes are exaggerated for clarity), which store raw sensor coverage data for computing quantitative measures of

sensor coverage. In order to determine the raw data stored in the voxel, I employ a straightforward ray-casting algorithm based on the geometry of image formation.

As is done with traditional ray-casting rendering techniques, a single ray is projected (Figure 2c), for each pixel, from the focus of the sensor, through the environment, to the point where it either intersects an object in the environment or exits the bounding box. If the ray were part of a rendering algorithm, it would perform scattering and integration of light to determine each pixel's final color value. However, it is only necessary here to determine which voxels it traverses (Figure 2d) before either intersecting an object or exiting the environment. Many algorithms exist for tracing a ray through a regularly sampled volume (e.g., Cleary and Wyvill 1988). A nice side effect of using this algorithm for computing coverage, is that occlusions are implicitly handled, as contributing rays are eliminated as they come into contact with objects in the environment.

2.2. Coverage Visualization

Raw coverage data stored in each voxel are then used to compute a coverage measure indicating how well the voxel is covered by the sensors in the layout. In particular, SPtool includes three such measures described individually below.

Spatial Resolution – number of rays intersecting a voxel; analogous to the number of pixels that observe the voxel.

Camera Count – number of cameras that have at least one pixel observing the voxel.

Angular Coverage – range of angles from which the voxel is observed; see Huang and Tseng (2003) for related work.

Once the coverage has been quantized into one of these measures, it can be mapped into a normalized value by having the user supply the minimum and maximum acceptable coverage (i.e., values normalized to the range [0-1] using this min and max are considered acceptable coverage). To indicate the level of coverage to the user, I employ two general techniques. In the simplest technique, voxels within the acceptable range are rendered as semi-transparent, color-mapped boxes, which I will refer to as the *volumetric* visualization. An alternative, *projective* visualization technique involves creating a coverage

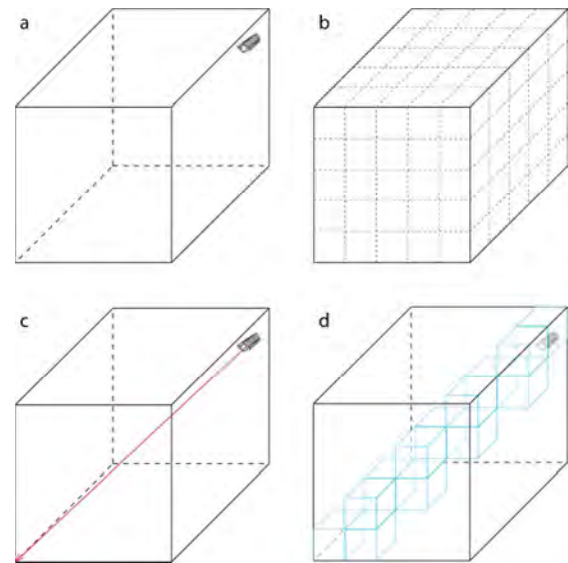


Figure 2. Sequence of steps comprising coverage analysis in SPtool. (a) A bounding box of environment, which contains a single video sensor, is computed first. (b) The bounding box is divided into a regular grid of voxels, which store the raw coverage data. (c, d) Rays from video sensors are traversed through the voxels, contributing to data stored in each voxel they intersect.

overlay, where the color that is mapped to the overlay at each individual pixel indicates the level of coverage along that pixel's line of sight.

Each of these visualizations has strengths and weaknesses when conveying sensor coverage to the user. The volumetric visualization conveys sensor coverage of individual voxels within the environment. Generally, the user can manipulate the view to determine the level of coverage for all 3D locations; however, this can be quite time consuming and often, this level of detail is not needed. In contrast, the projective visualization combines the coverage from multiple voxels along a single line of sight to create a synthetic overlay. This visualization can be used to confer a simple, high-level overview of the level of coverage achieved over a large area, and is best used when viewing the environment from afar – particularly useful for orthographic projections of the environment.

3. DISCUSSION

SPtool provides a useful interface for accomplishing one primary task, quickly developing effective video sensor layouts for a target environment. A user familiar with SPtool can build a moderately complex environment and video sensor layout and estimate its effectiveness (eg, coverage of monitored areas) and cost in well under one

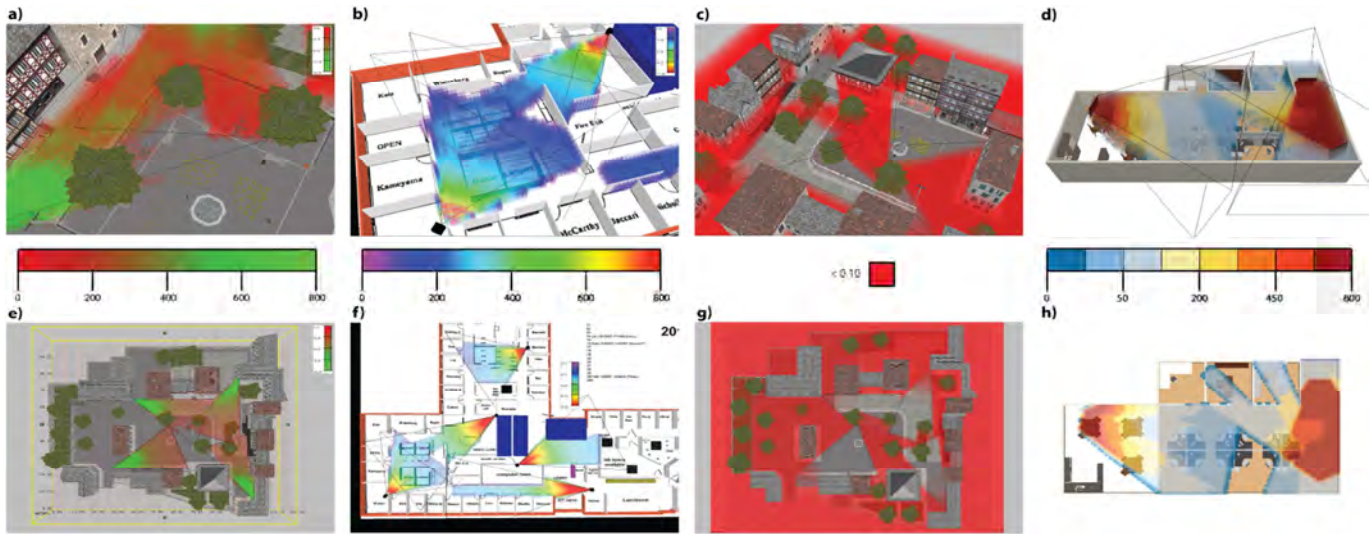


Figure 3. Examples of 3D environments and sensor coverage produced in SPTool. a-d) Volumetric visualization of coverage. e-h) Projective visualization of coverage. Panels c and g illustrate 'holes' in the coverage, where coverage does not meet the minimum coverage level set by the user.

hour. Increases in the required level of accuracy and environment detail will, of course, affect development time.

In general, though, the techniques described here are not limited to surveillance networks alone or to the implementation described here. For instance, the algorithm for determining sensor coverage presented here is applicable to any sensor that can have its sensing elements reasonably extended into space. For instance, SPTool has additionally been used for creating LIDAR sensor layouts in addition to traditional video sensor layouts, and may be applicable to pervasive sensors, such as passive infrared sensors.

Furthermore, the methods presented here provide an excellent starting point for constructing more complex surveillance tools. For instance, in previous work regarding coverage analysis using 2D representations of the environment, (Erdem and Scarloff 2004; Murray et al. 2007; Yabuta and Kitazawa 2008), the computed coverage was used to compute an optimal sensor layout without user intervention. Since these algorithms operate over a 2D space, the computed sensor layouts can only be considered an approximation. Voxel representations of coverage, such as the one employed in the present work, form an ideal basis over which coverage, which is already stored per voxel, can be optimized using only modest revisions to the 2D optimization algorithms.

References

BECKER, E., GUERRA-FILHO, G. AND MAKEDON, F. 2009. AUTOMATIC SENSOR PLACEMENT IN A 3D VOLUME. *WORKSHOP ON APPLICATIONS OF COMPUTER VISION (WACV)*.

CLEARY, J. G. AND WYVILL, G. 1988. ANALYSIS OF AN ALGORITHM FOR FAST RAY TRACING USING UNIFORM SPACE SUBDIVISION. *THE VISUAL COMPUTER* 4, 65-83.

ERDEM, U. M. AND SCARLOFF, S. 2004. AUTOMATED CAMERA LAYOUT TO SATISFY TASK-SPECIFIC AND FLOORPLAN-SPECIFIC COVERAGE REQUIREMENTS. *COMPUTER VISION AND IMAGE UNDERSTANDING* 103, 156-169.

VAN DER HENGEL, A., HILL, R., WARD, B., CICHOWSKI, A., DETMOLD, H., MADDEN, C., DICK, A. AND BASTIAN, J. 2009. AUTOMATIC CAMERA PLACEMENT FOR LARGE SCALE SURVEILLANCE NETWORKS. *2ND INTERNATIONAL CONFERENCE ON PERVERSIVE TECHNOLOGIES RELATED TO ASSISTIVE ENVIRONMENTS (PETRA)*.

HUANG, C. AND TSENG, Y. 2003. THE COVERAGE PROBLEM IN A WIRELESS SENSOR NETWORK. IN *ACM INTERNATIONAL WORKSHOP ON WIRELESS SENSOR NETWORKS AND APPLICATIONS (WSNA)*, 115-121.

MCCAHL, M., NORRIS, C., AND WOOD, D. 2002. EDITORIAL. THE GROWTH OF CCTV: A GLOBAL PERSPECTIVE ON THE INTERNATIONAL DIFFUSION OF VIDEO SURVEILLANCE IN PUBLICLY ACCESSIBLE SPACE. *SURVEILLANCE & SOCIETY* 2, 110-135.

MURRAY, A. T., KIM, K., DAVIS, J. W., MACHIRAJU, R., AND PARENT, R. 2007. COVERAGE OPTIMIZATION TO SUPPORT SECURITY MONITORING. *COMPUTERS, ENVIRONMENT AND URBAN SYSTEMS* 31, 133-147.

O'ROURKE, J. 1987. *ART GALLERY THEOREMS AND ALGORITHMS*. NEW YORK: OXFORD UNIVERSITY PRESS.

YABUTA, K. AND KITAZAWA, H. 2008. OPTIMUM CAMERAPLACEMENT CONSIDERING CAMERA SPECIFICATION FOR SECURITY MONITORING. *IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS*, 2114-211.

A New System Dynamics Framework for Modeling Behavior of Vehicle Sharing Systems

Dimitris Papanikolaou

Media Laboratory Massachusetts Institute of Technology
75 Amherst Street,
Cambridge, MA, USA, 02139
dimp@mit.edu

Keywords: System Dynamics, Resource Allocation Networks, Diffusion Models, Vehicle Sharing Systems, Heterogeneity.

Abstract

One-way vehicle sharing systems are convenient mobility systems consisting of parking stations and a fleet of shared vehicles. Users can pick up a vehicle from any station and drop it off to any other station. However, due to asymmetric demand patterns eventually all vehicles end up at the stations with no demand, decreasing throughput performance. This paper presents an ongoing research for a new computational framework in System Dynamics that describes distribution of resource allocation in a vehicle sharing system under non homogenous demand patterns by simulating the resource flow between areas of high density to areas of low density as demand pattern changes. The framework will be used as study tool to understand behavior and explore organizational solutions.

1. INTRODUCTION

Today's urban mobility schemes face problems that limit their efficiency. On one hand, increasing land prices and vehicle ownership costs compared to the low utilization rates makes private vehicle ownership an unsustainable solution. For example, in the US the average household has nearly 2 vehicles, which spend around 90% of their time parked while they require more than twice their footprint in urban land to be able to travel from an origin to a destination (National Household Travel Survey, 2001). On the other hand, poor service coverage, schedule inflexibility, and lack of personalization, make public transit a suboptimum compromise.

1.1. Vehicle sharing

Vehicle sharing is an effective method for increasing utilization of privately driven vehicles in a public manner. A vehicle sharing system consists of a decentralized network of parking stations and a fleet of shared vehicles (figure 1). Users can pick up a vehicle from any station and drop it off to any other station (one-way trips). Existing examples are bike sharing programs; however this trend is now entering automobile markets too.

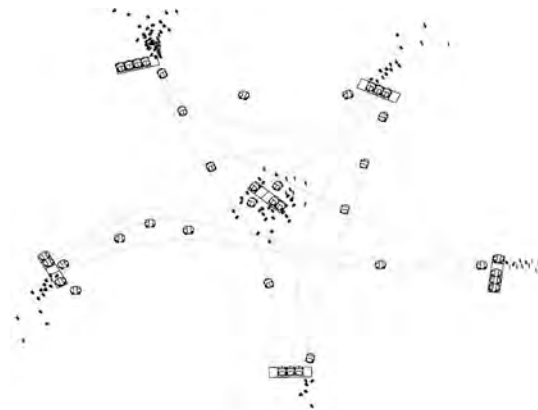


Figure 1. A vehicle sharing system.

Despite their great convenience vehicle sharing systems have significant drawbacks. Due to uncorrelated departures and arrivals at the stations, soon some stations end up having no vehicles while other stations end up having no parking spaces (Kaltenbrunner et al. 2008). This inventory imbalance, not only decreases throughput, but it furthermore increases trip time as drivers search for parking availability. To maintain level of service a vehicle sharing system needs to constantly feed origin points with vehicles while drain destination points from occupied parking spaces. Existing fleet management strategies achieve this by empirically redistributing vehicles among the stations using trucks,

which is a complex, inefficient and highly unsustainable solution.

Understanding the dynamics of resource allocation in vehicle sharing systems is essential not only for controlling and sizing existing infrastructures but also for exploring potential impact of new organizational policies. Often some of the questions of significant importance are: How does throughput performance depend on the demand pattern? How will the resources be allocated during its operation? How will trip time be affected from overflowing effects? In this paper we present the current progress on a new System Dynamics framework that aims to address these questions by dynamically describing resource allocation in vehicle sharing systems with limited capacity. The framework's main purpose is to be used for exploring management policies and assessing ability of sharing systems to deliver service under different demand pattern scenarios, however we expect it will have potential applications in many domains.

1.2. Understanding the problem

Throughput performance decreases either because departure rate drops due to unavailability of vehicles at origins or because trip time increases due to unavailability of parking spaces at destinations. The following causal loop diagram in System Dynamics illustrates this observation.

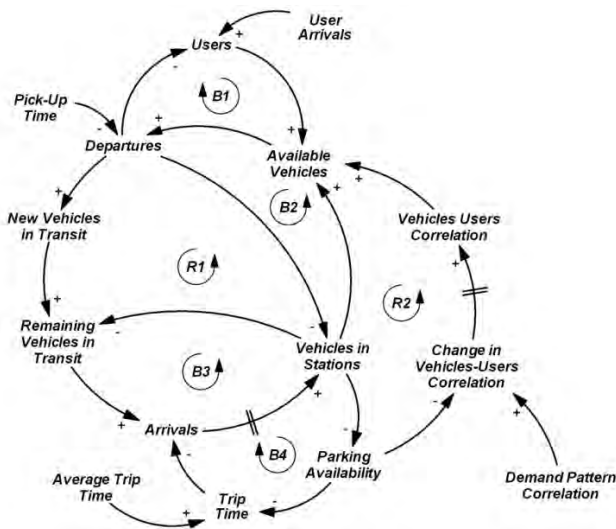


Figure 2. A causal-loop diagram of the problem.

In equilibrium all inventory stocks remain stable so sources and sinks are indistinguishable. During disequilibrium however, vehicles are reallocated from areas with higher

density in users than vehicles (sources) to areas with lower density in users than vehicles (sinks). This process is a form of spatial diffusion.

2. RELATED WORK

2.1. Diffusion models

Diffusion models describe the dissipation of a resource from areas of high density to areas of low density. Diffusion rate is relative to the distribution of the density of the resource in the area and therefore ceases when density is the same everywhere. As an example consider the dissipation of a highly pressured gas in a void room. As the pressure of gas is higher than that of the air, the gas will continue dissipating in the room. Eventually dissipation stops when gas and air have the same pressure. Diffusion processes have been studied by two pioneering fields in modeling and simulation of complex systems; System Dynamics (SD) and Agent Based Systems (AB).

2.2. System Dynamics and Agent Based Systems

System Dynamics is a method for studying, designing, and managing complex feedback systems by modeling their *macroscopic* structure through causal loop diagrams and simulating their behavior through hydraulic compartment models (stock-flow models) in a top-down manner (Forrester, 1969). SD models are deterministic continuous compartment models, working with differential equations. The great strength of the SD methodology is its high abstraction, educational clarity, and computational robustness. Agent Based modeling on the other hand simulates dynamics of complex systems by replicating the *microscopic* behavior of discrete individual agents through behavioral rules and running the model in a bottom-up manner, without explaining the structure of it, lacking educational clarity as a decision making tool. Furthermore AB is computationally intensive and does not provide a solution for the full range of the solution space (Rahmandad and Sterman 2006). For the above reasons SD is chosen as the modeling approach for this study while we use AB as a means to evaluate our results.

2.3. Modeling diffusion in System Dynamics

Diffusion processes are typically modeled in SD literature by cascading compartments that describe different states of the resource and simulating the diffusion rate between them (Figure 3). Compartment models have applications to many problems due to their simplicity,

reliability, and robust behavior, such as the famous SEIR model for propagation of epidemics or the Bass model for diffusion of market innovation (McKendrick, 1925; Bass, 1969).

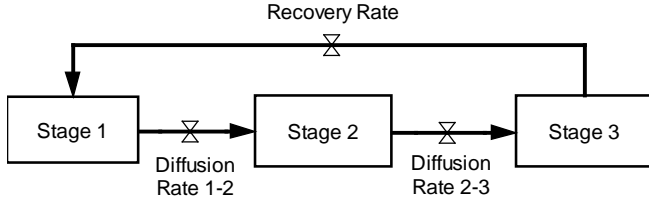


Figure 3. Modeling diffusion as cascaded compartments in SD

Since compartments represent different states of the flowing resource rather than what portion of space these states occupy respectively, most compartment models fail to endogenously describe the impact of heterogeneity of distribution in the behavior of the model; instead, when this is necessary, heterogeneity is described as an exogenous variable (Rahmandad and Sterman 2006). In most cases where heterogeneity is not the main diffusion factor this is an acceptable assumption. In a spatial diffusion however like our case, the diffusion rate depends primarily on the spatial correlation of the distributions of users and vehicles at the stations, and moreover on how many total inventory capacity source and sink areas include: since a trip needs both a vehicle and a user to coexist in a station, diffusion ceases when there are no more vehicles at the stations where users wait, or alternatively when there are no more users at the stations where vehicles are parked.

We therefore need a simple yet effective way to describe heterogeneity in a compartment model. In practice, people describe intuitively distributions with phrases such as “81% of drop-off demand concentrates on 22% of the stations”. Our model should be able to both capture intuition accurately and provide enough control to simulate different scenarios. Such a model should clearly describe what the threshold *boundary* between the high and low density is and how this boundary changes in time.

3. DESCRIBING A DISTRIBUTION AS TWO HOMOGENOUS GROUPS

Statistically, distributions are described using derivatives of standardized moments (e.g. the averaged sum of the n^{th} power of the difference of each sample from the mean); the more the standardized moments are, the better the description of the distribution is. In practice, most

descriptions use only the first three standardized moments (e.g. the mean, variance, and skewness). This is the same as approximating the discrete distribution of n samples with a continuous distribution of only two values above (Y_1) and below (Y_2) the mean such that the mean, variance, and skew are the same. Practically we are looking for X_1 , X_2 , Y_1 , and Y_2 if we know the mean, variance, and skewness of a distribution (Figure 4).

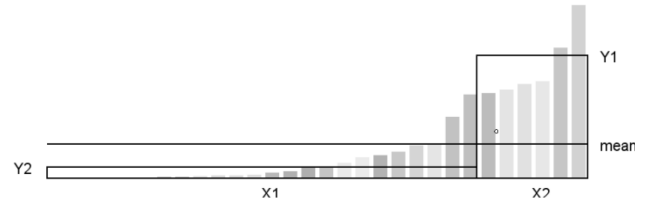


Figure 4. Simplification of a distribution

From the first three standardized moments we get the following four equations:

$$\mu_1(y) = \frac{(y_1 \cdot x_1 + y_2 \cdot x_2)}{x_1 + x_2} \quad (1)$$

$$\mu_2(y) = \sqrt{\frac{(y_1^2 \cdot x_1 + y_2^2 \cdot x_2)}{x_1 + x_2}} \quad (2)$$

$$\mu_3(y) = \sqrt[3]{\frac{(y_1^3 \cdot x_1 + y_2^3 \cdot x_2)}{x_1 + x_2}} \quad (3)$$

$$x_1 + x_2 = \text{sample population} \quad (4)$$

Solving the system of equations we get the following for x_1 , x_2 , y_1 , y_2 :

$$Y_1 = \frac{1}{2} \frac{1}{\mu_2(y)} \left((x_1 + x_2) \mu_3(y)^3 \pm \sqrt{(x_1(y) + x_2(y))^4 \mu_3(y)(y)^6 + 4(x_1(y) + x_2(y)) \mu_2(y)^3} \right) + \mu_1(y)$$

$$Y_2 = \frac{\mu_3^3(y) \cdot (x_1 + x_2)}{\mu_2(y)} - Y_1 + 2 \cdot \mu_1(y)$$

$$x_2 = \frac{\mu_2(y)}{(Y_2 - Y_1) \cdot (Y_2 - \mu_1(y))}$$

1990

Number of Children	Percentage
0	0
1	0
2	0
3	1
4	1
5	2
6	2
7	23
8	8
9	17
10	47
11	76
12	69
13	26
14	54
15	41
16	25
17	7
18	7
19	8
20	9
21	12
22	16
23	20
24	29
25	20
26	13
27	7
28	3
29	2
30	1

2010

Number of Children	Percentage
0	0
1	0
2	0
3	1
4	1
5	2
6	2
7	2
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2
16	2
17	2
18	2
19	2
20	2
21	2
22	2
23	2
24	2
25	2
26	2
27	2
28	2
29	2
30	2

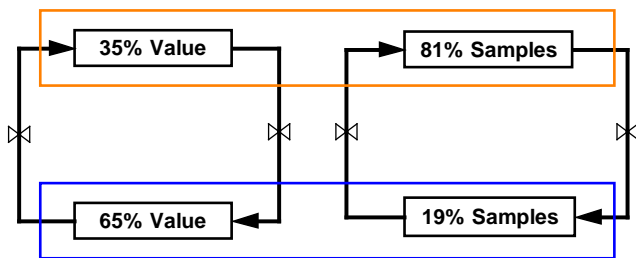
65 %

35 %

81 %

19 %

Figure 5 shows a distribution of samples with their average value, the same distribution sorted from highest to lowest, and its simplified description: 65% of value concentrates on 19% of samples while the rest 35% of value concentrates on 81% of samples. Figure 6 shows the same description as a stock-flow model: the circuit on the left represents the segregation of aggregate value into the two groups of samples while the circuit on the right represents the segregation of the samples.



However in our case throughput performance depends on the correlation, or overlapping, of two value distributions (vehicle and user inventory levels) over a group of samples (stations). We therefore need to introduce a third stock-flow circuit representing the second value distribution and correlate it with the existing one.

Consider an urban area with parking stations that share a common fleet of vehicles. Stations have known inventory capacity and average pick-up time. Users arrive at the origin stations, pick-up an available vehicle, and drive it to their desired destinations following a given demand pattern. If

Throughput performance is the percentile ratio of the current throughput rate to the maximum throughput rate according which the system can serve clients (capacity rate). Capacity rate puts thus a ceiling in measuring performance: demand that exceeds capacity cannot be used for evaluating performance since obviously the system cannot fulfill it.

We define a *demand pattern* as a pair of time-based distributions of pick-up and drop-off requests at the stations. Demand patterns allocate resources (vehicles) in the system depending on the correlation of these two distributions. Demand patterns therefore describe the *rate of change* of the distribution of the resources in the system. For example, in a highly correlated demand pattern for each station for each pick-up there is one drop-off. On the other hand, in a highly uncorrelated demand pattern massive drop-offs or pick-ups are persistently concentrated on few stations causing unbalanced fleet transfers, delays, and overflows. Service rate depends on the pick-up time, the accumulated demand, and the stock of available vehicles. For each station service rate is:

4.1. Initial equilibrium state

Suppose that the system starts in optimum equilibrium. In this ideal state, the demand pattern is such that in every station, for each outgoing trip there is an incoming trip, and as a consequence inventory levels remain stable. Vehicles stock is segregated into *vehicles in transit* and *vehicles in*

stations according to the ratio of the *average trip time* versus *pick-up time*. For example, if average travel time is 13min and the average pick-up time is 2min, then during equilibrium the fleet would be segregated into 86.6% vehicles in transit and 13.4% vehicles in stations.

4.2. Disequilibrium

At some point in time suppose that the demand pattern changes in a way that now some stations have persistently higher vehicle arrivals than user arrivals, while some other stations have persistently lower vehicle arrivals than user arrivals (yellow and blue groups in figure 7). Let us call A the first group and B the second. Obviously, inventories in A will be increasing (sinks) and user queues will be decreasing, while inventories in B will be decreasing (sources) and user queues will be increasing. Since departures rate depends on the minimum of users and vehicles waiting at the stations (there needs to be at least a vehicle and a user to have a departure) it turns out that throughput performance drops in both groups. Concurrently, a massive resource reallocation occurs: users are moving from A to B, while vehicles are moving from B to A.

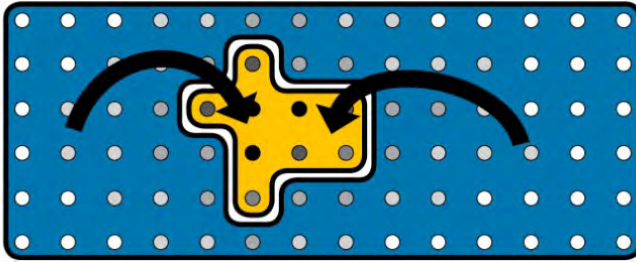


Figure 7. The system during disequilibrium. Stations are segregated.

4.3. Shift of disequilibrium

Suppose now that the demand pattern changes again such that group A is now changing in shape, number, and location (figure 8, top). As A and B change, a mutual exchange of stations between them occurs (figure 8, left); if the A expands geographically, then stations that were previously inside B are now entering A (figure 8, middle); if instead the A contracts, then stations from it will flow towards B; if however it simply moves or changes shape without changing in size then there is a mutual inflow and outflow of stations between A and B as the boundary changes (figure 8, right). It should be obvious that as stations enter and exit each group the aggregate levels of both users and vehicles in that group change too. For example, if a station changes from B to A, then the total

inventories and users in B will decrease, while in A will increase as defined by the average levels on each group.

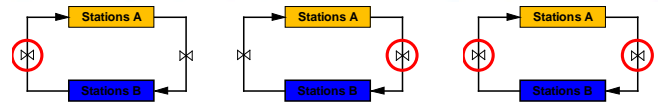
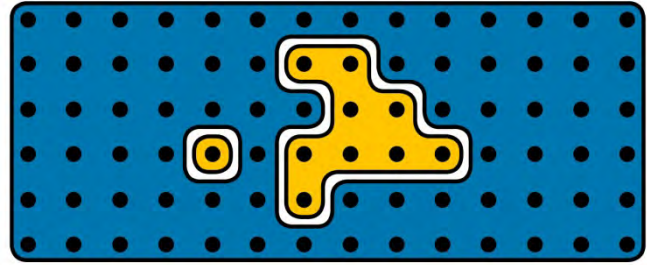


Figure 8. Transformation of groups A and B as demand pattern changes

4.4. Overflow of sinks

If demand pattern continues unchanged, soon A will start depleting from vehicles, while B will start depleting from parking spaces. When this happens, arriving vehicles to A that find no parking will drive to the closest available station towards B. As a consequence, vehicle arrivals in B will tend to increase relative to the departures, while the vehicle arrivals in A will tend to decrease relative to the departures. This behavior will increase the average trip time of trips from B to A, increase the vehicles in transit, and shift the boundary between the two groups: A will tend to expand while B will tend to contract. The increase in time depends on the length of the border between the two groups and the density according which stations are scattered in space (figure 9). For example, if it turns out that A will require x amount of additional stations from B to accommodate the overflowing demand, then the increase in linear distance will depend on the ratio of the required urban area to the current boundary length: clustered disperse patterns with long boundary lines should have less increase in time than centralized congested areas.

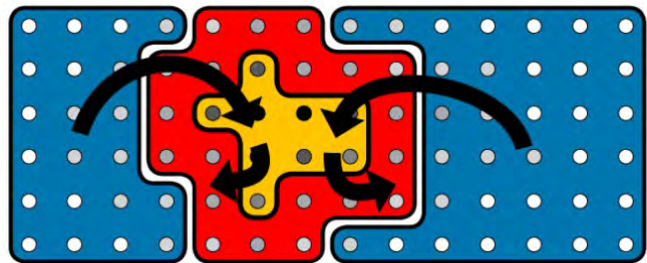


Figure 9. Overflow of stations with high drop-off demand. Border expansion.

4.5. New equilibrium state

Eventually the system reaches again a new suboptimum equilibrium state. In this new equilibrium state, less portion of the demand rate in A will find vehicles, but the overall average trip time will have increased. Throughput rate will be lower, average trip time will be higher and fleet in transit will be lower: fewer vehicles in transit will be traveling more. Depending on the demand pattern, the equilibrium level fluctuates continuously. The time that it takes the system to reach this new equilibrium depends on the average trip time and the stock in transit. Larger transportation networks in which average trip distances are higher will have longer delays than smaller systems.

5. STOCK-FLOW MODEL

To describe the correlation of users and vehicles at the stations we have built a model in *Vensim*, a popular software package for System Dynamics modeling, consisting of three stock-flow subsystems: *Stations*, *Users*, and *Vehicles*, each segregated into group A (top) and group B (bottom), and about 120 equations. The *Stations* model describes the spatial segregation of the two groups of the demand pattern. The *Users* model describes the user levels, arrivals and departures in group A and group B. The *Vehicles* model describes the vehicle levels, arrivals and departures in group A, group B, and in transit (figures 10 and 11). Throughput depends on the correlation of vehicle and user stocks while parking capacity and additional delays depend on the stations stock. The controlling parameters are the following: Portion of gross user demand rate to group A (and B); portion of gross vehicle arrivals rate to group A (and B); Average Trip Time; Average Pick-up Time; Change Rate of groups A and B. All other parameters are computed endogenously in the model.

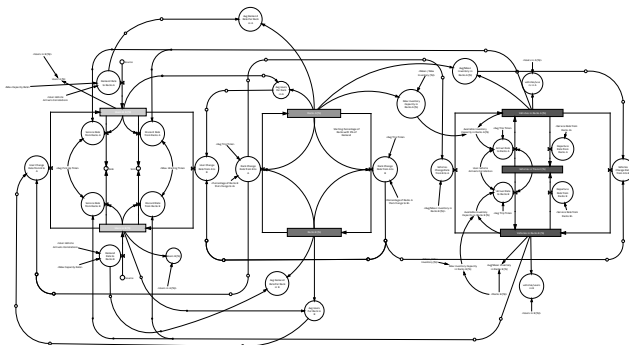


Figure 10. Working System Dynamics model build in Vensim.

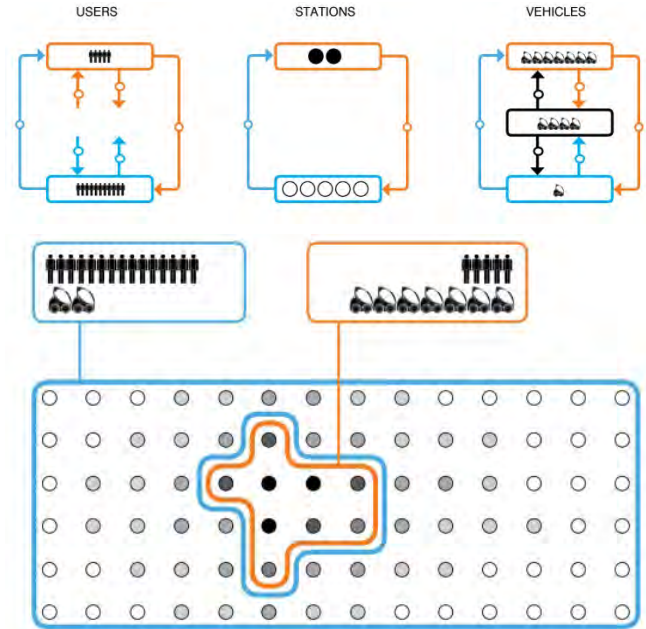


Figure 11. The model as a stock-flow diagram.

5.1. Stations

The *Stations* model consists of two stocks of stations representing group A, and group B, connected by two unidirectional flows representing the flow of stations between the two groups as demand pattern changes spatiotemporally. If parking spaces in group A start depleting, then arriving vehicles to group A will look for the closest available destination station. This behavior will tend to increase group A while decrease group B, shifting the border between them. The average flow of stations from group B to group A will be determined by the unmet demand for arriving trips (e.g. portion of arriving trips to group A that don't find available parking), the available parking per station in group B, and the average trip time to travel between two neighbor stations (closest distance). While *Desired Trip Time* is an exogenous input parameter in the system actual trip time will be determined by additional delays due to inventory saturation in the system. When inventories in group A start depleting, the aggregate vehicles arriving in group A will be continuously correcting their trip by adding extra distance. The correction rate per each trip will be defined by the additional distance to find a new parking station (which is the average nearest neighbor distance of the network), divided by the current average trip time (which will be constantly changing). The current average trip time will equal the desired trip time plus additional time delay. The additional time delay depends on the *Excess Trip Distance* which is a stock that is modified

by the *Trip Distance Change Rate*. Trip Distance Change Rate depends on the percentage of arriving trips that don't find parking space, the average excess distance (which is the average nearest neighbor distance of the network), and the average trip time which will be constantly changing.

5.2. Users

The *Users* model consists of 2 stocks, *Users A* and *Users B*, containing the users that are currently queued in the stations of each group. Each stock increases by the portion of the *Gross Demand Rate* that goes to the stations of that group, while decreases by the *Service Rate* and *Discard Rate* from these stations. The *Gross Demand Rate* is equal to the *System Capacity Rate*, while the portion of it that goes to each group depends on the input demand pattern characteristics. Since a departure can happen only if there is at least a user and a vehicle in an origin station, the *Service Rate* depends on the minimum of *Users* and *Vehicles* of each group. The *Discard Rate* depends on the stock of waiting users and the *Max Waiting Time*. As demand pattern changes, stations circulate from one group to the other taking with them the users that are currently accumulated in these stations. For example, if some stations move from group A to group B because of a demand pattern change, the users that will flow from *Users A* to *Users B* will depend on the *Average Users per Station* in A and the number of stations that flow from *Stations A* to *Stations B*.

5.3. Vehicles

The *Vehicles* model consists of 3 stocks: 2 stocks containing the vehicles in the stations in groups A and B (*Vehicles A*, *Vehicles B*), and one stock containing the vehicles in transit (*Vehicles in Transit*). As vehicles depart from each group they enter the transit stock and return back to one of the two groups according to the demand pattern. The time they spend in the transit stock depends on the *Average Trip Time* plus the additional trip time delay. In this model, we use a 1st order delay for the vehicles arrivals using the average trip time. The *Departures Rate* from each group is the same as the *User Service Rates* and depends on the minimum of *Users* and *Vehicles* and the *Pick-Up Time*. The *Arrivals Rate* to each group depends on the portion of the *Vehicles in Transit* that return to that group, and the *Average Trip Time*. As demand pattern changes, stations circulate from one group to the other taking with them the vehicles that they contain in their inventories. For example, if some stations move from group A to group B because of a demand pattern change, the vehicles that will flow from

Vehicles A to *Vehicles B* will depend on the Average Inventory level of *Stations A*, and the number of stations that flow from *Stations A* to *Stations B*.

5.4. Model behavior and early simulation results

To focus on the impact of demand pattern distribution characteristics, we start the model in equilibrium using the maximum capacity rate. At some point ($t=5$) we introduce a pulse change in the demand pattern such that 20% of the stations (group A) concentrate 60% of the demand rate for 2 time units (out of the total 24 of the simulation time), after which it returns to its normal uniformly distributed state. Figure 12 shows the throughput performance of group B (blue), the increase in trip time (red) of the trips to group A, the vehicle inventories at A and the available parking at A. Although the model needs further improvement and calibration, early simulation results show that it models increase in trip time as demand pattern becomes more asymmetric.

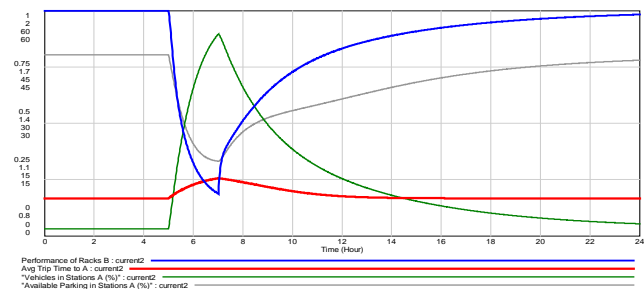


Figure 12. Simulation results showing increase of trip time (red).

6. LIMITATIONS

There is a tradeoff between control and accuracy and as a consequence our approach has several limitations. First of all, our model currently segregates exogenously the stations according to the correlation of the user and vehicle arrival rates (flows), as opposed to the correlation of the reallocated stocks. Since there is a significant lag between these two correlations (with stock correlation following flow correlation) and since actual throughput depends on the stocks correlation rather than the flows correlation, the two groups don't represent the most accurate way to simulate performance. A future possible approach would be to develop the SD framework such that it would endogenously segregate stations according to stock distribution. However for cases where trip time is small relative to the demand pattern changing time, this is an effect with negligible impact on performance. A second limitation, inherent to all

SD continuous forecasting models, is the simplification of behavior due to the mixing that occurs when aggregating resources in the stocks, that tends to drift the results as simulation time advances sometimes overestimates performance. Again, that effect can be minimized by carefully selecting the trip time and demand pattern change rate. Furthermore, our model represents all trips as having the same average trip time, pick-up time, and maximum waiting time which microscopically might make little sense, however macroscopically is a reasonable assumption. Generally the discussed SD model provides less accurate results but a much more robust control in the inputs.

7. FUTURE VALIDATION METHODS

We consider three future validation methods. As a first validation method we are building a second simulation model in Processing (Java) based on a combination of System Dynamics and Agent Based systems. The model uses the same equations as the SD model in Vensim but instead it models each station individually as opposed to aggregate approach of Vensim: it takes as input two controllable discrete distributions, one representing inflow of users and the other representing the vehicle arrivals over the stations, and simulates the behavior of the system by correlating the distribution of users and vehicles stocks. Vehicles are departing individually from each station, aggregate in the *stock in transit*, and return back to their destinations after having spent the average time in the *stock in transit*. This model has a more accurate behavior but a less controllable interface (figure 13). A second validation method is to build an Agent Based model of the system and compare the result with the Vensim model. This approach will give pseudo-accurate deterministic results with very limited interface control since each user, vehicle, and station will have to be programmed individually. One possible approach for this direction is to first run a simulation in the AB model recording both input parameters and output results, simplify the input parameters with the method described in this paper, and then use them as input in the SD model to compare the results. A third validation approach is to conduct an experiment by asking users to physically reallocate widgets on containers on a table following a given instruction set, record the results, and run a simulation with the same parameters, comparing at the end results from both sides.

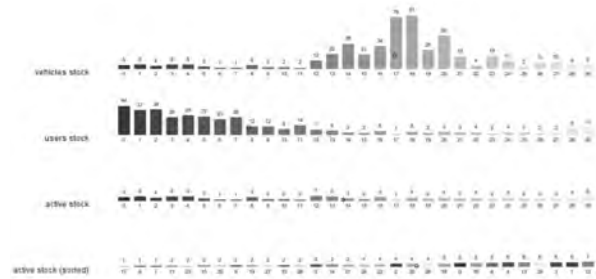


Figure 13. Programmed simulation model in Processing.

8. CONCLUSIONS

In this paper we discussed an ongoing research for a new framework in System Dynamics for understanding behavior of unidirectional vehicle sharing systems and the dependence of throughput performance on demand pattern characteristics and fleet size. Preliminary results show that the framework can model delays and captures essential dynamic behavior that is compatible with intuition and observed behavior of real systems. Next steps are completion and calibration of the model, and comparison of results with data from experiments that are yet to be conducted.

Acknowledgements

The author wishes to acknowledge Prof. William J. Mitchell, Kent Larson, and Prof. John Sterman for their insightful guidance as well as the Smart Cities group at the MIT Media Lab for all the help, support and collaboration.

References

- Bass, F. M. 1969. A new product growth for model consumer durables. *Management Science* 15 215-227.
- Forrester, J. (1969). *Urban Dynamics*. Pegasus Communications, Inc.
- Kaltenbrunner, A., Meza R., Grivolla J., Codina J., Banchs R. 2008. Bicycle cycles and mobility patterns - Exploring and characterizing data from a community bicycle program. <http://arxiv.org/abs/0810.4187>.
- McKendrick A.G. 1925. Applications of mathematics to medical problems. *Proceedings of the Edinburgh Mathematical Society*
- Mitchell W., Burns L., and Borroni-Bird C. (2010). *Reinventing the Automobile: Personal Urban Mobility for the 21st Century*. MIT Press.
- Rahmandad, H. and J. D. Sterman. 2006. Heterogeneity and Network Structure in the Dynamics of Contagion: Comparing Agent-based and Differential Equation Models. Under review in *Management Science*.
- U.S. Department of Transportation (2001). National Household Travel Survey: Summary of Travel Trends, 11-15.

Session 3: Implementation

45 **GRAPE: Using graph grammars to implement shape grammars**

THOMAS GRASL and ATHANASSIOS ECONOMOU
SWAP [+] and Georgia Institute of Technology

53 **Automated Energy Model Creation for Conceptual Design**

LILLIAN SMITH, KYLE BERNHARDT and MATT JEZYK
Autodesk

61 **An Integrated Approach to Algorithmic Design and Environmental Analysis**

ROBERT AISH and ANDREW MARSH
Autodesk

69 **Leveraging Cloud Computing and High Performance Computing Advances for Next-generation Architecture and Urban Design Projects**

FRANCESCO IORIO and JANE L. SNOWDON
Autodesk Research and IBM T. J. Watson Research Center

GRAPE: Using graph grammars to implement shape grammars

Thomas Grasl¹, and Athanassios Economou²

¹SWAP [+]
Lange Gasse 16/12,
1080 Vienna, Austria
tg@swap-zt.com

²Georgia Institute of Technology
245 4th St. NW,
Atlanta, GA, U.S.A., 30332
economou@coa.gatech.edu

Keywords: Shape, graph, grammar, implementation.

Abstract

An implementation of a shape grammar interpreter is described. The underlying graph-theoretic framework is briefly discussed to show how alternative representations from graph theory including graphs, overcomplete graphs and hyperedge graphs can support some of the intuitions handled in shape grammars by direct visual computations with shapes. The resulting plugin implemented in Rhino, code-named GRAPE, is briefly described in the end.

1. INTRODUCTION

Shape grammars provide a powerful generative approach for the description, interpretation and evaluation of existing designs as well as new designs (Stiny and Gips 1972; Stiny 2006). A nice account of various applications of shape grammars for the study of existing and new languages of design can be found in Knight (2003).

Still, while the shape grammar formalism has produced a long and excellent series of formal studies the corresponding efforts for automation of the tasks the shape grammars accomplish have not met the same success. Recent summaries of such approaches are given in various sources (Gips 1999; Chau et al. 2004). Typically these efforts have been concentrated in shapes of specific geometry, for example, straight lines (Krisnamurti, 1981), polygonal shapes (Tapia, 1999) or parametric curves (Jowers and Earl, 2010).

One of the challenges for a computer implementation of a shape grammar is the design of an underlying formalism that will allow for symbol-based substitutions of any part of a shape that may be of use in a design setting. The specific

formalism used here to support such computations has been built upon graph theory. Our efforts have been focused on the possibility of mapping shape grammars into graph grammars, the advantage being that graph grammars are more amenable to computer implementation. Moreover existing graph grammar packages can be used to develop tailor-made software. Much of the more complicated computation, such as subshape recognition, is thereby taken care of.

Graphs are a popular choice for representing the structure of shape (Heisserman, 1994; Kelles et al., 2009), and graph grammars have been used to implement specific grammars by also mapping the rule set from shapes to graphs (Grasl and Economou, 2010). Typically graphs are seen as multidimensional data structures that with an appropriate embedding formally structure and visually resemble the shapes they represent. Still, the encoding of shapes in terms of these alternative graph representations is not straightforward. Various approaches to representing shapes as graphs are described here and one of them is selected as the most useful for a shape grammar interpreter. The resulting formalism uses two parallel representations, a graph theoretic which is used for the computation, and a visual one that illustrates the shape and which can be queried for geometric properties such as angles, lengths and intersections. Since all computation is done on the graph representation, the shape rules are mapped into graph rules and all the matching subgraphs are mapped back into shapes in order to display to the user. The formalism has been implemented in Rhino and it is code-named Grape to affectionately recall the parallel representations of GRaphs and shAPes.

2. TWO PUZZLES IN VISUAL COMPUTATION

The two key features that distinguish shape grammars from other generative models for the description of designs – and in fact the two characteristics that make shape grammars so deeply grounded in design discourse are: a) recursion; and b) embedding (Stiny 2006). Recursion allows a design to be described in terms of successive states that each of them has to be constructed to allow for the construction of the next one; in this sense, recursion picks up wonderfully the notion of process in a design setting where the making of a design is understood and modeled as an active talk – or back-talk if we take Schön's words (Schön 1983) between the designer and the design she makes. This characteristic is akin to the recursive definition of formulas in mathematics whereas any term in a sequence is defined in terms of the term preceding it – a process that requires a starting point and a set of rules to apply to generate the terms of the sequence. This mindset is quite distinct from the definition of the so-called closed or explicit formulas that are applied to give direct results; for example, to compute the fiftieth member of the Fibonacci sequence using a closed formula, the only thing that is needed is a substitution of the corresponding numbers in the explicit formula. The corresponding solution of the problem using the recursive formula requires fifty calculations starting from the first member and proceeding to the last. Still, this delay, so to speak, has clear advantages in a spatial setting because it clearly shows in steps the construction of a design and brings to the foreground the rules and conventions used to make this design. In this sense the challenge for the implementation of a shape grammar has clearly to do with the design of a program to support this recursive visual construction of a design; the typical contemporary discourse of the writing of a script (in text format) in some modeling environment, and then executing to see the visual results, creates a non-visual setting for making designs, requires an alienated setting for visual discourse and hides entirely from design inquiry the opportunities that arise in a seamless visual computation.

The second characteristic of a shape grammar, embedding, is paramount in visual composition and completely absent from computer aided design systems. Embedding is the relation that explains the wild, so to speak, nature of shapes, and their ability to constantly fuse and make new shapes too. But there is no miracle behind this, nor a suspense of natural laws; in symbolic computation any combination of discrete symbols such as

letters, notes or numbers will be always and unambiguously reproduced, read, performed or computed; in visual computation any combination of shapes, produce an incredible complexity of additional, diverse, emergent, of different topologies and shapes that were not used in the original combination. In this sense the challenge for the implementation of a shape grammar has clearly to do with the design of a program to support spatial computation and visual continuity while using the limitations of symbolic media to capture this continuity; the typical contemporary discourse of simulating rule-based design in some modeling environment is extremely limited, fosters blindness to designers, and creates a procrustean world view entirely antithetical to the design inquiry.

The key idea to address both these two challenges is the notion of an evolving, reconfigurable structure; a structure (parts and relations) is required for any shape to be retrieved, illustrated and computed upon. Still if and when the opportunity arises, this structure has to be reconfigured to account for the new opportunity at hand (Stiny 1991).

There are many examples that can do – a useful one is depicted in Figure 1a – a full account of the emergence of this specific paradigm and its role in shape grammar scholarship is given elsewhere (Stiny, 1991).

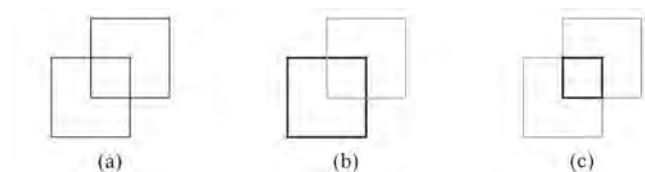


Figure 1: (a) A shape. (b) The same shape seen as two overlapping squares; (c) an emergent smaller third square.

The questions that arise looking at this shape quickly supersede the apparent simplicity of the argument. The careful observer – as well as the one familiar with the complexities, novelties and insights resulting from computations involving shapes, will immediately concede to the complexity that is hidden behind this – and many other, in fact all, pictorial settings. If the shape is seen as the intersection of two squares (Figure 1b) then the emergent smaller inner square is unaccounted (Figure 1c). And the Pandora's Box is opened: L-shapes, 8-shapes, +- shapes, parallel lines, quotation marks (<<) , and many more, all point to the wonderful unsettling nature of shapes and their resistance to be modeled in terms of discrete, atomic parts.

3. APPROACHES

First one must decide which graph representation to use for one's interpreter. Numerous have been proposed, we shall review some and then think about general and further possibilities. These thoughts do not include the problem of finding new intersections after a rule has been applied. This step will almost certainly not be implemented using a grammar, although it could make use of a suitable graph data structure. The graph or part of it must be recreated after each rule application. All graph representations below are based on the pictorial example of Figure 1.

3.1. Points to nodes and lines to edges

The most straightforward approach is to map points to nodes and lines to edges (Figure 2). Using a suitable embedding the graph may look much the same as the shape it represents. In architecture this kind of graph is perhaps best known as dual of the adjacency graph, it is sometimes called plan graph (March and Steadman 1971). The biggest drawback of this approach is its inability to model maximal lines (Stiny 1980). The maximal lines of a shape are the set of lines created by combining all collinear line segments that touch or overlap. An edge essentially represents a line segment. If a line is divided into several segments by crossing lines, it becomes difficult to query the graph for lines consisting of two or more segments within a rule. This results in difficulties finding emergent shapes.

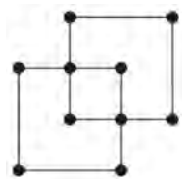


Figure 2: Mapping points to nodes and lines to edges: Plan graph

In order to overcome this problem Kelles et al. (2010) introduce what they term overcomplete graphs and simply join all node pairs of such a line (Figure 3). If there are n points along a line, $n * (n - 1) / 2$ edges are required. Thus a 10 x 10 grid of lines, not unreasonable in view of grammars such as the ice ray lattices (Stiny 1977) would require 900 edges. Certainly not impossible for modern computers, but still somewhat wasteful considering the complexity of the graph isomorphism problem and the fact that computing time will be proportional to the number of nodes and edges in the graph.

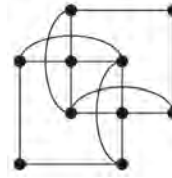


Figure 3: Mapping points to nodes and lines to edges: Overcomplete graph

An approach suited better for the representation of maximal lines would be one using hyperedges (Figure 4). One hyperedge can connect more than two nodes, making it easy to simply connect all points along a line by one edge. The drawback here is that few graph packages implement hyperedges. Still, graph grammars have dealt with them extensively (Drewes et al. 1997) and it would be an approach true to the maximal line concept.

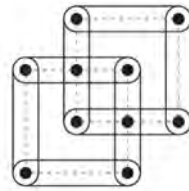


Figure 4: Mapping points to nodes and lines to edges. Hyperedge graph

3.2. Lines to Nodes and Intersections to Edges

Inverting the mapping of elements will result in further alternatives. One approach could be to represent each line segment as a node and have edges represent intersections (Figure 5). This possibility suffers from the same inadequacies concerning modeling maximal lines as the plan graph approach in Figure 2.

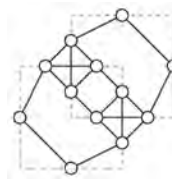


Figure 5: Mapping lines to nodes and intersections to edges: Plan edge graph.

In this case however representing individual line segments is not necessary. Mapping maximal lines to nodes instead (Figure 6) will result in a cleaner alternative with less edges and the ability to find emergent shapes. It would seem that this solution has all the advantages of the hyperedge approach in Figure 4, without the potential

drawback of having to use hyperedges. However, problems arise as soon as more than two lines intersect in one point, hence it can only be used on a very limited set of grammars, such as orthogonal grammars.

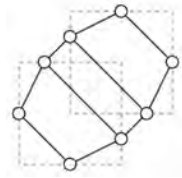


Figure 6: Mapping lines to nodes and intersections to edges: Hyperedge edge graph.

Again the only possibility to satisfy all needs is the introduction of hyperedges (Figure 7) with all the above mentioned reservations.

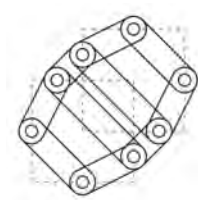


Figure 7: Mapping lines to nodes and intersections to edges: Overcomplete edge graph.

3.3. Points and Lines to Nodes

Finally both points and line segments could be represented as nodes (Figure 8). Again this approach lacks support for maximal lines, since line segments rather than lines are represented.

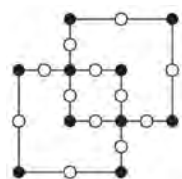


Figure 8: Mapping points and lines to nodes: Planar node-edge graph.

Using maximal lines instead of line segments (Figure 9) can easily improve upon the situation without having to use hyperedges. The necessity to introduce hyperedges can be traced back to the fact, that in a general shape both maximal lines and points can be related to more than two entities of the other class. That is to say there may be more than two points on a maximal line and more than two maximal lines may meet in a point. If both classes are represented by

nodes and edges are used for various kinds of relations, this restriction does not apply.

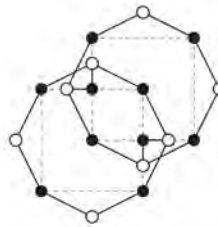


Figure 9: Mapping points and lines to nodes: Planar edge-edge graph.

Mapping several classes to nodes increases the total number of elements needed to represent a shape, but it also adds flexibility because additional classes could be added (Figure 10). Depending on the grammar at hand classes of interest could be labels, faces, solids, colors, infinite lines and more. Heisserman (1994) uses such a graph-based boundary representation for boundary solid grammars.

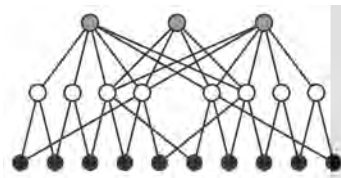


Figure 10: Graph representation of the shape in Figure 1a using three layers of objects: points (black), lines (white) and faces (grey).

3.4. Discussion

The approach chosen by the authors and used for the remainder of this paper is based on representing points and lines as nodes (Figure 9). It is a good compromise between few elements and an efficient and flexible data-structure.

4. CONSTRAINED TOPOLOGY PATTERNS

The graph as described above contains only topological information and as such would not be of much use for a shape grammar interpreter. Metric information can be added to the model by adding attributes to nodes and/or edges. It would suffice to add the coordinates of the points as attributes to the respective nodes as all other information such as lengths and angles could be derived. Depending on the nature of the implementation and the capabilities of the graph package it could make sense to store some additional information where appropriate. GRAPE stores the slopes of the maximal lines as attributes of the line nodes, in addition to the point coordinates.

The nature of graphs is that they do not care about their embedding. Therefore a search will return all isomorphisms. Thus a square will be returned in eight different variations (Figure 11); mostly this will be desirable, if not then additional constraints must be implemented.

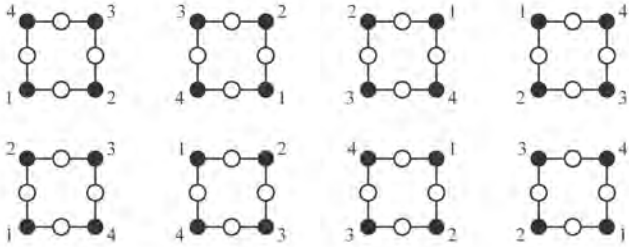


Figure 11: Isomorphisms of a quadrilateral topology.

In order to find a specific geometry it is then necessary to first search for the required topology and then further reduce the resulting set by adding constraints. This is much like one would proceed if describing a shape in a parametric modeling application. The advantage is that like one parametric model can represent numerous geometries; in this case one search pattern can find numerous shapes.

The unrestricted search pattern in Figure 12a will find all general quadrilaterals (Figure 12b) and hence must be restricted further to return only squares. Several sets of conditions can lead from a quadrilateral to a square. Which set is chosen can depend on the implementation details and the capabilities of the platform.

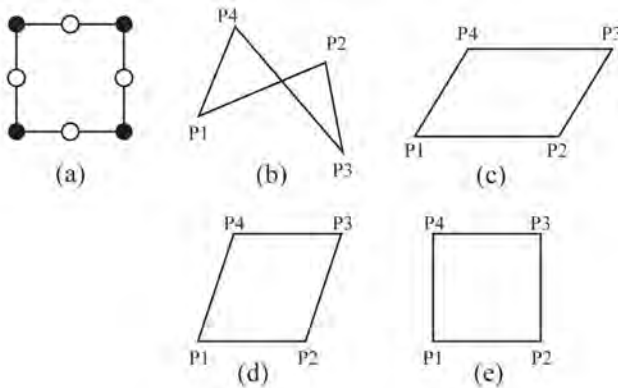


Figure 12: The topological search pattern (a) and a possible series of constrained shapes leading to a square (b-e).

In our example we will first request that opposing edges are parallel and that edge P_1P_2 is longer than or equal to edge P_2P_3 .

$$\begin{aligned} P_1P_2 &\parallel P_3P_4 \wedge \\ P_2P_3 &\parallel P_4P_1 \wedge \\ |\overrightarrow{P_1P_2}| &\geq |\overrightarrow{P_2P_3}| \end{aligned}$$

These conditions will return two isomorphisms for each parallelogram (Figure 12c). Further we can ask adjacent sides to be of equal length.

$$|\overrightarrow{P_1P_2}| = |\overrightarrow{P_2P_3}|$$

This will return four isomorphisms for each rhomb (Figure 12d). This is possible because for each rhomb two of the parallelograms found before will turn out to be the same. In symmetry terms a reflection was added to a symmetry group C_2 (parallelogram), resulting in a symmetry group D_4 (rhomb). Finally we must insist that the diagonals be of equal length.

$$|\overrightarrow{P_1P_3}| = |\overrightarrow{P_2P_4}|$$

This will return eight isomorphisms for each square (Figure 12e). Again this is made possible because two of the rhombs will turn out to be same and the symmetry group of the square is D_4 .

If not all eight isomorphisms are needed because, for example, the rule itself exhibits some symmetry and therefore cancels out some of the possibilities, additional constraints are required. Reflections can be avoided by defining a sense of rotation. A constraint on the cross product of two vectors could for example be used to return either left-turning or right-turning shapes.

$$(|\overrightarrow{P_1P_2}| \times |\overrightarrow{P_1P_4}|) \cdot z > 0$$

A single isomorphism is returned if a specific point is requested to be bottom-left most point of the topology.

$$\begin{aligned} P_1 \cdot x &\leq \{P_2, P_3, P_4\} \cdot x \wedge \\ P_1 \cdot y &\leq \{P_2, P_3, P_4\} \cdot y \end{aligned}$$

Some knowledge of symmetry subgroups, as discussed by Economou and Grasl (2009), is necessary while writing rules, otherwise some unnecessary matches could be returned.

5. SUBSHAPES

So far the graph data-structure used for the matching process was shown and the method of defining parametric

rules has been explained. However, once a rule is applied to the graph, the connection between the graph representation and the shape representation deteriorates. If new subshapes have emerged during the rule application, they will go undetected by the graph. .

Now the identification of any type of subshape within a shape is not a trivial task. There is a growing bibliography on the topic extensively dealt in the references already given. Here the key idea that structures the identification of the subshapes within a shape is the idea of maximal lines and its relation to boundary elements such as endpoints and intersection points (Stiny, 2006). There is an intuitive way to think of this idea; while for example, there are at the most three squares in the example of Figure 1, there is an indefinite number of cross-shapes (+) in the same shape. Two of them are shown in the Figure 13. It is clear that any number of cross-shapes can be embedded upon the shape including crosses of equal or unequal arm-lengths. However there are only two cross-shapes in this shape if we take into consideration the boundary conditions of the lines involved in this shape – that is, the endpoints and intersections of the two bigger squares. This work picks up this class of problems because it addresses in straightforward ways design concerns.

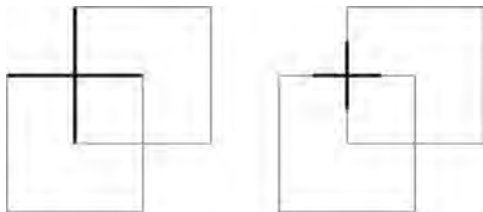


Figure 13: Two cross subshapes in the shape of Figure 1. A cross including segment boundaries and a smaller cross.

6. DERIVATION

The key idea for the implementation of a shape grammar is the notion of parallel description; one that accounts for what we see, a visual description, and another that accounts for the indexing, bookkeeping and actual implementation of what (in theory) we see, a symbolic description. Both representations are encoded in the design itself and the rules used to construct the design. The visual description consists of lines that merge and fuse at any time of the computation to create any shape conceivable. The symbolic description consists of a graph that captures at any time of the computation the relations of these lines one to another and to underlying carrier lines. Any execution of a shape rule

requires its translation in graph form, its computation within the graph structure of the design, and the retranslation of the graph in a pictorial setting.

The computation described above requires shapes that can be combined visually in any way possible, and symbols – atomic entities that can be meaningfully exchanged. The greatest problem with all shape grammar implementations is the capturing of this immediacy of visual computation.

The diagram of the shape grammar implementation in Figure 14 shows clearly these parallel representations. The shape is present in the world (the application) – but it has no apparent structure; once the designer decides to see it and act upon it in a specific way – say, pick up an embedded square and move it in across the direction of the diagonal of the square at a specific distance offsetting, - then the underlying structure (parts and relations) of the shape – in itself captured nicely by the visual structure of the graph – is queried for squares. The results of the symbolic computation are output in the application in a linear fashion and the user can identify all possible patches of the rules in a sequential feedback from the machine. Once the user executes the rule the shape transforms into its new state and the underlying graph is redescribed in terms of what is visually present. The computation in the model is entirely symbolic; graphs capture both the structure of the shape and the structure of the rules involved in the computation. The process is straightforward: once a rule (the graph rule) is applied to the shape (graph representation of some shape), the result of the computation (a graph) is mapped back into a shape. In other words, after a rule has been applied to the graph representation the result is mapped into a shape. Then the new shape, including any new subshapes, is transformed back into a graph. This loop ensures that characteristics of the shape, not easily detected using the graph alone, are captured.

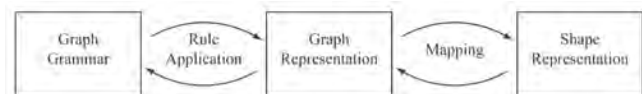


Figure 14: Parallel representation of the derivation.

7. GRAPE

After installing the GRAPE plug-in the ‘grape’ command (Figure 15) is added to Rhino, the host application. Once this command is executed the user can either select existing lines as initial shape or use one of

several pre-defined shapes. The chosen shape is mapped to a graph representation.

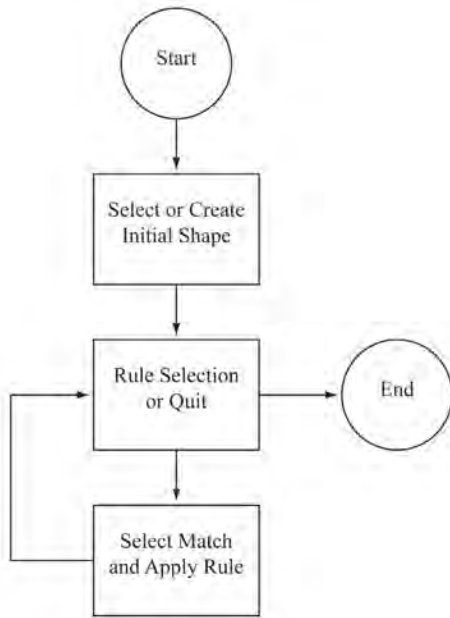


Figure 15: Diagram of the commands structure

The user can then select a rule from the catalog or exit the command. If a rule is selected the matching isomorphisms are returned and the user can cycle through the respective shapes until the desired match is shown. Finally the rule can be applied and the command returns to the catalog of rules for the selection of the next rule.

An example of a shape computation in Grape is given in Figure 16. Here the rules used for the computation are taken from classic examples in the field of shape grammars to relate to the existing discourse. Obviously any other shapes and rules would do. The three rules used in the computation are given at the top of Figure 16; the first moves an L-shape along its diagonal axis; the second rule moves a square along its diagonal axis and the third rule moves a copy of a square along its diagonal axis. The cross label to the side of each rule is used to fix the origin of transformation. The dotted lines visualize the effect of the application of the rule. The computation itself is shown in the lower half of Figure 16. The initial shape of Figure 1 (the double square) is used as an initial shape.

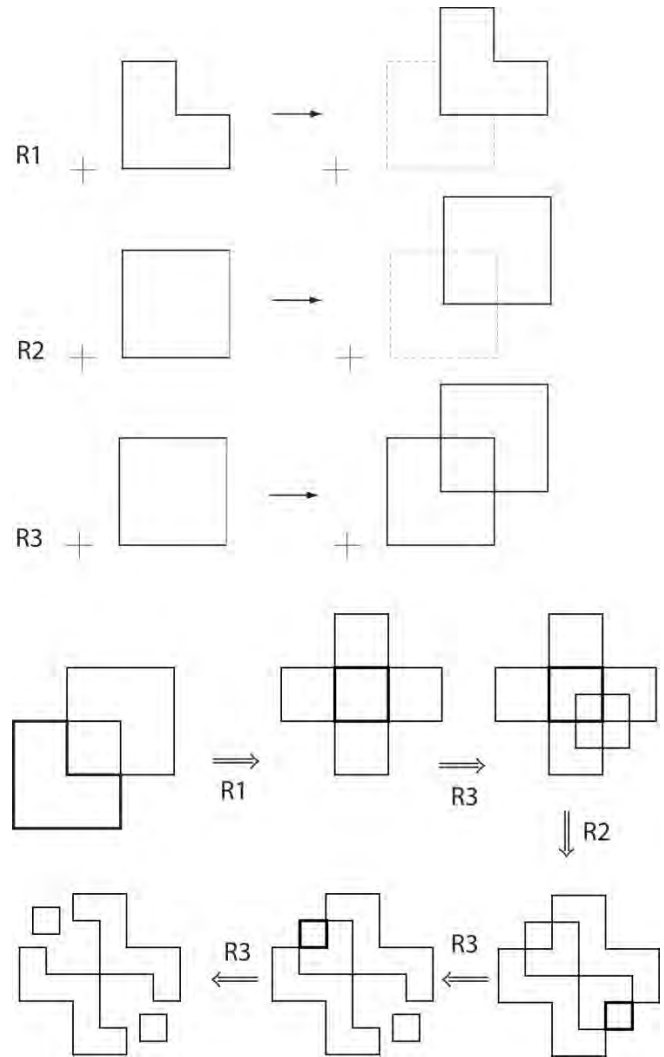


Figure 16: Example of a derivation using the initial shape of Figure 1 and three shape rules.

8. CONCLUSION

Grape simulates nicely some of the visual calculations that designers do effortlessly with a pencil and a paper – visual calculations that shape grammars capture nicely in their algebras of shapes and rules in schemas defined in these algebras. These visual calculations here have remained sufficiently abstract to foreground the two most important characteristics of shape grammars; recursion and embedding; recursion as the means to construct a shape in terms of the previous one in the calculation and embedding as the means to construct a shape in terms of an appropriate matching and application of a rule within the shape. The key mechanism that was used to present these two ideas was the notion of maximal lines and the description of any shape

and their subshapes in terms of such maximal lines. There are many more devices of the formidable armature of shape grammars - and visual computation at large, that would be useful to be automated, and this is one step towards this direction. Several directions present themselves readily.

Most shape grammars applications have been modeled after Stiny's two studies in formal composition (1976) and their foundational inquiry upon the significance and usage of the notion of rule. The rules in the first type of studies are defined in terms of parts and relations observed or found in some existing corpus while in the second are constructed from scratch and they do not have a specific existing corpus as a target for definition (or generative reconstruction). Grape has been designed to address these later exercises; in a sense it allows for a free definition of rules and a formal experimentation with these rules. A specific application designed to address the highly complex grammars that address specific architectural languages using graph grammars is given elsewhere (Grasl and Economou, 2010). And still more work needs to be done to provide a seamless platform between such efforts for computation.

Currently GRAPE handles grammars in U_{12} . Since the method described above can also handle additional dimensions and object types expanding the software to other classes of algebras is possible including the algebras of shapes, labels and weights. The task is not trivial and a major effort is required for its successful implementation.

Currently GRAPE handles rules entirely symbolically. Visual editing capabilities are absent and all rules are defined symbolically. It is trivial to allow the creation and alteration of rules directly via the graph grammar, however this does not greatly increase usability and really only poses an option for interested CAAD professionals. A graphical editor, much along the lines described by Tapia (1999), would of course be more appealing. This also constitutes a major effort and is being considered for future research.

Acknowledgements

GRAPE (G**R**aph **s**HAPE) is a plug-in written for C# for Rhino 5. The graph grammars were implemented using GrGen.NET.

References

- CHAU H. H., CHEN X., MCKAY A. AND DE PENNINGTON A. 2004. Evaluation of a 3D shape grammar implementation, In J.S. Gero (Ed.) Design computing and cognition '04. Kluwer. Boston. pp 357-376.
- DREWES F., KREOWSKI H.-J., AND HABLE A. 1997. Hyperedge Replacement Graph Grammars. In G. Rozenberg (Ed.) Handbook of Graph Grammars, Volume 1.
- ECONOMOU A. AND GRASL T. 2009. Point Worlds. Computation: The New Realm of Architectural Design [27th eCAADe Conference Proceedings]. 221 – 228.
- GIPS J. 1999. Computer Implementation of Shape Grammars. NSF/MIT Workshop on Shape Computation.
- GRASL T. AND ECONOMOU A. 2010. Palladian Graphs: Using a graph grammar to automate the Palladian grammar. F [28th eCAADe Conference Proceedings]. 275 – 283.
- HEISSERMAN J. 1994. Generative Geometric Design. IEEE Computer Graphics and Applications. **14**(2). 37 – 45.
- JOWERS I. AND EARL C. 2010. The construction of curved shapes. Environment and Planning B. **37**(1). 42 – 58.
- KELES H. Y., ÖZKAR M., AND TARI S. 2010. Embedding shapes without predefined parts. Environment and Planning B: Planning and Design. **37**(4). 664 – 681.
- KNIGHT T. 2003. Computing with emergence. Environment and Planning B. **30**(1). 125 – 155.
- KRISHNAMURTI R. 1981. The construction of shapes. Environment and Planning B. **8**. 5-40.
- SCHÖN D. 1983. The Reflective Practitioner. Temple Smith.
- STINY G. 1977. Ice-ray: a note on the generation of Chinese lattice designs. Environment and Planning B. **4**(1). 89 – 98.
- STINY G. 1980. Introduction to shape and shape grammars. Environment and Planning B. **7**(3). 343 – 351.
- STINY G. 1982. Spatial relations and grammars. Environment and Planning B. **9**(1). 113 – 114.
- STINY G. 1991. The algebras of design. Research in Engineering Design. **2**(3). 171-181.
- STINY G. 1992. Weights. Environment and Planning B. **19**. 413-430.
- STINY G. 2006. Shape: Talking about Seeing and Doing. MIT Press.
- STINY G. AND GIPS J. 1972. Shape Grammars and the Generative Specification of Painting and Sculpture. In C. V. Freiman (Ed.) Information Processing 71.
- TAPIA M. 1999. A visual implementation of a shape grammar system. Environment and Planning B. **26**(1). 59 – 73.

Automated Energy Model Creation for Conceptual Design

Lillian Smith, Kyle Bernhardt, Matthew Jezyk

Autodesk, Inc.
1560 Trapelo Rd,
Waltham, MA 02451

lillian.smith@autodesk.com
kyle.bernhardt@autodesk.com
matthew.jczyk@autodesk.com

Keywords: design process, building performance simulation, whole building energy analysis

Abstract

Architects today rarely use whole building energy analysis to inform their early design process. During this stage of design, the team is responsible for macro-level decisions (such as basic form and orientation) that can have the most significant effects on energy use. So why are so few architects using energy analysis tools at this stage? In many cases, this is because the tools available to perform energy analysis are too complex and time-consuming to learn and use during the fast-paced conceptual design phase because of the time needed to model and analyze the design. Often energy analysis tools are separate, standalone applications that require a separate thermal model to be authored. This adds more time and complexity and limits the number of professionals experienced with these tools. Understanding the results and using them in presentations is also quite difficult in current tools.

This paper describes a system that can alleviate these problems by automatically generating an energy model from an architect's basic massing model. Our approach allows an architect to focus on modeling the building form, rather than the thermal zones that are the focal point for most energy modeling software on the market today. The architect's massing model can then be modified and the energy model stays in sync, allowing comparative analyses to be made quickly and easily.

1. INTRODUCTION

Our background research took place during late 2008 and early 2009. We visited and met with teams from over 25 architecture and engineering offices, and supplemented this

primary research by working with several universities and environmental organizations. Research was focused primarily in the United States. The focus of our research was energy analysis during conceptual design. Through our research, we concluded that architects were failing to perform energy analysis during conceptual phases of design for a number of reasons. First, they thought the process was too difficult. When they tried to export models to various analysis programs, their exports often failed because their models were too complex or not built correctly from an energy perspective. Secondly, they were not interested in learning energy analysis programs or building a separate model for energy analysis. Finally, they were also stymied by having to make too many decisions early in their design process before they had determined basics like detailed wall compositions.

Achieving LEED® (Leadership in Energy and Environmental Design) credits has become a major driver for performing energy analysis as more building projects drive towards LEED certification through the USGBC. Architects said they had to spend a great deal of money to hire LEED® consultants to help with this analysis. While they saw the value of trying to analyze energy use early, they were performing this analysis late in the design process when required for LEED® certification. At this point in the project, however, it was too late to make macro-level changes to the building's form. As a result, the energy analysis often served to only meet a LEED® credit requirement and was not as impactful as it could have been.

Architects requested tools that were integral to their modeling applications, and could generate energy models quickly and compare different design alternatives. At this early stage they were more interested in relative

comparisons than in actual numbers for specific energy analyses. Finally, they were not looking for a tool to replace their engineering consultants; rather they wanted tools that would allow them to have better conversations about how to save energy, empowering the project team to make informed design decisions.

Our research was validated by findings from several other studies. The 2008 Autodesk/AIA Green Index reports that architects believe that using the design process to reduce building energy consumption includes an increased reliance on design software. Leveraging existing software and making it easy to learn and use is important because nearly three-fourths of architects (72%) are concerned that clients are not willing to pay the added first costs of green designs (Autodesk/AIA 2008). McGraw/Hill reports that while fewer than 20% of firms are simulating energy performance, in two years 80% see it as very important and want to simulate whole building energy use. (McGraw/Hill 2010). The same study reports that nearly all experts expressed the need for better software integration with energy performance modeling software used by engineering firms (McGraw/Hill 2010). Reasons cited for not using BIM for energy performance simulations were lack of tools, lack of functionality, and tools or models that are too complicated. (McGraw/Hill 2010). Glicksman and Urban point out the “daunting amount of information required [in eQUEST] for even the simplest designs” (Glicksman and Urban 2007).

2. PROJECT GOALS

Learning from our research, we defined the following project goals. First, we wanted the architect to be able to quickly model an idea the way that s/he understands the building and let the software rationalize the architect’s model into an energy model. To accomplish this goal, we needed to convert the architect’s building form into the thermal zones required for an energy model. Additionally, we needed to make reasonable assumptions for the operational characteristics of the building based upon minimal input from architects. The architect would need to be able to modify and customize the energy model, but would not be required to change any of the initial assumptions before conducting an analysis.

The next goal was for the automated energy model to be constantly up to date with the building form. The architect could make a change to the building form and the energy

model would update accordingly, always staying in sync. Introducing energy analysis into a parametric modeling program had the added benefit of empowering users to create more flexible and complex geometry than is easily supported in energy analysis programs. As a result, it was necessary to ensure that complex forms can be accurately rationalized into valid energy models.

3. THE AUTOMATIC ENERGY MODEL

Our software is integrated with Autodesk® Revit™ as a common BIM platform that is familiar to many architects. Revit already presents a set of workflow for preparing and exporting a detailed model to whole building energy analysis software like Autodesk® Green Building Studio™, but the inherent complexity found in a typical building information model poses certain technologic and usability challenges that limit its utility in early stage design. There were too many opportunities to fail with an energy model that was derived from a detailed building model. To make the process more reliable and focus on the earlier stages of the design process, we decided to leverage the existing conceptual design environment that also existed in Revit. In this environment, users model conceptual building forms without having to detail specific building elements. Basic building masses can be quickly created, modified, and parameterized. Tools complimentary to energy analysis, including a sun path tool for studying sun position over time, shadows, and solar radiation analysis, also made this a rich environment in which to work.

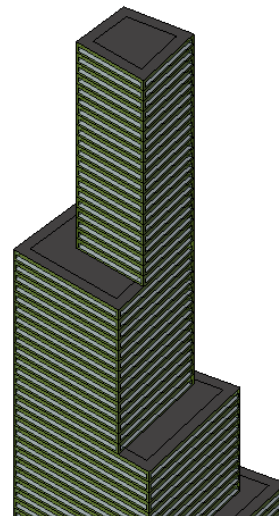


Figure 1. Basic form with floors and energy model

3.1. Designating Energy Model Geometry

To pursue our goal of automatically creating an energy model from an architect's massing model, we first had to determine which modeled geometry was actually intended to be part of an energy model. Energy analysis requires floor area, and existing Revit functionality allows us to "slice" geometric forms with reference floor levels. Therefore, we decided to use this floor slicing functionality as the first requirement for which geometry would be translated into an energy model. Geometry that was designated to have floor area was geometry that would be analyzed as a building (Figure 1). Other geometry without floor slicing, such as neighboring buildings, would simply act as shading devices (Figure 2).

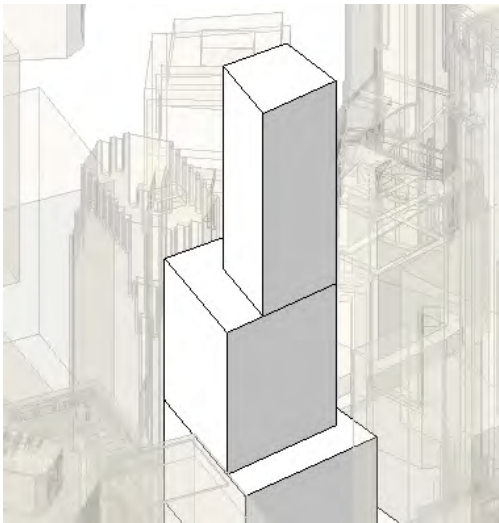


Figure 2. Basic building form without floors

3.2. Automatic Zoning

Thermal zones are the primitive 3D geometry elements of an energy model, with surfaces defining their geometry. With our focus on the architect's massing forms as the primary geometry, we needed to develop an automated process that generated thermal zones from those massing forms. The floor slicing functionality in Revit is a valuable tool for architects to calculate floor areas and volumes from complex shapes, but it also is useful for transforming the model into one that can be used for energy analysis. Zones generally are divided at least by floors of a building, so we created a rule whereby a zone equals a form plus a floor.

This rule produces a single thermal zone for each level of a massing form, spanning up to the next level (Figure 3).

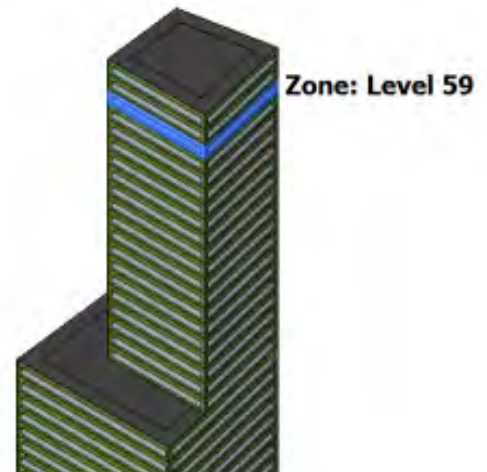


Figure 3. Automatic zoning by level

To further refine the zoning model and provide a more realistic energy model, we provided two additional automatic controls to our thermal zone creation. The first is an option to automatically generate a core and perimeter zone, a common thermal zoning approach. The second control divides the per-floor zoning of the building (excluding the core) into 4 thermal zones: northeast, southeast, northwest, and southwest for more accurate energy consumption estimates (Figure 4).

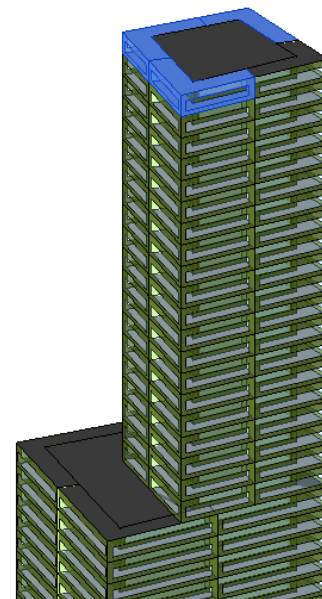


Figure 4. Energy model with core offset and perimeter zones

3.3. Automatic Glazing and Shading

In addition to automatic zones, we also needed to provide a fast and automatic way of modeling glazing and shading devices. We used a common metric of window-to-wall percentage as the primary control method. Providing default values and parameters for “Target Percentage Glazing” and “Target Sill Height”, we could automatically model windows on each exterior wall surface. Additionally we added “Target Percentage Skylights” with a “Skylight Width & Depth” parameter. Skylight defaults were set to 0% by default, but changing this value would automatically create skylights on any surface that was designated as a roof. Finally, a “Glazing is Shaded” parameter combined with a “Shade Depth” parameter would specify whether automatic shading geometry would be created over each instance of glazing (Figure 5).

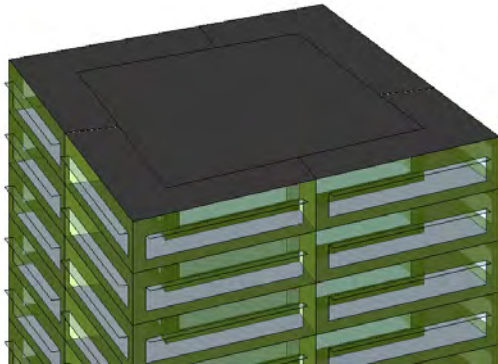


Figure 5. Automatic glazing and shading

3.4. Automatic surface determination

Next we needed an automated way of defining the surface types for the surfaces that make up thermal zones. In other energy modeling applications, this is typically a manual process, but we deemed automatic assignment critical for rapidly developing an energy model. To do this, we developed mapping logic (Table 1) that automatically translates geometric forms into the surface types required for energy modeling. Modifications to automatic determination logic are allowed, but only in certain cases so that a reliable energy model was guaranteed. For simplicity, we limited the number of surface types in Revit to only what was absolutely necessary. The code used a set of rules to map surface types to gbXML for export (Table 2).

Revit Surface Type	Automatic Determination Logic	Permitted Modifications
InteriorWall	Non-horizontal face with adjacent zones on both sides.	Changeable to opening only
ExteriorWall	Face eligible for “Wall by Face” command (existing Revit functionality) and adjacent to zones on one side only or face region of exterior wall with Underground = ON. If face is below ground plane (as set in Energy Settings) “Underground” parameter is set to ON and is not user modifiable.	If form face, then changeable to roof only. If face region on a parent exterior wall, then changeable to glazing.
Roof	Face eligible for “Roof by Face” command and adjacent to zones on one side only or face region of a roof with Underground = ON. If face is below ground plane (as set in Energy Data) “Underground” parameter is set to ON and is not user modifiable.	If form face, then changeable to Exterior Wall only. If face region of a parent roof, changeable to skylight only.
Floor	Floors (existing functionality) system-defined “Slab” property added to floor element which defines if floor is a slab. Checkbox is ON if floor has zone adjacencies on only one side (not user)	Unchangeable
Shade	Face with no zones adjacencies.	Unchangeable
Glazing	Face region of an exterior wall with Underground = OFF or interior wall OR automatically generated face regions of an exterior wall (auto-glazing)	If contained on an exterior wall, changeable to exterior wall. If contained on a interior wall, changeable to interior wall
Skylight	Face region of a roof with Underground = OFF OR automatically generated face regions of a roof (auto-skylights)	Changeable to Roof
Opening	Not automatically defined.	Changeable to glazing or interior wall
Zone	Created when floors are assigned via reference levels	Unchangeable

Table 1: Automatic surface determination

gbXML TypeEnum	Revit Surface Type	Additional Determination Logic
InteriorWall	Interior Wall	n/a
ExteriorWall	Exterior Wall	Underground OFF
Roof	Roof	Underground OFF
InteriorFloor	Floor	Slab = OFF
Shade	Shade	n/a
UndergroundWall	Exterior Wall	Underground ON
UndergroundSlab	Floor	Slab ON AND Level < Ground Plane
Ceiling	n/a	n/a
Air	Opening	n/a
UndergroundCeiling	Roof	Underground ON
RaisedFloor	Floor	Slab ON AND Level > Ground Plane
SlabOnGrade	Floor	Slab ON AND Level = Ground Plane
FreestandingColumn	n/a	n/a
EmbeddedColumn	n/a	n/a
FixedWindow	Glazing	n/a
OperableWindow	n/a	n/a
FixedSkylight	Skylight	n/a
OperableSkylight	n/a	n/a
SlidingDoor	n/a	n/a
NonSlidingDoor	n/a	n/a
Air	Opening	Face Region of a Exterior Wall or Interior Wall

Table 2: Surface types to gbXML mapping

3.5. Conceptual constructions

After automatically creating surfaces for our model, we need to define their thermal performance characteristics. The conceptual stage of design occurs before decisions are made on exact materials, so to guarantee reliable analysis we limited the choice of materials to a specific list customized for each surface type. Defaults, seen in green rows, can be set and saved in template files for different climatic regions (Table 3).

Revit Surface Type	Conceptual Construction	R-Value (ft ² ·hr ·F/Btu)	Unit Density (lbm/ft ²)	Heat Capacity (Btu/ft ² ·F)	Default t?
Exterior Wall	Lightweight Construction – high insulation	25	23	5	
Exterior Wall	Lightweight Construction – typical cold climate insulation	17	21	4	
Exterior Wall	Lightweight Construction – typical mild climate insulation	10	18	4	Yes
Exterior Wall	Lightweight Construction – low insulation	8	19	4	
Exterior Wall	Lightweight Construction – No insulation/Interior	3	18	4	
Exterior Wall	High Mass Construction – high insulation	17	115	23	
Exterior Wall	High Mass Construction – typical cold climate insulation	15	110	22	
Exterior Wall	High Mass Construction – typical mild climate insulation	11	110	22	
Exterior Wall	High Mass Construction – No insulation/Interior	1	110	22	
Exterior Wall--Underground	High Mass Construction – high insulation				
Exterior Wall--Underground	High Mass Construction – typical cold climate insulation				
Exterior Wall--Underground	High Mass Construction – typical mild climate insulation				Yes
Exterior Wall--Underground	High Mass Construction – No insulation/Interior				
Interior Wall	Lightweight Construction – No insulation/Interior	2	5	1	
Interior Wall	High Mass Construction – No insulation/Interior	1	110	22	Yes
Roof	High Insulation, Cool Roof	32	14	3	
Roof	High Insulation, Dark Roof	32	14	3	
Roof	Typical Insulation, Cool Roof	22	10	3	Yes
Roof	Typical Insulation, Dark Roof	22	10	3	
Roof	Low Insulation, Cool Roof	12	8	2	
Roof	Low Insulation, Dark Roof	12	8	2	
Roof	No Insulation, Dark Roof	2	7	2	
Floor	Lightweight Construction – high insulation	29	4	1	
Floor	Lightweight Construction – typical insulation	21	4	1	Yes
Floor	Lightweight Construction – low insulation	14	4	1	
Floor	Lightweight Construction – no insulation/Interior	4	2	1	
Exterior Floor	Lightweight Construction – no insulation/Interior	4	2	1	Yes
Slab on Grade	High Mass Construction – frigid climate slab insulation	16	123	25	
Slab on Grade	High Mass Construction – cold climate slab insulation	11	123	25	
Slab on Grade	High Mass Construction – typical no insulation	6	123	25	Yes
Interior Slab	Hi Mass Construction - Interior Slab	1	46	9	Yes

Conceptual Model	Conceptual Construction	U-Value Btu/(ft ² ·oF ·h)	Solar Heat Gain Coeffici ent	Visible Transmitta nce (Tvis)	Default t
Glazing	Single Pane Clear – no coating	1.09	0.81	0.88	
Glazing	Single Pane – Tinted	1.11	0.71	0.61	
Glazing	Single Pane – Reflective	0.89	0.28	0.13	
Glazing	Double Pane Clear – No coating	0.56	0.69	0.78	Yes
Glazing	Double Pane Tinted	0.57	0.61	0.55	
Glazing	Double Pane Reflective	0.42	0.19	0.10	
Glazing	Double Pane Clear – LowE Cold Climate, High SHGC	0.35	0.67	0.72	
Glazing	Double Pane Clear – LowE Hot Climate, Low SHGC	0.30	0.44	0.70	
Glazing	High Performance Double Pane Clear, LowE, High Tvis, Low SH	0.29	0.27	0.64	
Glazing	Triple Pane Clear, LowE Hot or Cold Climate	0.22	0.47	0.64	
Glazing	Quad Pane Clear, LowE Hot or Cold Climate	0.12	0.45	0.62	
Skylights	Single Pane – Tinted	1.11	0.71	0.61	
Skylights	Single Pane – Reflective	0.89	0.28	0.13	
Skylights	Double Pane Clear – No coating	0.56	0.69	0.78	Yes
Skylights	Double Pane Tinted	0.57	0.61	0.55	
Skylights	Double Pane Reflective	0.42	0.19	0.10	
Skylights	Double Pane Clear – LowE Cold Climate, High SHGC	0.35	0.67	0.72	
Skylights	Double Pane Clear – LowE Hot Climate, Low SHGC	0.30	0.44	0.70	
Skylights	High Performance Double Pane Clear, LowE, High Tvis, Low SH	0.29	0.27	0.64	
Skylights	Triple Pane Clear, LowE Hot or Cold Climate	0.22	0.47	0.64	
Skylights	Quad Pane Clear, LowE Hot or Cold Climate	0.12	0.45	0.62	

Table 3: Conceptual constructions

3.6. Building Operation Characteristics

The definition of the energy model geometry and thermal characteristics are only part of the definition of a complete energy model. We also needed to define how the building would be occupied, and what methods it would use to meet the occupancy energy needs. Again we took the approach of trying to set as many intelligent defaults as possible to reduce the number of steps required for a successful analysis. In our application, users are not forced to make many upfront decisions to conduct preliminary analysis. We pre-define a series of energy settings parameters and allow multiple values to be set correctly based on a single parameter edit from the user. For example, setting the 'Building Type' parameter to the value of 'Office' sets the planned usage schedule of the building and defines the internal energy gains automatically for the users. A retail store is assumed to be open more hours per year than an office building. For mixed use projects, the building operating schedule can be overridden and space assignments can be made for specific zones. However, by defining default settings based on building type, users were able to get to comparative analysis faster.

Assumptions and default settings are critical to useful analysis and are based on leading industry standards and research. Schedules (e.g. Occupancy) are based on the California Non-residential New Construction Baseline Study 1999. Envelope thermal characteristics, Lighting Power Density, and HVAC efficiency are based on ASHRAE 90.1 2007 and ASHRAE 90.2 2007. Equipment power density & Domestic Hot Water loads are derived from the California 2005 Title 24 Energy Code. Occupancy density and ventilation values come from ASHRAE 62.1-2007. The HVAC defaults for building type, size, and other miscellaneous building characteristics are based on the 2003 Commercial Buildings Energy Consumption Survey (CBECS).

3.7. Overriding automatic assignments

Creating an automatic energy model allows users to get from their concepts of a form to analysis results quickly. However, many users' first concerns are how they can customize the automatic energy model that the computer has created. There are two levels of control over energy model inputs, either at the global project level or at the instance level. Users can change the percentage glazing of every exterior wall in the project using a project-level control, or

select specific faces, like just the south face of the model to override the percentage glazing for just that face (Figure 6).

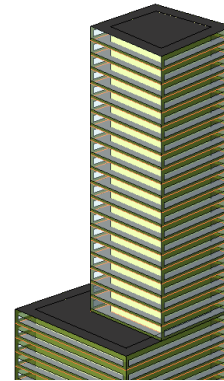


Figure 6. Glazing overridden on one face

Similarly, a zone can use a space type that is based on the entire building type (such as office building) or defined to meet specific needs (such as office - open plan). Automatic glazing can be turned off on a face and users can sketch a custom glazing configuration on that face. Automatic zoning of core and perimeter divisions can also be turned off and, using the rule that a form plus a floor slice equals a zone, one can completely customize the zoning layout by manually modeling the intended geometry.

3.8. Updates to model geometry

The energy model that is created from a mass form is always in sync with that form. Because the energy model is simply another way of viewing the form, any changes to the form are immediately translated to the energy model. The user is not required to manually sync up two disparate models in two separate applications.

4. CONCLUSIONS AND FUTURE RESEARCH

Usability testing with architects and students reveals that most users of our tools can obtain energy results from analyzing a simple model that they have built in less than 2 hours on first use of the product. This is a great improvement over the several days required in the past to get similar energy analysis results. Additionally, we are seeing encouraging results from the analysis servers. The Autodesk Green Building Studio service, which analyzes these energy models, has seen a market increase in the number of runs that it processes on a monthly basis. After the release of our automatic conceptual energy modeling analysis system in October 2010, the total number of analysis runs is up on average 600% over detailed Revit

energy model runs (Figure 7). Though it has only been a few months, the large increase in the number of successful analysis runs suggests that the automatic, conceptual energy model has increased usage and adoption in our user community. Note this data only illustrates an increase in usage. We do not yet have sufficient data to show a correlating improvement in the applicability and/or understandability of the results. Further research and surveys are required to understand how users are interpreting the results and using them in practice.

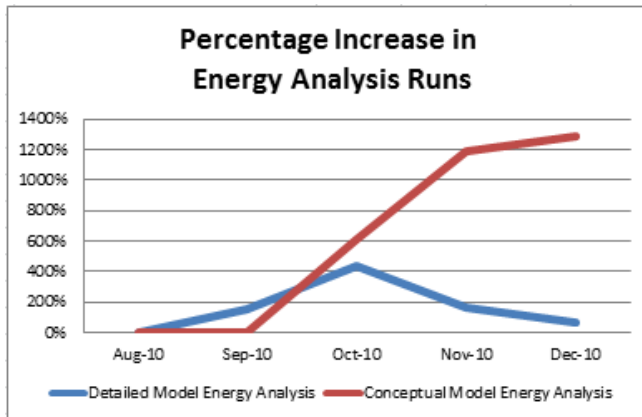


Figure 7. Percentage Increase in Conceptual Energy Analysis Runs

To validate our tools, we hired an independent consultant to create multiple test models to analyze energy use against regional and national commercial building benchmark data. In addition, DOE-2 inputs generated by our software were reviewed by a team of energy analysts to certify the data in the model was within acceptable standards of practice for DOE-2 energy models. Typically, accuracy of plus or minus 25% is considered acceptable for initial studies and the results from a majority of building types fell within this range.

While our consultants found a satisfactory general picture of the accuracy found in simple conceptual models, the ability of the software to correctly reflect the relative effect of design changes to overall energy use is far more important. In other words, what is the change in energy use as a result of changes in the overall design? How can we make the results from building geometry or shell construction modifications easy to access, especially as conceptual design decisions are made? Design iterations were run and compared to see the effects of changes to building parameters using an 18,500 square foot library as a test model. The eight iterations that are shown in Table 4

below demonstrate that changes in design result in the expected corresponding changes in energy use. This ability to quickly assess the effects of design changes to building energy use illustrates the power of this tool. For example, the energy savings of various glazing areas, window types, and wall u-factors shown below can be used to drive the initial design concept and avoid a potential (and more costly) design change that may be desired later to achieve energy performance goals.

Run	Roof U-Factor	Wall U-Factor	HVAC System	Glazing Area %	Glazing Type	Glazing U-Factor	Glazing SHGC	Estimated Savings
Baseline	0.06	0.08	Code	30%	Double	0.56	0.69	
1	0.06	0.08	Code	40%	Double	0.56	0.69	-5.7%
2	0.03	0.06	Code	40%	Double	0.56	0.69	-2.7%
3	0.03	0.06	Code	30%	Double	0.56	0.69	3.5%
4	0.06	0.08	Code	40%	Triple	0.22	0.47	5.6%
5	0.03	0.06	Code	40%	Triple	0.22	0.47	8.6%
6	0.06	0.08	Code	30%	Triple	0.22	0.47	8.7%
7	0.03	0.06	Code	30%	Triple	0.22	0.47	12.3%

Table 4: Library Envelope Changes and Effects

While studies by our independent consultants showed that our automatic energy model generation does provide reasonable results in most cases, the studies coupled with our own review also pointed out areas for improvement in three primary areas. Lack of control over internal load density controls can cause these load values to be too high. While we could try to tweak these values to more closely match today's energy efficiency standards for specific building types, allowing user control over internal load density would provide the ability to more accurately model the desired inputs. Energy Cost Data is another area that could benefit from more user control. While using an automatic value is often adequate for project in the United States, users on the global market need better controls here. Third, our mechanism for automatic zoning needs improvement. The algorithm that we use for the automatic division of the perimeter zone can cause different exposures to be grouped in the same zone (Figure 4). While the argument can be made that breaking the perimeter zones in this way is better than not doing it at all, many energy analysts feel that this approach is wrong and we need to improve this function.

Creating an automatic energy model is beneficial for architects and designers in analyzing forms that they design. The next step is to go farther, asking questions about what is the most energy efficient form given a set of constraints. What is the best orientation for this kind of building at this

location? What is the most advantageous form for this amount of square footage of this building type? The automatic energy model could be leveraged to automate the analysis of many design choices and quickly illustrate the advantages and disadvantages of different choices before making decisions.

Acknowledgements

We would like to thank all of the participants in our user research and testing as well as our subject matter experts, developers, designers, and testers for making this product a reality.

References

U.S. GREEN BUILDINGS COUNCIL. 2009. LEED-NC Green Building Rating System for New Construction & Major Renovations v.3.0. <http://www.usgbc.org/DisplayPage.aspx?CMSPageID=1970>.

AUTODESK AND AIA, 2008. The 2008 Autodesk/AIA Green Index, 4

MCGRAW HILL CONSTRUCTION 2010 Green BIM How Building Information Modeling is Contributing to Green Design and Construction. 6,26.

GLICKSMAN, LEON R. PHD, URBAN, BRYAN J. 2007 A Simplified Rapid Energy Model and Interface for Nontechnical Users. 3.

AUTODESK, INC. Autodesk Revit 2011 Software
<http://www.autodesk.com/revit>

AUTODESK, INC. Autodesk Green Building Studio Software. <https://gbs.autodesk.com/>

RLW ANALYTICS, INC. 1999. California State-Level Market Assessment and Evaluation Study. http://www.energycodes.gov/publications/research/documents/baseline/California_rpt.pdf

ASHRAE. 2007. ASHRAE 90.1-2007. Energy Standard for Buildings Except Low-Rise Residential Buildings. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

ASHRAE. 2007. ASHRAE 90.2-2007. Energy Efficient Design of Low-Rise Residential Buildings. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

THE CALIFORNIA ENERGY COMMISSION. 2005. California 2005 Title 24 Energy Code.

ASHRAE 2007. ASHRAE 62.1-2007. Ventilation for Acceptable Indoor Air Quality. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

US ENERGY INFORMATION ADMINISTRATION. 2003. Commercial Buildings Energy Consumption Survey (CBECS).

DOE-2. 2006. Lawrence Berkeley National Laboratory. University of California, Berkeley.

DOE-2. 2007. eQUEST the Quick Energy Simulation Tool. <http://doe2.com/equest>.

An Integrated Approach to Algorithmic Design and Environmental Analysis

Robert Aish¹, Andrew Marsh²

Autodesk Platform Solution¹

robert.aish@autodesk.com

Autodesk AEC-Strategic Technology²

andrew.marsh@autodesk.com

Keywords: Design Computation, Geometric Modelling, End-User programming, Environmental Analysis

Abstract

This paper describes the motivation and design of DesignScript, an end-user domain-specific programming language for algorithmic architectural and geometric design. Furthermore, the integration and use of DesignScript within the context of an environmental analysis software application is described that invokes a representative subset of analysis functionality.

1. INTRODUCTION

There are many examples of environmental analysis tools being developed as standalone vertical applications. Often these analysis tools have their own modelling capabilities or could import models built in dedicated modelling applications. More recently, environmental analysis tools have been integrated with external geometric modelling applications. Typically these modelling applications harness more sophisticated geometric functionality. This modelling functionality is usually driven by user interaction using direct manipulation modelling techniques. The objective here is first, to enable the user to represent his design using the appropriate geometric functionality and at an appropriate level of detail and second, be able to use this representation directly with the environmental analysis tools (to support the ‘design-analysis-redesign’ loop).

This paper explores one such integration of an algorithmic model technique, which we call DesignScript, integrated within a commercial environmental analysis tool, called Ecotect (Autodesk 2011).

In this paper we will:

- Describe the rationale for DesignScript.
- Describe Ecotect.
- Describe the integration of DesignScript and Ecotect and an example of the use of this integration.
- Offer some conclusions about what has been achieved and ideas for future directions.

An important underlying goal of this work is to create tools which give less experienced users access to more sophisticated functionality without providing that functionality in ways which may obfuscate the underlying concepts. We may need to ask the following questions:

- What pre-requisite concepts should the user have already acquired before using the application (which the application designer can assume to be understood by the user)?
- What concepts should the user be able to acquire during the use of the application, but which do not have to be understood as a pre-condition for the application’s initial use?

In the context of an integrated approach to algorithmic design and environmental analysis this should include both computation and environmental concepts. The application is not just providing design functionality, but has the potential to act as a learning environment. The result of using the application is not just an improved design, but new concepts and knowledge that have been acquired by the user in the course of creating that design.

2. THE RATIONAL FOR DESIGNSCRIPT

DesignScript is a domain specific end-user programming language, designed to be embedded within a host application. The target domains include computer aided design (CAD) tools as well as analysis applications, in this case, Ecotect. This section explains the rationale for the language and a number of the design choices that were involved.

The role of a computer language is to enable ideas and events to be recorded, abstracted and executed. The key issue in the design of a domain specific language is to create a mediating representation between the world of the user and the computational resources. The language must allow the ideas and events of the user's domain to be expressed in a logical way. Essentially the design of a domain specific language is a cognitive proposition constructed by the language designer and presented to the user. The language designer reviews the user's domain and tries to understand how the user thinks about this domain. The language designer then expresses this in the form of a language. The language designer presents the language to the user, with the following proposition: "This language is a logical abstraction of the way we think that you think about your domain."

2.1. DesignScript as a Domain-specific Language

A domain specific language is always going to involve a trade-off between the direct encoding on the domain specific constructs into the language and the abstraction of these constructs into more general forms. The direct encoding of the domain specific constructs may have the advantage of making the language immediately familiar and understandable by the user. The advantage of abstracting these concepts is to allow the user of the domain specific language to become familiar with more general computing concepts and to allow extensibility beyond the current boundaries of the domain. Therefore the trade-off is between shorter term goals (immediate understanding) and long term goals (acquisition of wider knowledge and extensibility). The art of domain specific language design is to optimize the abstraction without making this a barrier to its initial use and understanding of the language. The success of the language also depends on the willingness of the user to understand this trade-off. Using the language has the prospect of allowing the user to achieve success that he might not have thought possible, but requires some investment on his part to think more abstractly about the

domain than he might have previously considered. Success will be unlikely without some challenges and some effort. The process of developing and using this language is not just about making the user more productive. The expectation is that this language presents challenges that are essentially interesting and will spark the user's curiosity.

The specific domain is complex geometric modeling in a computer aided design system and potentially on a scale of 10^5 to 10^7 separate geometric entities and associated complex geometric modeling operations. Each entity may involve a number of operations. Additionally, the work flow which characterizes this domain requires a 'step wise' process of model building and revision and the repeated re-execution and re-generation of these models. To effectively manage this complex modeling and re-generation process requires the explicit representation of dependencies (between geometry and operations) at different steps (or states) in the modeling process. The 'natural' structure of this data lends itself to representation as a data flow language, in which variables (representing different states) are configured as a directed acyclic graph (DAG). The design choice to adopt a data flow representation is supported by the use of this paradigm in other applications which address this domain.

A conventional data flow language is used to model a series of dependent states. In this convention, each entity or variable has only one state. However in this domain, the user's conceptualization is of an entity which is the result of the application of multiple (geometric) operations. If these multiple operations had to be represented in a conventional data flow language, then this would require that the single entity had to be represented by a series of independent variables (one for each operation or state), each requiring a named identity (as a variable). But this is not how the user conceptualizes their domain.

So in the domain on which we are focusing there are in fact three concepts of state:

- 1) The state of an entity after the application of a single, atomic (geometric) operation.
- 2) The aggregation of such states to represent the complete sequence of modeling operation on a specific entity.
- 3) Different configurations of the resulting graph.

3. END-USER PROGRAMMING

As mentioned above, the role of a computer language is to enable ideas and events to be recorded, abstracted and executed. The key issue in the design of a domain specific language is to create a mediating representation between the world of the user and the computational resources. The programmer uses the language to deliver applications which are of value to the user. In the special case of end-user programming, the user and the programmer are the same person. This gives an advantage that the programmer is intimately aware of the user's domain, but as an aspiring programmer the user may be less skilled than their professional counterparts.

Essentially we are addressing two questions:

- 1) What are the specific ideas and events that the user wants to record? What are the possible ways these can be abstracted both to help the user detect, understand and extend any underlying logic in what has been recorded and to make the resulting execution more efficient?
- 2) The notion of an end-user programmer is not static. The design logic represents some progression in ambition and skill from the non-programmer to the novice programmer and potentially to the proficient programmer. But at each stage in this progression, the aspiring programmer (as a user) has to fulfill his other professional functions. He must still be productive. Therefore, we are addressing the question, "how can a computer language be sufficiently broad in scope so that it can allow the user to be productive at different levels of programming skill and at the same time help in the development of those skills: essentially to combine productivity and knowledge acquisition in the same tool.

The intent here is to create a language that is adapted to the needs of three specific types of user:

Direct-Manipulation Designer: a user who has no understanding of conventional programming languages, but is familiar with creating complex geometric representations in a computer aided design application.

Advanced Designer: a user who is also a novice programmer, who wants to contribute some end-user programming in the creation of complex geometric representations of designs.

Programming Designer: a user who is a more accomplished programmer, who is familiar with existing programming concepts and their applications.

Let us consider these three types of users in more depth. In the context of the Direct-Manipulation Designer, the DesignScript language can be used to record conventional user interaction in a computer aided design system, in a way that the user is not aware of the syntax of the language. Also in this context, the language is simply recording the user's action as is, but may reflect some underlying geometric dependencies.

How does this influence our design choices for the language: If this was the only type of user to be considered, then the language 'design choice' would be for a syntax intended for a machine-only client and not intended to be human readable. Still, this user type would still have the concept of commands being executed in a list as well as the concept of a state, including previous states which can be recovered with an undo command. These users' first steps toward additional programming may include concepts such as "recording" reusable lists of commands in macros.

In the context of an Advanced Designer, the user can begin to understand the structure and syntax of the language by seeing the correspondence between geometric constructs and their representation in the language.

Also in this context, the user can start to appreciate that a computer language can help in progressing from a literal record of user interaction to a more abstract representation that may have a number of distinct advantages: It may be more understandable, compact, re-usable, or more efficiently executed.

How does this influence our 'design choices' for the language: Once we introduce the ideas of the language being human readable and being specifically offered as an educational tool, this dramatically alters the 'design choice' for the language structure and syntax. One of the important considerations that informed these design choices was that the type of data being recorded (geometric dependencies) was best represented by an associative, data flow language. Data flow languages are a natural way to express the state-based characteristics of geometric modeling and other design representations. In fact, we extended current ideas in data flow languages to enable this approach to handle complex conditions found in replicated data.

Finally, in the context of the Programming Designer, the user may be programming exclusively in the language notion, without reference to the corresponding geometry.

Also in this context, the user may be familiar with existing programming concepts and expect to be able to port or re-implement such code within this domain-specific language.

4. THE ANALYTICAL MODEL IN ECOTECT

The Autodesk® Ecotect™ Analysis desktop application provides a framework for carrying out a wide range of performance analyses on a single analytical model. When using Ecotect itself, this analytical model is either created manually by the user or imported as 3D geometry and then prepared and augmented with the additional information required for whatever calculations are to be applied.

4.1. ANALYTICAL MODEL

Depending on the type of calculation, the analytical model requirements in Ecotect are very flexible and may include incomplete specifications. For example, solar incidence studies need no more than a geometric definition of the surfaces being studied, together with any surrounding objects that are likely to create an obstruction, and some simple material properties such as solar transmittance, surface reflectance and the refractive index of any glazing

involved. There is no requirement for a complete building model or anything even resembling a building - only the raw geometry from that part of the project that is of interest is required. This is also true of many of its other shadow, shading and acoustic calculations.

A full thermal analysis, on the other hand, requires a very different analytical model. In addition to more detailed thermal performance properties for all its materials, the application requires a complete definition of each room or space within the building as well as their spatial inter-relationships so that all possible heat flow paths in to, out of and between spaces can be determined. An energy model needs even more spatial detail, including the specific zones controlled by individual sensors or serviced by different parts of the Heating Ventilation and Air Conditioning (HVAC) system. In addition to spaces and zones, both thermal and energy models require detailed definitions of all shading devices, overhangs and site obstructions in order to accurately calculate solar gains at different times of the day and year.

The analytical models for other building performance calculations such as daylighting and visual impact tend to lie somewhere between these two extremes.

Thus, analytical models always represent some level of abstraction or simplification of the highly geometrically detailed building information models (BIM) typically generated for planning and construction purposes (Eastman 2007). Recent extensions for conceptual design and analysis for buildings in BIM authoring systems, for example, Autodesk® Revit®, have made them capable of automatically generating a range of analytical models directly from their project databases. As these databases become able to store more performative design information, the more analytical models become an additional subset of the building information model and are therefore able to be quickly generated at any time without requiring additional external input.

4.2. ANALYSIS ENGINE

Ecotect's flexible analytical model requirements are something that can be utilised by other applications that need to perform analysis. This would allow users to select the complexity or simplicity of their models to suit their specific needs, or break very large calculations up into many smaller calculations (each specific to some part of the

overall model) and still obtain accurate and useful performative feedback from within the host application.

If the analytical model can be entirely and reliably generated and re-generated from a source database – where all of the required geometric and material property information is available – and the results of the calculations are being used or visualised within the controlling application, then the underlying analytical model and the Ecotect interface can be implicit and hidden. When used this way, Ecotect and its components can be thought of as a set of analysis solvers that provide transparent services to the controlling application. This is the approach that was taken in the integration with DesignScript.

5. INTEGRATION OF DESIGNSCRIPT & ECOTECT

The initial experiments that utilise Ecotect from within DesignScript concentrated on the analysis of incident solar radiation. In many design situations, the availability or otherwise of solar radiation over a site or on significant façade elements is a major driver of building form.

Thus the intent was to investigate what could be done algorithmically with this kind of calculation, starting with relatively simple dynamic forms.

5.1. Incident Solar Radiation Example

In the calculation of incident solar radiation, model geometry can be considered in two separate groups:

- Those objects that could potentially obstruct the sky when viewed from the perspective of the surface of interest, and
- Those objects that are behind or below the surface so can never be an obstruction.

This information can significantly accelerate solar calculations as all non-reflective objects in the second group can effectively be ignored. Whilst the specific objects contained in these two groups will likely be different for each surface of interest, there exist very fast techniques for culling out-of-view geometry that have been developed for ray-tracing purposes and that have direct application here.

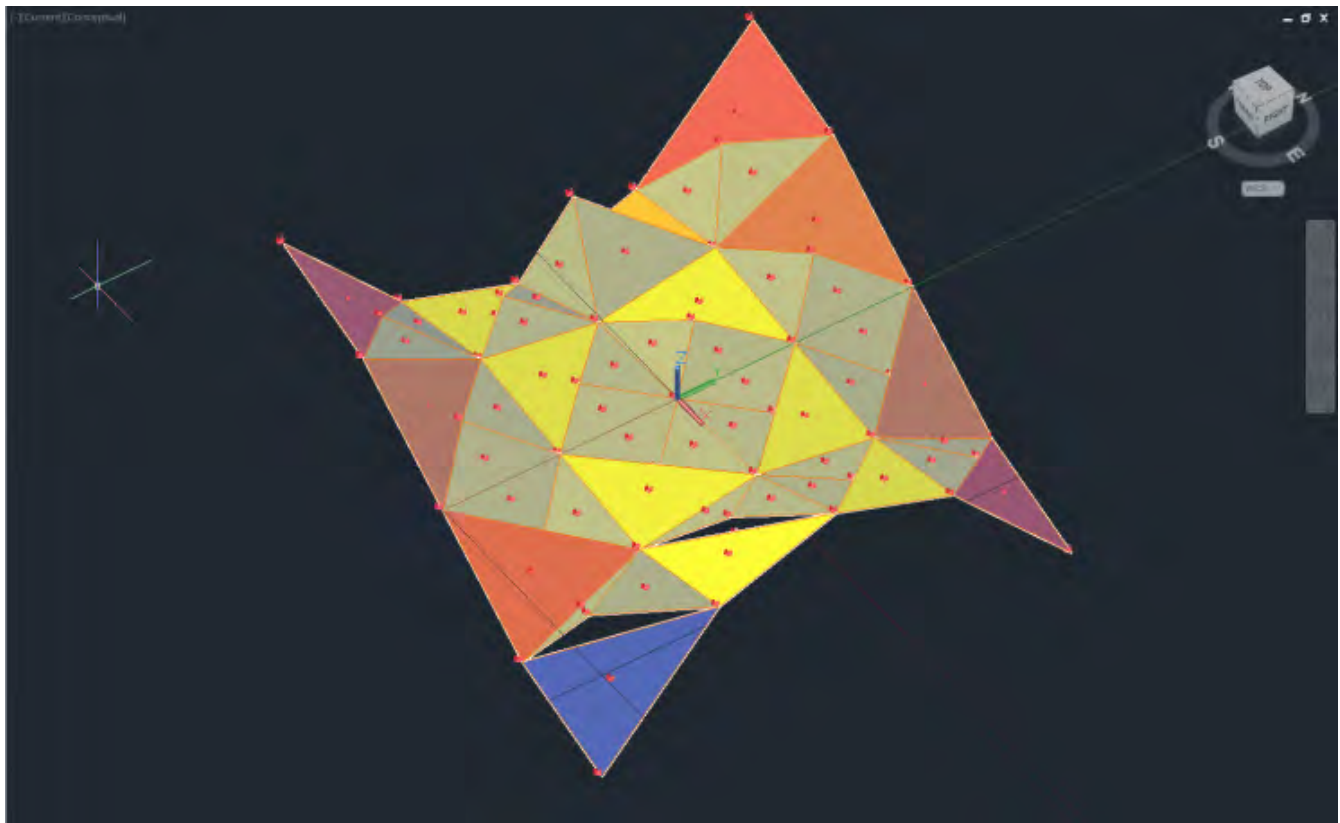


Figure 1. DesignScript roof model with insolation calculation using Ecotect.

In the case of DesignScript, those objects in the first group, termed obstructions, can be further divided into two additional groups:

- Static obstructions that will not move or change throughout the calculations, and
- Dynamic obstructions that could move during, or as a result of, the calculations.

These are very useful groupings for computation efficiency as it means that the effects of static obstructions need only be calculated once and the results may be cached for re-use during each dynamic iteration. Typically the number of potentially dynamic obstructions is much less and these can be accommodated in a separate calculation performed on each iteration of the simulation. As Ecotect uses shading masks to cache overshadowing data (Marsh 2005), it is possible to quickly merge the effects of static and dynamic obstructions by simply maintaining two shading masks for each surface of interest (static and dynamic) and accumulating them after dynamic effects are updated.

Thus, even for models with very large numbers of static obstructions, the speed of dynamic updates is a determinant only of the number of surfaces of interest and potentially dynamic obstructions. These numbers are known by DesignScript ahead of each dynamic interaction so it can react by, for example, refactoring the geometry during dynamics and then returning to full detail for one final and more accurate update at the end of the dynamic transaction.

Also, when generating and maintaining the analytical model in Ecotect, DesignScript need only send the positional information for all static obstructions once. Then, on each dynamic iteration, only those potentially dynamic obstructions that have actually changed need be updated before the next round of calculations.

5.2. Usage Test Case

To test the DesignScript and Ecotect integration, a simplified roof model was built, as follows (see Figure 1):

- A rectangular array of ‘base points’ was constructed
- A single ‘focal point’ was positioned within the field of ‘base points’
- The focal point was used as a ‘repeller’ to create a new field of ‘roof points’ based on the direction and distance of each of the field of ‘base points’ from the focal point. This created a humped roof configuration
- The $n*m$ field of ‘roof points’ was used to create an $(n-1)*(m-1)$ array of ‘roof polygons’
- A shading component was created in DesignScript as a user defined class. This shading component takes a base polygon as one of its inputs and also a ‘key direction’ point. Each shading component consists of two triangular facets in a folded pyramidal configuration based on the two sides of the polygon opposite ‘key direction’ point. The height of this pyramidal configuration was based on the dimensions of the base polygon and an ‘aspect ratio’ parameter
- This shading component was arrayed over the array of ‘roof polygons’, using the sun position as the key direction point.
- Various other faceted geometry constructs where modelled to represent a ‘context model’.
- An Ecotect analysis component was created in DesignScript as a user defined class, but linked to external functions in the Ecotect library. This component takes three inputs:
 - 1) ‘Context’ model
 - 2) ‘Active’ model (to compute insolation)
 - 3) The city location

Both the context and active model are considered when calculating the shading on the active model. The Ecotect analyses component creates (as output) a new set of triangles co-located with the active model, generating and updating shading masks for each triangle that are used to colour code them to indicate levels of insolation, a measure of solar radiation energy received on a given surface area in a given time.

In this example, the facets of the shading components were used as the active model, the additional faceted geometry was used as the context model, and the city location was selected as San Francisco.

This model allowed a number of different design options to be explored:

- The size and spacing for the rectangular array of ‘base points’ could be changed (thus altering the size and ‘granularity’ of the roof)
- The location of the focal point could be changed (thus altering the ‘humpiness’ of the roof)
- The design and aspect ratio of the shading component could be altered
- The model could be reoriented
- Alternative city locations could be explored

6. CONCLUSION AND FUTURE DIRECTION

One of the benefits of using Ecotect in this study was that it allowed access not just to the incident solar radiation solver itself, but also to many of the components of that solver. Thus, rather than simply set up the context, run the calculation and interpret the results, it is possible for an external application to create individual shading masks, select the specific objects to include/exclude in the calculation of each mask, merge the results of several masks together and thereby effectively control, if it chooses, the input and sequencing of each step within the overall calculation process. In addition, individual component classes that feed into the solver can be utilised, allowing access to solar positions, incidence angles, hourly weather data and the calculation of instantaneous insolation values on any surface at any time.

This is an important point as it illustrates a significant change in paradigm from typical simulation applications and their invocation workflows. Traditional black-box solvers that demand very specific input and produce calculation results in a set format are of only limited use to tools that use interactive manipulation or generative algorithms. Whilst they can be accommodated, they do not provide the degree of flexibility and customisability that these kinds of tools are capable of taking advantage of.

Like the insolation example described here, there are likely to be many optimisation strategies that are applicable to different generative algorithms and specific interaction methods – and equally unlikely that a solver developer will be able to envisage all of them and implement the required interfaces. What is clear is that, to take full advantage of these potential optimisations, solvers need to provide access to a range of services at many different levels within their calculation processes. This allows external application developers to not only bring additional insight, but to implement, test and refine those insights to potentially improve the workflows involved, ideally supporting a continuum of novice to expert designers.

Whilst it can obviously be argued that this level of access could lead to misuse, the counter argument is that it could equally lead to more effective and appropriate use – with many more minds than just the initial solver developer actively considering how to configure and tune the underlying components.

There is a lot more work that can be done here on optimising both the processes and the data flows between Ecotect and DesignScript. However, what is important is that much of this is entirely within the control of the scripts themselves as the interface allows them to maintain their own results sets as internal data and both pre and post-process calculated information in any way that is appropriate to a particular calculation or dynamic model.

References

- AUTODESK ECOTECT ANALYSIS. 2011. WWW.AUTODESK.COM/ECOTECT-ANALYSIS
- Marsh, A.. 2005. THE APPLICATION OF SHADING MASKS IN BUILDING SIMULATION. NINTH INTERNATIONAL IBPSA CONFERENCE, MONTREAL, CANADA. PP. 725-732.

Leveraging Cloud Computing and High Performance Computing Advances for Next-generation Architecture, Urban Design and Construction Projects

Francesco Iorio¹, Jane L. Snowdon²

¹ Autodesk Research
210 King Street East
Toronto, ON, Canada, M5A1J7
francesco.iorio@autodesk.com

² IBM T. J. Watson Research Center
1101 Kitchawan Road, Route 134
Yorktown Heights, NY, USA, 10598
snowdonj@us.ibm.com

Keywords: Cloud Computing, High Performance Computing, Architecture, Urban Design, Construction, Simulation, Optimization.

Abstract

Architecture and urban design projects are constantly breaking barriers of scale and complexity and continuously seek improved efficiency, sustainability, building energy performance, and cost-effectiveness. Simulation and large-scale data processing are now fundamental elements of this process. Recent advances in algorithms and computational power offer the means to address the complex dynamics of an integrated whole building system. However, scalability is a significant barrier to the realization of whole building systems tools for design, control and optimization.

This position paper presents a set of techniques such as fast design parameter-space exploration, large-scale high-accuracy simulation, and integrated multi-disciplinary optimization for semi- or fully-automated designs. These techniques are extremely computing intensive, and have traditionally only been available to the research community. But, once enabled by advances in cloud computing and high performance computing, these techniques can facilitate the interactive design process resulting in improved outcomes and reduced development cycle times.

1. INTRODUCTION

According to the World Business Council for Sustainable Development (2009) and the U. S. Department of Energy (DOE) (U.S. DOE, Energy Information Administration 2008a), buildings account for 40% of the world's total energy consumption and, in 2005, and nine gigatons of global carbon dioxide (CO₂) emissions, well

above the transportation and industry sectors. In the United States alone, commercial and residential buildings account for 38% of all CO₂ emissions and 72% of electricity consumption (U.S. DOE, Energy Information Administration 2008b; U.S. DOE, Environmental Information Administration 2008c). Furthermore, buildings use 13.6% of all potable water, or 15 trillion gallons per year, and 40% of raw materials globally, which is equivalent to 3 billion tons annually (Hutson et al. 2004, Lenssen and Roodman 1995). Much of the energy consumption by commercial buildings is spent on lighting (26%), followed by heating and cooling (13% and 14%, respectively) (U.S. DOE 2007). Investing in energy efficient light bulbs and insulation materials and in automated shading has proven to reduce the energy demands on cooling and lighting (Lee et al. 2007). However, incremental improvements achieved by implementing individual energy efficient technologies alone are not sufficient to successfully meet the challenging objectives set forth by the Intergovernmental Panel on Climate Change (IPCC) and other directives issued by cities, for example PlaNYC in New York City (IPCC 2007, PLANYC 2007).

Reducing energy consumption by buildings and improving energy efficiency in buildings are two approaches to help achieve the goals. Frost and Sullivan (2009) cites the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) stating only 11% of building cost is construction, while 50% of the lifecycle cost is in the operation of the building.

Nobel Prize laureate and U. S. Secretary of Energy Dr. Steven Chu stated: "We need ... computer design tools for commercial and residential buildings that enable reductions

in energy consumption of up to 80 percent with investments that will pay for themselves in less than 10 years.” (Chu 2009). This grand challenge will require a system integration approach to building design, aided by computer tools with embedded energy analysis, computer monitoring, and real-time control of building systems.

A smarter building can be defined as integrating major building systems on a common network where information and functionality between systems is shared to improve energy efficiency, operational effectiveness, and occupant satisfaction (IBM, 2010). A smarter building is a complex system of systems that span heating and air conditioning, lighting, security, access control, entertainment, people movers, water, and monitoring, control and maintenance systems. Ideally, these systems would have well managed and integrated physical and digital infrastructures that make the building safe, comfortable, and functional for its occupants and sustainable for the environment. Through the use of sensors, digital smart IP enabled meters and submeters, digital controls, and analysis tools to automatically monitor and control services for its users, buildings can be studied in much greater detail to investigate whole systems solutions. Since the average lifetime of a commercial building is fifty years, it is crucial to focus on the design, construction, and operation of both new buildings and retrofits.

An integrated design process (IDP) for new buildings and building retrofits that incorporates energy simulation, lighting analysis, computational fluid dynamics and digital information about a building’s structure, occupancy, and plumbing, electrical and mechanical systems and costs, will enable architects, builders and contractors, facilities managers, and building owner/operators to achieve optimized building performance through the selection of the best materials, windows, equipment, subsystems and processes. These virtual models and tools, combined with cloud and high performance computing (HPC), will improve productivity and allow rapid assessment of design alternatives (Lunzer and Hornbæk, 2008) to optimize the energy performance of a building. A shift in culture among building stakeholders is needed to adopt collaboration, interoperability, and the effective and active use of operational building data as the way of the future.

Cloud computing is an emerging style of computing in which applications, data and IT resources are provided as

services to users over the web. Cloud computing is a way of managing large numbers of highly virtualized resources such that, from a management perspective, they resemble a single large scalable resource, which in turn can be used to deliver services. Several cloud computing paradigms exist, for example, Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) (Zhu et al. 2009). This is a disruptive computing and business paradigm, poised to change the traditional organizations’ IT infrastructure management and business practices, and to dramatically improve productivity and cost effectiveness of software solutions. As an infrastructure paradigm, cloud computing is extremely well suited to efficiently handle traditional “web” workloads, whereas its adoption in different areas of computing is not without technical and strategic challenges that, without solution, inhibits acceptance and adoption.

We present an analysis of the benefits and challenges of using cloud computing and HPC in the context of the design and operation of next-generation architecture and urban design projects, and suggest some potential solutions.

2. ARCHITECTURE, URBAN DESIGN PROJECTS AND SOFTWARE APPLICATIONS

Architecture and construction projects require a vast array of software solutions that range from Product Lifecycle Management (PLM) to Computer Aided Design (CAD) systems, logistics, and energy usage simulation to environmental impact analysis. These software tools improve the productivity of designers, architects, engineers, builders, managers and facility workers who need to operate in the best way to successfully complete the assigned projects within budget and time constraints.

Simulation has been traditionally used in architectural and engineering projects to validate designs created by individual domain experts, and is usually performed on reduced-complexity system models in order to contain the overall analysis time and cost. The United States Green Building Council (USGBC) encourages the use of simulation in its Leadership in Energy and Environmental Design (LEED) certification process.

Today’s building energy simulators are not able to capture the multi-scale dynamics of a whole building system and the majority of models provide results relevant only to steady state conditions. More recently though, the

requirements for additional efficiency and cost reduction in large-scale projects, combined with the availability of more sophisticated tools to evaluate and predict the resulting project's performance (e.g. a building's total energy consumption) as accurately as possible, are improving the collaborative nature of design. With these more sophisticated tools come increases in the amount of data produced, which ultimately requires increases in compute power to process various simulation-driven optimization strategies.

The PLM range of software applications is relatively easy to migrate to the cloud environment, as their fundamental operations are of a transactional nature and large, distributed databases have been commonly used in cloud-based web applications for some time. These characteristics facilitate the economic justification of a cloud-based business model. CAD and more general design applications, on the other hand, especially when applied to large-scale projects, present significant challenges that must be addressed to allow the full exploitation of cloud computing economies of scale.

Simulation is also a discipline, which, despite the vast amount of expertise accumulated over recent decades, has not been fully exploited as a process optimization tool, mostly due to the inherently high costs associated with dedicated HPC resources. Fully addressing these challenges opens up tremendous new opportunities that can revolutionize the way architecture and urban design projects are planned and executed.

Below, computing advances are outlined where scalable resources can have a significant impact on the development of solutions in support of the grand challenges of this domain.

2.1. Intelligent Application Execution Prediction

Design software is characterized by a workflow composed of alternating stages: a user performs an operation (action stage) and then reviews the results of the operation (review stage). When large amounts of computing resources are available, computing power can be dedicated to predict a user's behavior in order to reduce the perceived application latency. Implicitly, the more action-review workflow cycles that can be performed during the design process, the better the outcome. Predictive methods can greatly reduce the time taken for each cycle thereby

allowing more cycles to take place in the same amount of time.

In the review stages, while the user is evaluating the effects of the last operation or set of operations, the application will analyze the user's workflow and attempt to predict which set of operations the user is most likely to perform next. Then it will allocate computing resources to actually perform in parallel all those operations, speculating that one of them will actually be selected by the user in the next action stage (Igarashi and Hughes, 2001). If the speculation is successful, the latency of the selected operation can be significantly reduced by overlapping it with the time the user takes to review his/her progress.

2.2. Fast Parameter-space Exploration

Design, architecture and engineering are disciplines that require the evaluation of numerous options and the expertise of individual domain experts provides guidance towards the choices that must be made at all stages of a project. Despite the skill level of a project's individual contributors, there is a limit to the variety of options and combinations of components an architect or an engineer can evaluate in a reasonable amount of time during each design cycle on a large project.

It is necessary to develop effective methodologies and tools to improve the speed of design space exploration for optimal configurations that reduce energy consumption, improve overall efficiency, and/or the total cost of the project itself in a robust way so that variations in the different uses of buildings, energy loads, and degradations in equipment operation can be considered in a timely manner. Access to large amounts of computing power would lead to the ability to evaluate hundreds or thousands of different design variations and their effects on the overall project. This would provide an opportunity for more informed decision-making and for further confirmation of a choice made by a subject matter expert.

2.3. Large-scale Simulation

Next-generation architecture projects face increasing budget constraints, demands for greater efficiency and reductions in environmental impact. As the multitude of variables inherent in construction projects increases and the need for more sophisticated simulation models for design and construction grows, for example to better understand the real non-linear behavior of materials, a much higher level of

detail in the simulations are required in order to have the most accurate possible analyses. In the past, greatly simplified models have been considered sufficient for estimating energy consumption. While the potentially double-digit error ranges that result may be acceptable for buildings expecting typical energy consumption, these simplifications are not viable for a low-energy building where a difference of 10% would be unacceptable for a theoretically net-zero energy building.

Complex and very detailed analysis models require simulation to a scale only available to large private, academic, and government research laboratories. Recent advances in computational power, computer software, and clever mathematical algorithms and heuristics offer industries access to such tools across a building's life cycle providing the means to address the complex dynamics of a whole integrated building system. Burns et al. (2010) discuss the needs, the current state and gaps, and the research approach for simulation-based design.

2.4. Multi-disciplinary Optimization for Semi-automatic Design

Multi-disciplinary optimization can be defined as the field of engineering that uses optimization methods to solve design problems incorporating a number of disciplines. The importance of sound software design practices is often underestimated when developing large-scale integrated simulations. Typically each discipline develops or uses software modules and well-defined interfaces between these modules are lacking.

As a further refinement to fast parameter-space exploration, numerical analysis principles can be applied to find optimal project configurations based on soft and hard constraints such as energy consumption, carbon emissions, among others. Architecture designs can be defined as completely parametric problems as illustrated in Shea, Aish and Gourtovaia (2005). The concept can be extended to define not only shapes but also other characteristics of architecture projects, such as the choice of construction materials for every component, placement of fixtures, shape and size of windows, mechanical systems integration strategies, and so on.

When combined in a single problem domain, the number of variables easily overwhelms the capabilities of an individual simulation. While analysis of the results of

individual simulations and fast parameter space exploration can provide strong guidance, multi-disciplinary optimization combines analytical methods with domain-specific heuristics to provide a very integrated decision-making tool. This technique can result in dramatic improvements in overall cost reduction and efficiency, and if applied to a vast majority of future architecture and engineering projects, it can substantially reduce the environmental impact of next-generation urban planning, but not without substantial increases in the complexity of the system.

3. CHALLENGES

3.1. CAD User Experience

Interactivity and complex 3D visualization are both fundamental characteristics of modern CAD systems. The main source of potential compromises in using a cloud-based interactive computing environment is the latency introduced by the communication network, which in most circumstances is the Internet. The very low response latency typical of CAD applications running on a user's workstation is hard to replicate while running in the cloud, although the large amount of computing power required for increasingly important operations like complex simulation and multi-disciplinary optimization is not available in a desktop system.

3.2. Data Transfer and Data Security

Organizations operating in the engineering sector generate large amounts of security sensitive data such as project plans, CAD drawings, and project schedules. Transferring security sensitive data to a public cloud is then a serious trust issue for some organizations, who cannot afford the risk of having data stolen or compromised. Transferring the data to the cloud for on-demand processing can be impractical and cumbersome, especially in the presence of large datasets, which is common for large engineering projects; some form of incremental data transfer is then required. Buecker et al. (2009) provide technical guidance on cloud security.

Faced with exponential data growth and increasing regulatory requirements, organizations want to protect their data and applications at reduced costs. Services exist today that enable security-rich managed protection of critical data on-site or off-site, help reduce the total cost of ownership with cloud-based data backup solutions, and ease management of various industry regulation requirements.

3.3. Simulation and Algorithm Scalability in the Cloud

The scale and complexity of the problems that future simulation algorithms will be required to solve, combined with the fundamental architecture of cloud systems, does not suit the traditional HPC model which relies on a much higher level of predictability from the underlying hardware and infrastructure. Standard cloud nodes are not well suited to run classical HPC algorithms, so new variations of algorithms are needed to make better use of the commodity hardware based cloud configurations. Only recently cloud providers are beginning to offer on-demand services for high performance computing.

3.4. Fault Tolerance

Moving an entire organization's business on to cloud systems can have dramatic consequences in case of faults like network outages. The benefit and the appeal of using public or semi-private computing clouds would be greatly diminished if fault-tolerance were not appropriately handled and if the solutions were not properly demonstrated to organizations. Extreme flexibility in the underlying infrastructure needs to be factored in when designing solutions that must be able to accommodate sudden faults and/or sudden increases in computing demands.

4. POTENTIAL SOLUTIONS

4.1. CAD User Experience

In order to avoid disrupting the user experience, a careful evaluation of applications and user behavior is necessary. CAD applications exhibit a peculiar kind of workflow that alternates phases of intense interaction to perform simple operations and review, to phases of intense computation, necessary to perform complex operations. Systems like "Snowbird/Snowflock" described in Lagar-Cavilla et al. (2007) are capable of detecting and reacting to this alternation performing incremental, transparent, live virtual machine migration from a user's workstation to the cloud and vice-versa. This capability provides an extremely useful new paradigm to extend application computing scalability using vast cloud-based resources, while minimizing changes in the application's user interface modules and general behavior.

4.2. Data Transfer and Data Security

To overcome data transfer latency to the cloud, two strategies offer some potential solutions:

- I/O operations originating from a users' workstation and directed to an organization's central storage system such as a Storage Area Network can be intercepted by dedicated appliances, which can batch, compress and encrypt a number of popular storage protocols such as iSCSI, and forwarded to the cloud's storage system. In case of network fault the operations are queued and transmitted after the network is restored.
- User's workstations are partially virtualized, with "shadow" virtual machines running in the cloud to replicate all the user's operations, including storage accesses.

Any of these strategies allow for a persistent copy of the relevant data to be maintained and updated in the cloud, ready for use by a number of scalable computing workloads.

Data security can also be augmented by using partitioning schemes, such that a complete dataset is never completely available in a single system, similar to distributed, peer-to-peer content distribution systems; this strategy is particularly well suited for large datasets representing models on which simulation is performed, as domain decomposition is then necessary to distribute the calculation over a large amount of computing nodes.

The application of distributed data exchange security models based on virtualization like Trusted Virtual Domains (TVD) in Linwood Griffin et al. (2005) and Bussani et al. (2005) would also be a further step towards the comprehensive system required by organizations to entrust data to a cloud provider.

4.3. Simulation and Algorithm Scalability in the Cloud

Due to the extreme commoditization of the architecture of cloud systems, there are several unique characteristics that must be accounted for to avoid hard scalability limitations. Still, several lessons can be learned from past research in distributed computing to leverage the very dynamic nature of these systems.

First, workloads must be created having a single virtual machine as the fundamental computing entity, relying on as little external information and communication as possible.

Second, new programming models that account for the dynamic and heterogeneous nature, and the massive scale of

cloud systems, need to emerge to mitigate algorithm development complexity and to avoid continuous manual refactoring subject to the volatile underlying platforms and networks. For example, taking some hints from numerical optimization as described in Iorio (2009), an effective layering strategy can be highly beneficial.

Third, when the individual computing entity is a pre-configured virtual machine, adaptive, dynamic domain decomposition and computing strategies must also be adopted to leverage the extreme amount of computing potentially available to cloud-based applications.

4.4. Fault Tolerance

The fully virtualized nature of cloud systems allows for a more streamlined management of fault tolerance. Applications and workloads designed to run within a virtual machine are completely agnostic of their location, their individual performance, and the exact topology of the network fabric.

The characteristics of the infrastructure and the constraints posed by it on the applications actually result in a much easier exploitation of multiple fault-tolerance strategies that depend on this strict separation of information. As an example, modern cloud infrastructures can produce full virtual machine snapshots in a very short time, and upon detection of a software or hardware issue can provision a new virtual machine that maintains the same characteristics of the one that failed.

This feature makes the adoption of a “checkpointing” fault-tolerance strategy much easier from the application developer standpoint in a very elegant way. Traditional HPC systems, based on a message passing interface, either do not possess this characteristic, or must be designed and implemented at the application level, which is much more time consuming and error prone.

5. LIVING LAB CASE STUDIES

The discrepancy between the modeled and actual energy performance in completed construction projects is becoming clearer to practitioners and certification bodies. This trend, together with the recognition of the need for widespread implementation of massive energy efficiency improvements in both new and existing buildings, has led to a number of projects that provide testbeds for experimentation called “living labs.” For a commercial building, the Digital 210

King project in Toronto (Attar et al. 2010) brings together a detailed Building Information Model (Eastman, 2007) with sensor data collected from the building control system as well as meters and submeters. At a sustainable community level, the Millennium Water Olympic Village project in Vancouver (Bayley et al. 2009) applied a multitude of efficiency measures to great effect achieving a close to net-zero energy community. At an urban level, the highly publicized yet-unrealized Masdar City project in Abu Dhabi (www.masdar.ae) aims to be entirely fueled by renewable resources, primarily from solar energy.

At regional and national levels, the U.S. Department of Energy recently announced the establishment of an Energy Innovation Hub focused on developing technologies to make buildings more energy efficient (U.S. DOE 2010). Led by The Pennsylvania State University, the Energy Innovation Hub is located at the Philadelphia Navy Yard Clean Energy Campus and will function as a living lab. Hub members include nine universities, two U.S. National Laboratories (Lawrence Livermore and Princeton), and five industrial firms including IBM. A highly multidisciplinary approach is underway to take full advantage of the diverse knowledge sets needed to develop a significantly improved outcome. As lead for Integrated Design Processes and Computational Tools, IBM is collaborating with other Hub members including Carnegie Mellon University, Lawrence Livermore National Lab, Penn State, Purdue, United Technologies Research Center, and Virginia Tech on this task. IBM will develop a cloud computing framework and infrastructure to support the interoperability of information and tools that will help ensure a range of small to mid-sized to large firms can take advantage of these technologies. In addition, IBM will implement performance monitoring and diagnostic systems for the integrated control systems.

6. CONCLUSION AND FUTURE OUTLOOK

The cloud environment offers the very compelling opportunity of using computing power on a scale previously unavailable to most users of design and engineering software applications. The main challenges are thus both in defining novel avenues to use that computing power and in ensuring that the user experience remains familiar to widely adopted standards, augmenting the user’s capabilities rather than exacting new workflow compromises.

Promising techniques that leverage the already virtualized nature of the cloud environment exist and can be

used to exploit the vast opportunities cloud computing has to offer to the engineering community, and will make it a viable option for architecture and engineering organizations to increase the efficiency of their practices and operations.

Moreover the constant reduction in the cost of cloud-based computing resources will dramatically increase the amount of computing power available on demand, thus enabling full democratization of an entire portfolio of powerful tools, with the prospect of both reductions in project costs and environmental impact, and increases in overall efficiency over the entire project lifecycle.

Acknowledgements

Francesco Iorio acknowledges all members of Autodesk Research for the excellent insight, and Prof. Eyal de Lara at University of Toronto for their work on cloud systems and for the discussions on application behavior in cloud computing.

Jane Snowden gratefully acknowledges Bruce D'Amora and Florence Hudson for their review and useful comments.

References

- ATTAR, R., PRABHU, V., GLUECK, M., AND KHAN, A. 2010. 210 KING STREET: A DATASET FOR INTEGRATED PERFORMANCE ASSESSMENT. IN PROCEEDINGS OF THE 2010 SPRING SIMULATION MULTICONFERENCE (SPRINGSIM '10). ACM, NEW YORK, NY, USA, ARTICLE 177, 4 PAGES.
- BAYLEY, R. 2009. THE CHALLENGE SERIES – MILLENNIUM WATER: THE SOUTHEAST FALSE CREEK OLYMPIC VILLAGE.
- BUECKER, A., LODewIJKX, K., MOSS, H., SKAPINETZ, K., WAIDNER, M. 2009. CLOUD SECURITY GUIDANCE: IBM RECOMMENDATIONS FOR THE IMPLEMENTATION OF CLOUD SECURITY. IBM REDPAPER, 2009.
- BURNS, J., COLLIS, S., GROSH, J., JACOBSON, C., JOHANSEN, H., MEZIC, I., NARAYANAN, S., WETTER, M. 2010. APPLIED AND COMPUTATIONAL MATHEMATICS CHALLENGES FOR DESIGN, DYNAMICS AND CONTROL OF ENERGY SYSTEMS. DRAFT WHITEPAPER V3. http://www.engineering.ucsb.edu/~mggroup/wiki/images/1/17/WP_Computation_Bulidings_November_2010_Version_3.docx
- BUSSANI, A., ET AL. 2005. TRUSTED VIRTUAL DOMAINS: SECURE FOUNDATIONS FOR BUSINESS AND ITS SERVICES. IBM RESEARCH REPORT RC23792.
- CHU, S. 2009. STATEMENT OF STEVEN CHU SECRETARY OF ENERGY BEFORE THE COMMITTEE ON SCIENCE AND TECHNOLOGY. U.S. HOUSE OF REPRESENTATIVES, WASHINGTON, D.C., U.S.A. <http://www.energy.gov/news/7055.htm>
- EASTMAN, C., TEICHOLZ, P., SACKS, R. AND LISTON, K. 2007. BIM HANDBOOK: A GUIDE TO BUILDING INFORMATION MODELING FOR OWNERS, MANAGERS, DESIGNERS, ENGINEERS AND CONTRACTORS. JOHN WILEY & SONS, NEW JERSEY, 108.
- ENERGY EFFICIENCY AND RENEWABLE ENERGY, U.S. DEPARTMENT OF ENERGY, 2007. 2007 BUILDINGS ENERGY DATA BOOK. http://buildingsdatabook.eren.doe.gov/docs/DataBooks/2007_BEDB.pdf
- FROST AND SULLIVAN 2009. BRIGHT GREEN BUILDINGS: CONVERGENCE OF GREEN AND INTELLIGENT BUILDINGS, <HTTP://WWW.CABA.ORG/BRIGHTGREEN>
- HUTSON, S.S., BARBER, N.L., KENNY, J.F., LINSEY, K.S., LUMIA, D.S., AND MAUPIN, M.A., 2004. ESTIMATED USE OF WATER IN THE UNITED STATES IN 2000: RESTON, VA., U.S. GEOLOGICAL SURVEY CIRCULAR 1268, 46 P.
- IBM 2010. SMARTER PLANET PRIMER. <HTTP://SMARTERPLANET.TUMBLR.COM/POST/58347576/A-SMART-PLANET-PRIMER-TO-VIEW-FULL-SCREEN-CLICK>
- IGARASHI, T. AND HUGHES, J. 2001. A SUGGESTIVE INTERFACE FOR 3D DRAWING. UIST '01. ACM, PP. 173-181.
- IORIO, F. 2009. ALGORITHMS RE-ENGINEERING AS A FUNDAMENTAL STEP TOWARDS EXPLOITABLE HYBRID COMPUTING FOR ENGINEERING SIMULATIONS. CECAM WORKSHOP: ALGORITHMIC RE-ENGINEERING FOR MODERN NON-CONVENTIONAL PROCESSING UNITS.
- IPCC 2007: METZ, B., DAVIDSON, O.R., BOSCH, P.R., DAVE, R., MEYER, L.A. (EDS). 2007. CLIMATE CHANGE 2007: MITIGATION OF CLIMATE CHANGE. CONTRIBUTION OF WORKING GROUP III TO THE FOURTH ASSESSMENT REPORT OF THE INTERGOVERNMENTAL PANEL ON CLIMATE CHANGE, CAMBRIDGE UNIVERSITY PRESS, 851 PP.
- LAGAR-CAVILLA, H. ANDR'ES, TOLIA, N., DE LARA, E., SATYANARAYANAN, M., O'HALLORON, D. 2007. INTERACTIVE RESOURCE-INTENSIVE APPLICATIONS MADE EASY. ACM/IFIP/USENIX 8TH INTERNATIONAL MIDDLEWARE CONFERENCE. PP. 143-163.
- LEE, E. S., HUGHES, G. D., CLEAR, R. D., FERNANDES, L. L., KILICCOTE, S., PIETTE, M. A., RUBINSTEIN, F. M., SELKOWITZ, S. E. 2007. DAYLIGHTING THE NEW YORK TIMES HEADQUARTERS BUILDING, FINAL REPORT: COMMISSIONING DAYLIGHTING SYSTEMS AND ESTIMATION OF DEMAND RESPONSE, LAWRENCE BERKELEY NATIONAL LABORATORY, BERKELEY, CA.
- LINWOOD GRIFFIN, J., JAEGER, T., PEREZ, R., SAILER, R., VAN DOORN, L., C'ACERES, R. 2005. TRUSTED VIRTUAL DOMAINS: TOWARD SECURE DISTRIBUTED SERVICES. FIRST WORKSHOP ON HOT TOPICS IN SYSTEM DEPENDABILITY.
- LUNZER, A. AND HORNBAEK, K. 2008. SUBJUNCTIVE INTERFACES: EXTENDING APPLICATIONS TO SUPPORT PARALLEL SETUP, VIEWING AND CONTROL OF ALTERNATIVE SCENARIOS. ACM TRANSACTIONS ON COMPUTER-HUMAN INTERACTION. 14, 4, ARTICLE 17. 44 PP.
- PLAN NYC: A GREENER, GREATER NEW YORK. 2007. http://www.nyc.gov/html/planyc2030/downloads/pdf/full_report.pdf
- ROODMAN, D.M., LENSSSEN, N. 1995. A BUILDING REVOLUTION: HOW ECOLOGY AND HEALTH CONCERNS ARE TRANSFORMING

- CONSTRUCTION. WORLDWATCH PAPER 124, WORLDWATCH INSTITUTE, WASHINGTON, D.C.
- SHEA, K., AISH, R., GOURTOVAIA, M. 2005. TOWARDS INTEGRATED PERFORMANCE-DRIVEN GENERATIVE DESIGN TOOLS. AUTOMATION IN CONSTRUCTION, 14(2). PP. 253-264.
- U.S. DEPARTMENT OF ENERGY, ENERGY INFORMATION ADMINISTRATION 2008a. EMISSIONS OF GREENHOUSE GASES REPORT. [http://www.eia.doe.gov/oiaf/1605/ggrpt/pdf/0573\(2008\).pdf](http://www.eia.doe.gov/oiaf/1605/ggrpt/pdf/0573(2008).pdf)
- U.S. DEPARTMENT OF ENERGY, ENERGY INFORMATION ADMINISTRATION 2008b. ASSUMPTIONS TO THE ANNUAL ENERGY OUTLOOK. [http://www.eia.doe.gov/oiaf/aeo/assumption/pdf/0554\(2008\).pdf](http://www.eia.doe.gov/oiaf/aeo/assumption/pdf/0554(2008).pdf)
- U.S. DEPARTMENT OF ENERGY, ENVIRONMENTAL INFORMATION ADMINISTRATION 2008c. EIA ANNUAL ENERGY OUTLOOK 2008. [http://www.eia.gov/oiaf/aeo/pdf/0383\(2008\).pdf](http://www.eia.gov/oiaf/aeo/pdf/0383(2008).pdf)
- U.S. DEPARTMENT OF ENERGY. 2010. PENN STATE TO LEAD PHILADELPHIA-BASED TEAM THAT WILL PIONEER NEW ENERGY-EFFICIENT BUILDING DESIGNS. <http://www.energy.gov/news/9380.htm>
- WORLD BUSINESS COUNCIL FOR SUSTAINABLE DEVELOPMENT. 2009. TRANSFORMING THE MARKET: ENERGY EFFICIENCY IN BUILDINGS. www.wbcsd.org.
- ZHU, J., FANG, X., GUO, Z., NIU, M., CAO, F., YUE, S., LIU, Q. 2009. IBM CLOUD COMPUTING POWERING A SMARTER PLANET. CLOUDCOM 2009, LNCS 5931, PP.621-625.

Session 4: Generative Design

- 79 **Generative Fluid Dynamics: Integration of Fast Fluid Dynamics and Genetic Algorithms for wind loading optimization of a free form surface** OUTSTANDING PAPER AWARD
ANGELOS CHRONIS and ALASDAIR TURNER
University College London
- 87 **Use of Sub-division Surfaces Architectural Form-Finding and Procedural Modelling**
SHAJAY BHOOSHAN and MOSTAFA EL SAYED
Computation and Design Group and Zaha Hadid Architects
- 95 **Irregular Vertex Editing for Architectural Geometry Design**
YOSHIHIRO KOBAYASHI and PETER WONKA
Arizona State University

Generative Fluid Dynamics: Integration of Fast Fluid Dynamics and Genetic Algorithms for wind loading optimization of a free form surface

Angelos Chronis, Alasdair Turner and Martha Tsigkari

University College London, Gower Street,
London, UK, WC1E 6BT

angelos.chronis.09@ucl.ac.uk, a.turner@ucl.ac.uk, mtsigkar@FosterandPartners.com

Keywords: Computational Fluid Dynamics (CFD), Fast Fluid Dynamics (FFD), Genetic Algorithms (GA), wind load, optimization.

Abstract

The integration of simulation environments in generative, performance-driven form-finding methods is expected to enable an exploration into performative solutions of unprecedented complexity in architectural design problems. Computational fluid dynamics simulations have a wide range of applications in architecture, yet they are mainly applied at final design stages for evaluation and validation of design intentions, due to their computational and expertise requirements.

This paper investigates the potential of a fast fluid dynamics simulation scheme in a generative optimization process, through the use of a genetic algorithm, for wind loading optimization of a free form surface. A problem-specific optimization environment has been developed for experimentation. The optimization process has provided valuable insight on both the performance objectives and the representation of the problem. The manipulation of the parametric description of the problem has enabled control over the solution space highlighting the relation of the representation to the performance outcome of the problem.

1. INTRODUCTION

The recent advancements of digital fabrication and the continuous development of computer aided design tools have not only liberated architecture from its geometrical constraints but they have also infused in it a computational realm of experimentation into form and function of unprecedented complexity. With parametric design systems being only the start point and computationally simulated

dynamical systems of form finding the quintessence of this exploration, architecture has in part shifted from the design and drafting to the creation of the generative systems from which the final form emerges (Kolarevic 2003). The architectural discourse has lately been rigorously engaged in embodying the potential of this achievable complexity not only in the aesthetic but also in the performative aspects of contemporary architecture.

The integration of simulation environments in the architectural computational framework has been a key factor in this exploration, as they provide the test bed for experimentation with the various aspects of architectural performance. This has led to a substantial development of both the simulation algorithms and their interfacing tools as well as their applicability in the building industry, facilitated also through the increase of available computational resources but also significantly through novel computational approaches and paradigms (Malkawi 2004). Although it is expected that the incorporation of these simulation environments in computational generative processes will enable a critical and systemic thinking into performance driven architecture and shape innovative performative solutions, their integration in a streamlined computational framework throughout all stages of the design process remains a challenge (Shea et al 2005).

One of the areas that are still largely unexplored, in terms of its integration into generative design approaches is that of computational fluid dynamics (CFD). CFD simulations have a vast variety of applications in architecture, ranging from wind load analysis on buildings to ventilation and energy performance, contamination dispersion or even acoustical analyses (Malkawi 2004). However due to the complexity of the simulated phenomena

and the consequent computational demand of the simulation algorithms available, their use is restricted to final design stages for evaluation and validation of design intentions. There is evidence though, as it will be argued here, that through the use of less accurate but also computationally less demanding approaches, CFD simulations may be able to be applied not only at earlier design stages to inform preliminary design decisions, but even more to be integrated in generative form finding processes. In our approach we aim to explore the potential of a real-time fluid dynamics simulation scheme in the optimization of the wind loading of a free form surface canopy, through the use of a genetic algorithm (GA). It has to be stated, that the aim of this approach is not to attempt to simulate the physical phenomena with the maximum possible accuracy, but rather to investigate the potential of a resource effective simulation scheme in a conceptual stage generative approach.

2. BACKGROUND

The phenomena of fluid dynamics are described by the Navier Stokes equations and are considered one of the most challenging engineering problems. Navier-Stokes are notoriously difficult to approach analytically and therefore numerical methods are used to approximate their solution. This approximation introduces an amount of error in the simulation, the tolerance of which depends on the application and the goals of the simulation (Lomax et al 1999). The computational approaches to the simulation of fluids have been continuously expanding the past decades and there is a great variety of available simulation algorithms today, ranging from particle to mesh based approaches and from lower to higher accuracy models. Despite their continuous development though, the available simulation tools have not yet reached the expectations of researchers in combining accuracy with speed (Chen 2009). The computational timeframe of the simulations ranges from hours and days to even weeks, when involved in optimization processes (Hanna et al 2010).

The areas of application of CFD in architecture are numerous and include among many others the analysis of indoor climate (Hartog et al 2000), ventilation performance (Chen 2009) and heating, ventilation and air conditioning (HVAC) systems (Hien & Mahdavi 1999) as well as the analysis of urban scale conditions, such as street layout configurations (Xiaomin et al 2006) and airflow analysis in dense urban environments (Chung & Malone-Lee 2010). In a number of these studies (Chung & Malone-Lee 2010;

Hartog et al 2000; Hien & Mahdavi 1999) it is highlighted that there is a need to incorporate CFD simulations at earlier design stages. More specifically, what is pointed out is the inconsistency between the synthetic and the analytic design stages and therefore it is suggested that optimization needs to be initiated at early design stages to be more effective. The suggestions include parametric design systems, hybrid simulation approaches and visualization schemes to approach the simulation data sets.

In the field of wind engineering, the applications of CFD simulations are also numerous and span all scales of the urban environment, from high rise buildings (Huang et al 2007) to building roofs (Kumar & Stathopoulos 1997). Although in most cases the study of wind loads is validated through wind tunnel experiments, the CFD simulations are considered a fundamental tool in wind load analysis (Baker 2007). Again the integration of CFD simulations in earlier design stages is attempted through different approaches, such as the development of tools that streamline the pre and post processing of the simulation (Kerklaan and Coenders 2007) or the use of special effect visualization-oriented tools, similarly to the approach taken here, to dynamically sketch and assess the performance of design alternatives of tension membrane structures, before formally assessing them through wind tunnel tests (Mark 2007).

2.1. Fast Fluid Dynamics

The numerical scheme used in our approach was developed for the computer graphics industry (Stam 1999) and therefore the visual aspect but most importantly the speed of the simulation are more crucial than the accuracy of the solver. Although the simulation scheme does not disregard the physics of the fluid mechanics, due to the lower order numerical solvers used to approximate the solution, the solver suffers from an amount of numerical dissipation. Indeed recent experiments with the specific fluid simulation scheme show that the solver cannot accurately predict turbulent flows by comparing the results to standard experimental data for a number of different cases, i.e. flow in a lid-driven cavity, fully developed channel flow, forced convective flow and natural convective flow in a room (Zuo and Chen 2007; 2009; 2010). The same studies however consider the solver's accuracy as adequate for a number of cases and objectives, such as building emergency management, preliminary sustainable design and real-time indoor environment control. The results on the speed of the simulations, on the other hand, are outstanding,

with a timeframe that ranges from 4 to 100 times faster than real time. Moreover, when implemented on a parallel computing scheme the *fast fluid dynamics* (FFD) simulation timeframe are reported to be 500 to 1500 times faster than a CFD simulation. It is not aimed to attempt a validation of these results, here, but on the contrary to explore the potential of the fluid solver in a generative preliminary design optimization process.

2.2. GAs and CFD

Genetic algorithms have been widely used for optimization in many engineering fields and lately they have been increasingly applied in architectural oriented optimization problems. Very briefly, a GA is mimicking the natural selection process through a population of candidates that represent a solution to the given problem by assessing their fitness and reproducing their genetic material. One significant advantage of GAs that makes them inherently useful in architectural context is their effectiveness in searching through very large and complex solution spaces, where deterministic methods cannot perform that well. The integration of CFD simulations and GA optimization techniques is also not a novelty in other engineering areas. In the automotive, aerospace and nautical industries, the use of CFD in optimization techniques is fully integrated and automated (Duvigneau & Visonneau 2003) and falls under the general category of *shape optimization* (Roy et al 2008).

In architectural related research though, there are very few attempts to couple CFD and GAs. These are mainly limited in the analysis of indoor environments and more specifically the optimization of HVAC systems. Malkawi et al (Choudhary & Malkawi 2002; Malkawi et al 2003; Malkawi et al 2005) have combined GAs and CFD to study the performance of a mechanically ventilated room, through the integration of a parametric model and the partial automation of the optimization process, which is also assisted by the user. Although the intervention of the user in the optimization procedure in this case is significant in driving the optimization procedure towards the desired solution space, the use of a commercial fluid solver does not allow for the adaptation of the simulation environment. Other approaches include the use of GAs and CFD for the optimization of HVAC systems in office spaces (Lee 2007; Kim et al 2007) as well as other performance criteria, such as the optimization of window design problems (Suga et al 2010) and smoke-control systems in buildings (Huang et al 2009). Again with a few exceptions these approaches do not

significantly adapt the simulation environment to the needs of each problem or attempt to optimize the computational framework of the simulation. Furthermore most of these studies are considered with orthogonal geometries, thus limiting their applicability in form finding methods.

3. THE OPTIMIZATION FRAMEWORK

As already mentioned, the aim of this approach is to explore the potential of a faster but less accurate CFD simulation scheme in the optimization of the performance, in terms of wind loading of a free form surface. The computational efficiency of the process was overall an important driver in the development of the simulation framework and therefore certain simplifications and compromises had to be made to make the attempt feasible. The free form surface was deliberately vaguely defined, and material and site specific parameters were not considered, in order to reduce the complexity of the problem and shift the interest mostly towards the potential of the solver in providing valuable insight on a simple problem, through a generative design process. Moreover the study was constrained to one constant wind direction to further reduce the problem's complexity, although the use of a range of wind directions is strongly considered as a further step. Finally the wind loading calculation is also restricted to the mean wind load on the surface for simplification, while at a further stage more complex wind loading functions could be taken into account.

The fluid solver used here is based on a three dimensional implementation (Ash 2006) of Stam's (1999) fluid solver. The main simulation routines of the solver are the advection, diffusion and projection routines (see also Stam 2003) and as they form the core of the simulation algorithm they have not been significantly changed. Other parts of the simulation scheme have been adapted and developed to suit the needs of the study. Stam's fluid solver is a grid-based method, i.e. the solution is derived by resolving the equations for each cell in a grid over each time step. The solver is based on a fixed-size grid cell, which does not allow the adaptation of the mesh resolution in areas of interest, but facilitates the integration of other parts of the simulation scheme. The implementation of the fluid solver, as well as all other parts of the optimization framework were developed by the authors in the Processing open-source programming language (Figure 1).

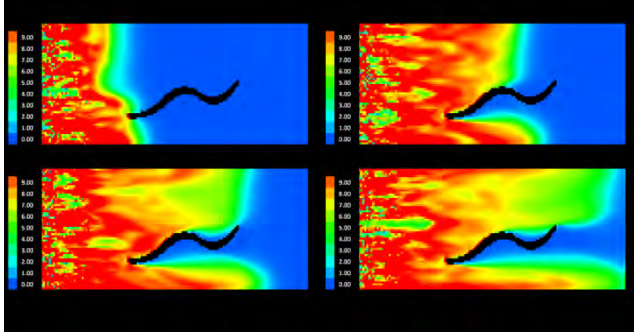


Figure 1 Fluid domain section - velocity field

3.1. Boundaries

One important aspect of the CFD simulations is the handling of the domain boundaries. As the fluid solver was developed for visualization purposes, the external boundary routines of the solver had to be adjusted to simulate a continuous channel flow instead of the original bounding box domain. Again, more complex and sophisticated boundaries that would take into account site parameters, such as the ground surface roughness, would increase the complexity of the problem and therefore were avoided. The modeling of a more complex urban environment was indeed attempted but dropped due to computational resource requirements.

The internal boundaries of the fluid domain are derived by the free form surface and are therefore more complex to handle as they need to be redefined for each instance of the GA population. For this reason a meshing algorithm had to be developed. To describe the internal boundary meshing algorithm a description of the free form surface needs to be given. The surface is defined as a B-Spline (URBS) surface, which is controlled by a set of 25 control points, arranged in a two dimensional grid of 5 control points in each direction. To impose some constructability on the scheme, the 4 corners of the control grid are fixed to the ground, as they could serve as anchor points for the structure. The remaining control points are freely movable by the optimization algorithm inside a certain range of height, width and length, depending on the degrees of freedom which have varied during each optimization scheme, as it will be described further on. The range of movement is also constrained by the adjacency of the points, to avoid folding and distortion of the surface. This configuration defines the parametric range of movement of the surface inside the fluid domain and consequently the range of the solution space (Figure 2).

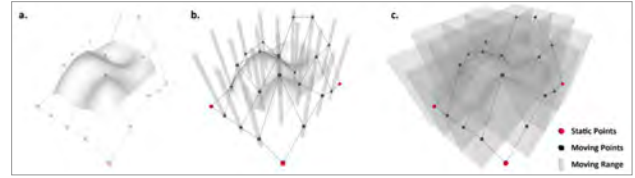


Figure 2 Free form surface configuration (a) and movement range for one (b) and three (c) degrees of freedom.

To set up the internal boundaries of the domain, for each parametric instance of the surface, a meshing algorithm had to be developed. For internal boundaries, at least two cells of the fluid domain in each dimension have to be occupied to counteract the velocity and density fields of the fluid in both directions normal to the surface. The meshing algorithm runs for every (u, v) position on the surface (at given intervals) and assigns a solid boolean for the corresponding pair of cells of the fluid domain in each dimension. This boolean is used inside the boundary routine of the solver to adjust the condition at the boundary cells at each time step of the simulation. The boundary cells are also used for the calculation of the wind loading of the surface which is implicitly calculated from the density and velocity fields (Figures 3, 4).

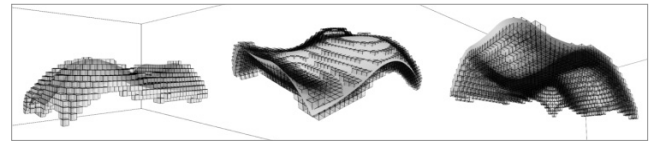


Figure 3 Internal boundaries generated by the meshing algorithm for one instance of the surface.

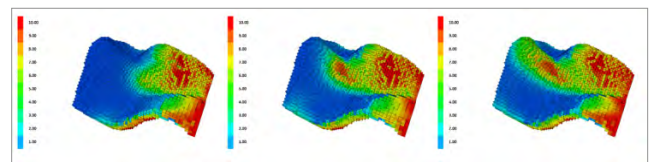


Figure 4 Wind loading calculation – three time steps.

3.2. The genetic algorithm

Each member of the GA population represents an instance of the parametric surface and corresponds to a set of genes which define its parameters. Three different encoding schemes for the GA genes were used in our experiments, each with different results. Two of these schemes were a direct encoding of the moving position of the control points of the surface, movable in one (z component only) and three dimensions (x,y,z) respectively.

To reduce the searchable solution space and increase the convergence of the optimization process, a topological description of the surface was also introduced as an encoding scheme. In this scheme, the control points of the surface were derived from a sine function, whose parameters were encoded in the genes in order to preserve the topological relation between the control points (Figure 5).

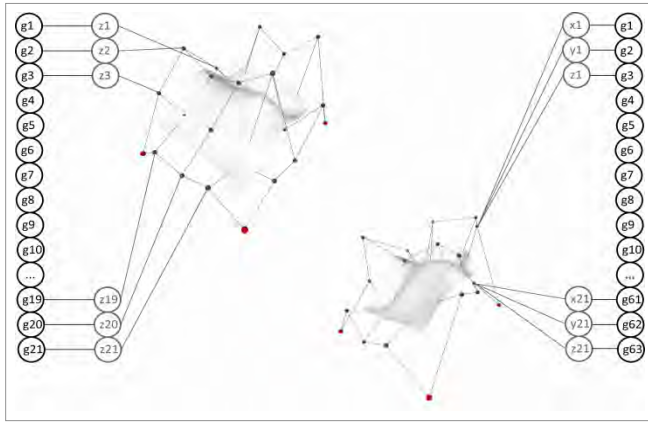


Figure 5 The genotype and phenotype correspondence in two encoding schemes.

In the scope of this effort two different gene crossover methods were also used, i.e. a multiple point and a single point crossover method which is also helpful in preserving the topology of the surface. In the multiple point crossover method each new member of the population inherits the genes from each of its parents in random order, therefore loosing the topological relationship between adjacent control points of the surface. In the single point crossover method, on the other hand, each new member inherits a uniform portion of its parents' chromosomes, preserving the topological relationship between adjacent points.

The population size of the GA in our experiments ranged from 36 to 64 members and the number of offspring did not exceed the 1500 generations, due to computation time limitations. As it has already been mentioned, the time efficiency of the process was an important factor in the development process therefore the optimization process did not reach its full potential in many of the experiments as it was restricted by the timeframe which has reached up to 8 hours on a typical configurations computer. One thing worth mentioning is that although the GA is able to generate multiple instances of the surface in no time, each of them has to be evaluated individually, in a serial manner, through

the CFD simulation engine, as the generation of multiple CFD domains to parallelize the optimization process would exceed the capabilities of any machine. However a further step could involve the distribution of the optimization process over a network.

The steps of the optimization process were the following:

- 1) An initial population of surfaces is randomly generated by the GA.
- 2) For each member of the initial population the CFD domain is preprocessed and the internal boundaries are defined.
- 3) The simulation engine runs until it converges and the fitness of the member is evaluated as the mean wind load on the surface in a given time period.
- 4) The fluid domain and the internal boundaries are reset.
- 5) When the whole of the initial population has been evaluated and ranked, a new member is generated by the GA and steps 2 to 4 are repeated for each new member (Figure 6).

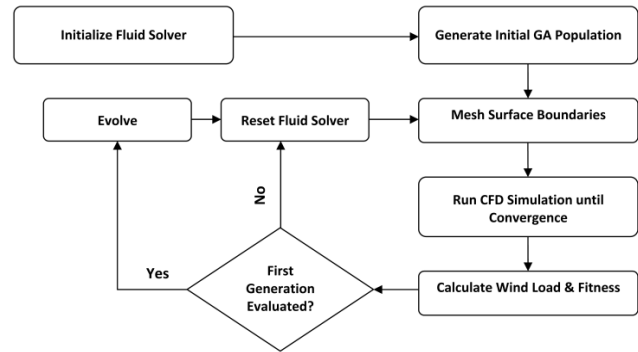
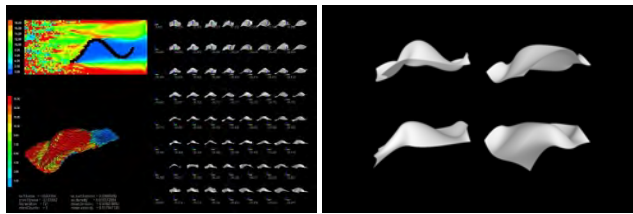


Figure 6 The optimization flowchart.

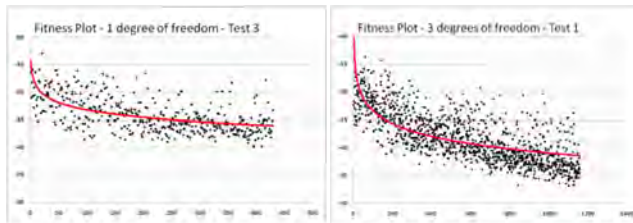
4. RESULTS

The results of the experiments with the first two (direct) encoding schemes were, as expected, similar, with a slight increase in the performance range achieved by the three degrees of freedom scheme. In both cases the optimization process was successful in optimizing the performance of the surface, steadily decreasing the wind loads, though not reaching its full potential, in the given timeframe. The optimum range of generated surfaces in both schemes presented significant uniformity in their form, characterized

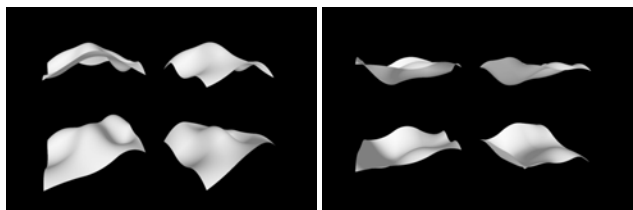
by symmetry in the vertical to the wind direction as well as by a wave-like profile in the parallel direction leading to an overall aerodynamic shape. In both schemes however the optimum forms did not converge to a single trend, but were divided into two schemes, almost inverted from each other in form but both performing equally in the CFD domain. This lack of uniformity could be indicative of a possible preliminary convergence to local optima, also due to the computational demands of the process, or possibly of an under-constrained problem set up which leads to multiple solutions. Therefore it led to further experimentation in an effort to constrain the solution space through both a different crossover method and a different encoding scheme (Figures 7-12).



Figures 7, 8 Three degrees of freedom - Optimization screenshot and 4 fittest members



Figures 9, 10 Fitness plots for one and three degrees of freedom.

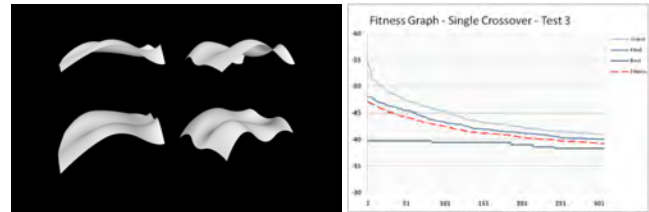


Figures 11, 12 One degree of freedom - 4 fittest members of the two optimization trends

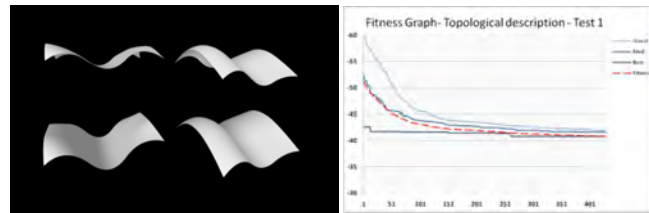
The single crossover method introduced was effective in reducing the time frame of the optimization, achieving a similar performance range in almost half the amount of generations, without though effectively converging to one uniform solution. The shape of the optimal surfaces followed again two distinct trends with two inverted shape

patterns similar to the ones observed with the multiple crossover method (Figures 13, 14).

On the contrary, through the topological description scheme, the optimization process converged to one single optimal trend throughout all the experiments, without though reaching the same performance range, indicating that the preservation of the surface topology was effective in the restriction of the solution space but also in the restriction of the achievable performance (Figures 15, 16).



Figures 13, 14 Single crossover method - 4 fittest members and fitness graph



Figures 15, 16 Topological description scheme- 4 fittest members and fitness graph

5. DISCUSSION

The incorporation of the FFD solver in the optimization process has been effective in providing an experimentation platform for generative techniques. Although the computational demand of the process remains high, it has been feasible not only to integrate the complex CFD simulations in a generative approach but furthermore to generate results, with regards to the objectives of this study, in a complex solution space. More importantly the implementation of the fluid solver in an object-oriented programming environment has enabled the adaptation of the simulation environment to the study's objectives.

In terms of the problem-specific generated results, the optimization process has provided valuable insight on the problem, through the identification of performance trends, in relation to the different encoding schemes. The manipulation of the problem's representation has provided control over the generated solution space, highlighting the

relationship between the parametric definition range and the resulting performance range. This control though can by no means be regarded as a way to a deterministic solution to the studied problem, if not properly assessed in relation to more realistic design parameters. In the scope of this study, several inseparable parameters of what would constitute a realistic architectural problem are deliberately omitted, for simplification. Nevertheless the focus of the study was on the potential of the implementation of the solver in a preliminary design stage and thus it can be regarded as a starting point for further experimentation.

It needs to be mentioned that for these results to be useful one essential further step is the formal assessment of their performance using standard simulation packages, or potentially physical tests, as the inaccuracy caused by the numerical dissipation of the solver could possibly be misleading. Moreover, the use of generic parameters, in both the simulation framework and the description of the problem, does not facilitate the assessment of the results in an architectural context. A more thorough modeling of the simulation environment as well as the incorporation of other important optimization constraints, such as pedestrian comfort or other types of loading, are important further steps as well. Further work in progress also includes the use of multi-objective GAs to incorporate other optimization constraints in the problem, the use of artificial neural networks, trained by the solution space to increase the time efficiency and achieve a two-step optimization process as well as the implementation of further developed versions of the fluid solver to produce more accurate results.

6. CONCLUSION

In this study we explored the potential of a fast fluid dynamics simulation scheme in a generative optimization process. The implementation of the fluid solver in a problem-specific optimization framework has been effective in providing informative insight on the representation of the problem. Although more complex representations and objectives as well as a formal validation of the generated results would be required to address real architectural problems, this implementation is considered as a proof of concept and further experimentation with the simulation framework is expected to yield more interesting results.

References

- ASH, M. 2006. 'Fluid Simulation for Dummies', [online] Available at: <http://mikeash.com/pyblog/fluid-simulation-for-dummies.html> [Accessed 01 September 2010].
- BAKER, C. 2007. 'Wind engineering-Past, present and future', in *Journal of Wind Engineering and Industrial Aerodynamics* 95(9-11), 843-870.
- CHEN, Q. 2009. 'Ventilation performance prediction for buildings: A method overview and recent applications', in *Building and Environment* 44(4), 848-858.
- CHOUDHARY, R., MALKAWI, A. M., 2002. 'Integration of CFD and genetic algorithms', in *Proceedings of the Eighth International Conference on Air Distribution in Rooms*, Copenhagen, Denmark.
- CHUNG, D. H. J. AND L. C. MALONE-LEE 2010. 'Computational Fluid Dynamics for Urban Design', in *Proceedings of the 15th International Conference on Computer-Aided Architectural Design Research in Asia CAADRIA*, April 2010, Hong Kong, China, 357-366.
- DUVIGNEAU, R. & VISONNEAU, M. 2003. 'Shape optimization strategies for complex applications in Computational Fluid Dynamics', in *Proceedings 2nd International Conference on Computer Applications and Information Technology in the Maritime Industries*, May 2003, Hamburg, Germany.
- HANNA, S., HESSELGREN, L., GONZALEZ, V. & VARGAS, I. 2010. 'Beyond Simulation: Designing for Uncertainty and Robust Solutions', in *Proceedings of the Symposium on Simulation for Architecture and Urban Design at the 2010 Spring Simulation Multiconference*, April 2010, Orlando, USA.
- HARTOG, J. P. D., KOUTAMANIS, A. AND LUSCUERE, P. G. 2000. 'Possibilities and limitations of CFD simulation for indoor climate analysis', available from: Repository of Delft University of Technology.
- HIEN, W. N. & MAHDAVI, A. 1999. 'Computational Air Flow Modeling for Integrative Building Design', in *Proceedings of the 6th International Building Performance Simulation Association Conference - 'Building Simulation 1999'*, September 1999, Kyoto, Japan.
- HUANG, S., LI, Q. & XU, S. 2007. 'Numerical evaluation of wind effects on a tall steel building by CFD', in *Journal of Constructional Steel Research* 63(5), 612-627.
- KERKLAAN, R. A. G. & COENDERS, J. 2007. 'Geometrical modeling strategies for wind engineering', in *Proceedings of the IASS Symposium on Shell and Spatial Structures - 'Structural Architecture: Towards the future, looking to the past'*, Venice 2007, Italy, 81-82.
- KIM, T.; SONG, D.; KATO, S. & MURAKAMI, S. 2007. 'Two-step optimal design method using genetic algorithms and CFD-coupled simulation for indoor thermal environments', in *Applied Thermal Engineering* 27(1), 3-11.
- KOLAREVIC, B. 2003. 'Computing the Performative in Architecture', in *Proceedings of the 21st eCAADe Conference - 'Digital Design'*, Graz 2003, Austria.

- KUMAR, K. S. & STATHOPOULOS, T. 1997. 'Computer simulation of fluctuating wind pressures on low building roofs', *Journal of Wind Engineering and Industrial Aerodynamics* 69-71, 485-495.
- LEE, J. H. 2007. 'Optimization of indoor climate conditioning with passive and active methods using GA and CFD', in *Building and Environment* 42(9), 3333-3340.
- LOMAX, H., PULLIAM, T. H., ZINGG, D. W., PULLIAM, T. H. & ZINGG, D. W. 2001. *Fundamentals of computational fluid dynamics*, Springer Berlin.
- MALKAWI, A. M. 2004. 'Developments in environmental performance simulation', *Automation in Construction* 13(4), 437-445.
- MALKAWI, A. M.; SRINIVASAN, R. S.; YI, Y. K. & CHOUDHARY, R. 2003. 'Performance-based design evolution: the use of Genetic Algorithms and CFD', in *Proceedings of the 8th international IBPSA conference – Building Simulation 2003*, August 2003, Eindhoven, Netherlands.
- MARK, E. 2007. 'Simulating dynamic forces in design with special effects tools', in *Proceedings of the 25th eCAADe Conference – 'Predicting the Future'*, Frankfurt 2007, Germany, 219-226.
- ROY, R., HINDUJA, S. & TETI, R. 2008. 'Recent advances in engineering design optimisation: Challenges and future trends', *CIRP Annals - Manufacturing Technology* 57(2), 697-715.
- SHEA, K., AISH, R. & GOURTOVAIA, M. 2005. 'Towards integrated performance-driven generative design tools', *Automation in Construction* 14(2), 253-264.
- STAM, J. 2003. 'Real-time fluid dynamics for games', in *Proceedings of the Game Developer Conference*, March 2003.
- STAM, J. 1999. 'Stable fluids', in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, August 1999, Los Angeles, USA, 121-128.
- SUGA, K.; KATO, S. & HIYAMA, K. 2010. 'Structural analysis of Pareto-optimal solution sets for multi-objective optimization: An application to outer window design problems using Multiple Objective Genetic Algorithms', in *Building and Environment* 45(5), 1144-1152.
- XIAOMIN, X., ZHEN, H. & JIASONG, W. 2006. 'The impact of urban street layout on local atmospheric environment', *Building and Environment* 41(10), 1352-1363.
- ZUO, W. & CHEN, Q. 2010. 'Fast and informative flow simulations in a building by using fast fluid dynamics model on graphics processing unit', *Building and Environment* 45(3), 747-757.
- ZUO, W. & CHEN, Q. 2009. 'Real-time or faster-than-real-time simulation of airflow in buildings', *Indoor Air* 19(1), 33-44.
- ZUO, W. & CHEN, Q. Y. 2007. 'Validation of fast fluid dynamics for room airflow', *Proceedings of the 10th international IBPSA conference – Building Simulation 2007*, September 2007, Beijing, China.

Use of Sub-division Surfaces in Architectural Form-Finding and Procedural Modelling

Shajay Bhooshan, Mostafa El Sayed

Zaha Hadid Architects, Computation and Design Group
10 Bowling Green Lane
London EC1R 0BQ, United Kingdom
shajay.bhooshan@zaha-hadid.com

Keywords: Physically based simulations, form finding, procedural architectural modelling, and subdivision surfaces.

Abstract

Growth in the role of simulation, in conceptual design and evaluation of building performance, is coupled with the need for architects to manage the dichotomy between high resolution geometries used in simulation and lower resolution, free-form manipulation-friendly geometries. We argue for the benefits of Hierarchical Subdivision-surfaces (HSS) in this process, concentrating specifically on its uses in several areas. First, we discuss HSS for the design and approximate simulation of Shell structures using particle-spring methods and its correlation with FEA analysis. Second, we show the benefits of HSS for the procedural modeling of large numbers of such shells. Third, the pre-engineering rationalisation of such designs is explained and, lastly, the preparation of such geometries for downstream production, including fabrication and construction, is discussed. The paper will present case studies for each area, the design of architectural competition entries within the areas, and mock-ups of structures designed and fabricated using these methods.

1. INTRODUCTION

This paper stems from a preceding short-term research project that posited itself within the argument for *parametric design research* to focus on design methods that enable an operative pathway from design intent to its manifestation. The objective was to investigate the use of physically-based parameters as effective constraints in conceptual design. Integral belief to this was that conceptual design workflows should privilege real-time interaction between the designer and the computational model.

Physical *form-finding* using *hanging chains* and associated architectural design methods as pioneered by Antonio Gaudi and their subsequent exposition by Heinz Isler, Frei Otto, and others are very well documented and is common knowledge among architects. Their digital simulation using particle-spring simulation frameworks are also fairly established following the work of early (architectural) exponents including Axel Killian (Killian and Ochsendorf, 2005) and renewed interest amongst various academics in recent times. Common algorithms of digital form-finding use a particle-spring framework to simulate *hanging chains* and *hanging grids*, and produce results similar to Dynamic Relaxation (Fraschini, Prete and Laurino 2009). The research aimed to extend this work to adequately represent the complexities of scale, time-constraints and delivery mechanisms of contemporary architectural practises.

This paper presents a proof of concept for enabling an efficient workflow between conceptual design and downstream delivery within an architectural production pipeline. Simulation accuracy, production grade geometry etc. are outside the scope of this paper. As such, the paper will focus on its essential contribution: a simulation-based architectural design workflow, adopted from the computer graphics (CG) industry, incorporating the use of hierarchical subdivision surfaces as a means to provide intuitive control for the designer whilst still maintaining an awareness of down-stream architectural production processes. The argument will be confined to the design and simulation of *Shell* structures.

2. WORKFLOW OVERVIEW & TERMINOLOGY

A typical architectural design workflow involves the following steps:

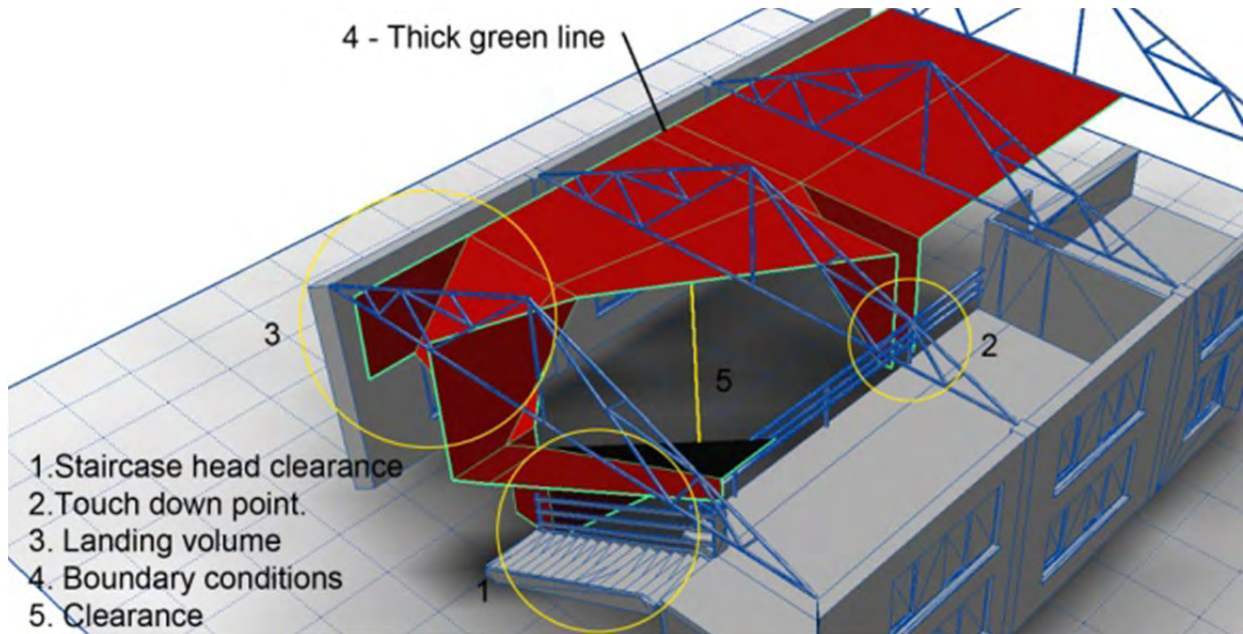


Figure1. Low-poly mesh showing design-intentions.

2.1. Setup / Initial Conditions

The designer usually models a predominantly quad-faced, low-resolution mesh (*low-poly mesh*). This embeds essential topological features, such as holes and the placement of (curvature) singularity points (see Figure 2). Key parameters of the design intent are also specified—clearances, touch-down points, boundary connections, etc. (see Figure 1).

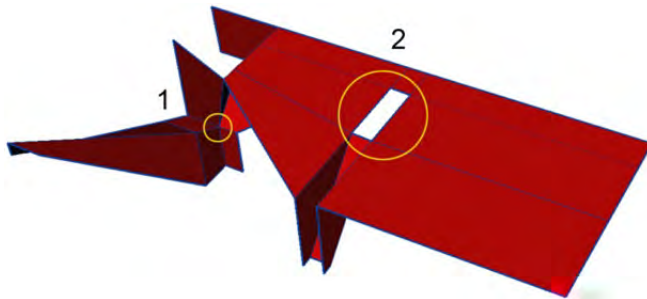


Figure 2. Low-poly mesh showing topological intentions; (1) point of (curvature) singularity, (2) hole.

2.2. Storing Information

All faces and their corresponding edges are stored in a set: *face-edges set*.

2.3. Sub-division

Subsequently the low-poly mesh is converted into a higher resolution mesh (*high-poly mesh*) using a subdivision (*subD*) algorithm (see Figure 3). We used Autodesk Maya smooth-mesh®, a Maya specific implementation of the

Catmull-Clark sub-division surface. This mesh is treated as a ‘cloth’ object and simulated subject to external forces – usually gravity– and internal spring forces, until an equilibrium condition is reached.

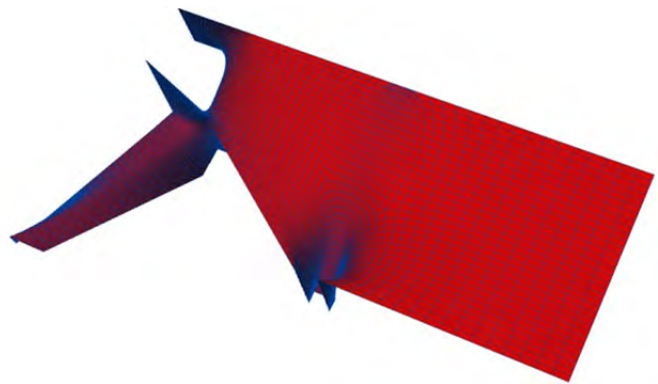


Figure 3. High-poly mesh.

2.4. Tracking

During the process of subdivision, the original face-edges set is also sub-divided and as such now contains additional edges. These edges are easily tracked on the high-poly mesh due to the exponential and algebraic relation of their numeric IDs to those in the original set (see Figure 4 bottom). Further, since these edges originally described closed (face) outlines, they can now be used to describe boundary NURBS patches (see Figure 4 top). The low-poly mesh itself proves useful in various down-stream applications as will be delineated in the next sections.

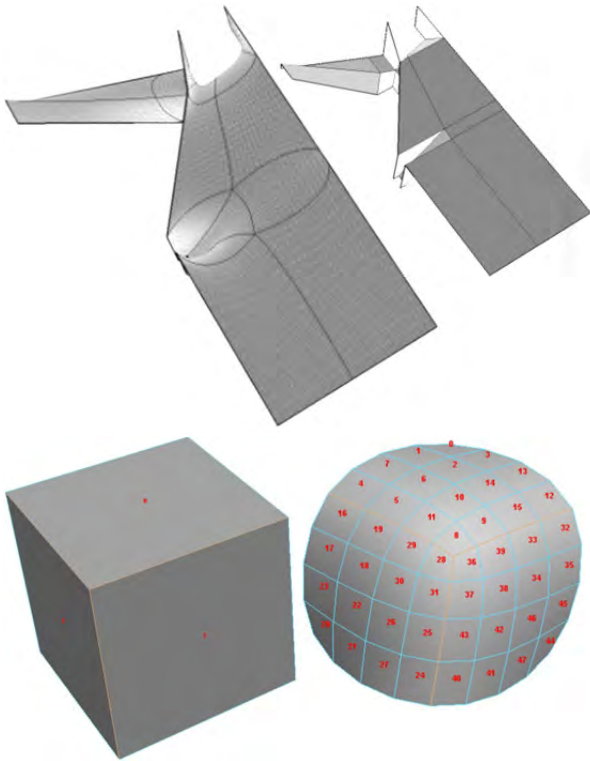


Figure 4. (top) Tracking low-poly mesh face-edge sets on high-poly mesh. (bottom) Exponential relation between numeric IDS of high-poly and low-poly meshes

2.5. Export

The above NURBS patches are subsequently exported to downstream applications and processes.

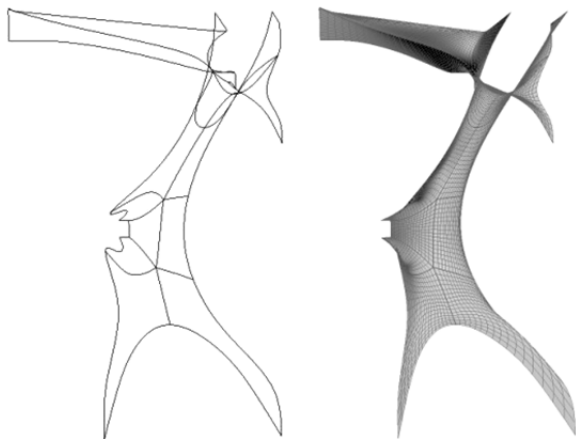


Figure 5. Using boundary curves to describe NURBS patches.

The paper will now proceed to discuss the following significant issues stemming from the argument above:

- Design and approximate simulation of Shell structures.
- Procedural modelling of large number of such shells.

- Pre-engineering rationalisation of such designs.
- Preparation for production, fabrication, construction.

3. APPROXIMATE SIMULATION & DESIGN OF SHELL STRUCTURES

Significant issues that affect the workflow of a designer within the form-finding process are (a) the current design-to-physical manifestation discourse mentioned previously which usually entails that design intent itself is evolved through an *iterative and negotiated* interaction between the designer, a simulation engine, and design/physical constraints, and (b) the complex nature of simulation algorithms as also the black-box nature of the simulation tools themselves. Consequently, the design intent is established either as *pre-simulation* control of initial and boundary conditions, and *post-simulation* addition of features and elements in relation to the simulated surface. There is minimal interference *during-simulation*. Of interest to the current argument, however, is the usefulness of our *workflow* in each of the phases, as will be delineated below.

3.1. Pre-simulation

Pre-simulation is an integral part of the evolution of design intent and must support iterative manipulation. Common digital form-finding methods usually require a mesh – *simulation mesh*, especially in simulating hanging-grids. Subsequently, vertices of the mesh are treated as point-mass objects and the edges treated as springs, and the entire configuration is iteratively advanced until an equilibrium configuration is reached. One method to generate a reasonably dense simulation mesh, from a user-input set of boundary curve(s), is to use Constrained Delaunay Triangulation (CDT). Alternatively, our workflow uses a subD algorithm to subdivide the low-poly mesh to generate this simulation mesh, offering the following advantages to the designer.

First, the iterative design studies mentioned above become intuitive and easy to due to the malleable nature of the low-poly mesh. These iterative studies are integral part of evolution of design intent whilst working with simulation tools. As such, ease, intuition and speed of manipulation are key. Eventually the low-poly mesh will embed key features of design intent including clearances, touchdown points, boundary conditions, etc.

Second, a more intuitive control over mesh topology including placement of (curvature) singularity points, holes,

etc. is implicitly available. Having control over the placement of singularity points and holes, enables the designer to place these with regard to the overall tessellation of the initial surface. This correlation, apart from ensuring good valence for most vertices, proves useful in other downstream processes as well, as will be explained later.

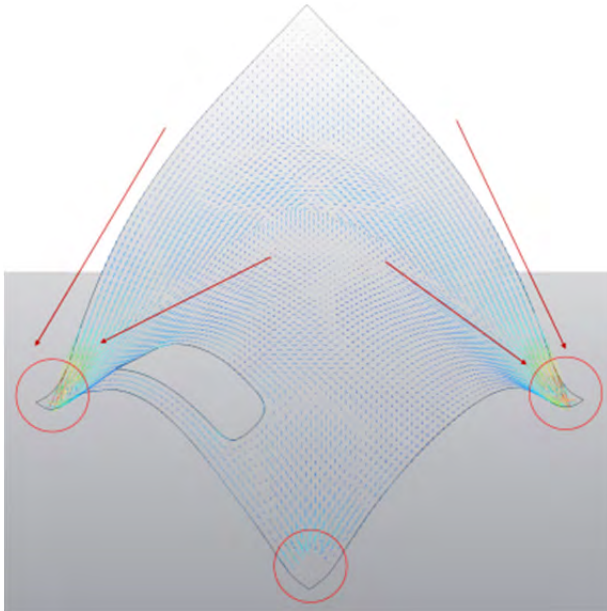


Figure 6. Principal stress directions (red arrows) from FEA analysis in Autodesk ALGOR. Red circles show supported regions.

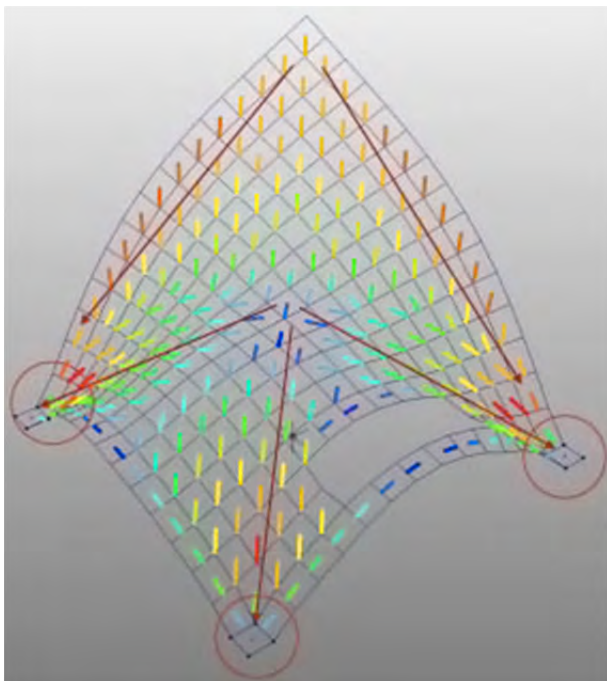


Figure 7. Approximate principal force directions from Custom Maya plug-in implementing the ‘rain-flow’ analogy.

3.2. During-simulation

The during-simulation stage of the workflow revolves around the user’s qualitative understanding of the form and its structural performance. Further to the design-to-production discourse mentioned previously, the formation of a qualitative and intuitive understanding of complex topologies and their structural behaviour has gained interest and importance. However it remains elusive to the designer despite of physics-based nature of form-finding algorithms.

Previously, graphical analytical systems such as the traditional 2D line-of-thrust and the recent Thrust Network Analysis method (Block 2007) provide such insight into the relations between geometric properties of surfaces – curvature, thickness etc, and their structural properties – load paths, principal force directions, types of forces etc.

In keeping with the spirit of these ‘graphical’ systems we attempted two visualisations to be used during-simulation: in-simulation stress/strain visualizations and approximation of principal stress directions (see Figure 6) using the rain-flow analogy (see Figure 7): “loads will flow along curves with the steepest ascent on the shell surface to its supports.” (Borgart 2005).

The visualisations were intended to be correlated with post-simulation finite element analysis (FEA). Whilst we continue these empirical tests to aid our understanding of geometry and structure, of interest to the current argument is the usefulness of subD algorithms in the process. Given that real-time feedback was of the essence and that both visualisations are reasonably expensive to compute, we used the ‘smoothing’ and interpolative properties of subD algorithms to selectively calculate required values at limited ‘sample’ points and interpolate to the rest of the points on the simulation mesh. The localised nature of the algorithm and its time-tested robustness ensured reasonable approximation whilst enabling real-time feed-back.

3.3. Post-simulation

3.3.1. Correlation with simple FEA

As mentioned previously, we intended to correlate during-simulation visualization with post simulation FEA of the simulated surface under self weight and uniform loading. This process usually involves exporting the geometry to an FEA package which subsequently re-meshes the geometry as per FEA requirements. Alternatively we

used the subdivided high-poly mesh and found that it did not produce a significantly qualitative difference, for our purposes. Further, the exponential and simple algebraic relationship between the low-poly mesh components (vertices, edges, faces) and those of the exported high-poly mesh meant that the import and correlation of FEA values was straight-forward.

3.3.2. Procedural geometry generation on simulated surface as host.

Further, quad-faced topology represents a good choice for a range of tile-based procedural generation of geometry on the simulated host surface, due to the flexibility it offers on converting to triface, hexagonal-faced and other types of polygons. The quad-faced typology also exhibits inherent directionality which is a desirable aesthetic quality.

3.3.3. Manual addition of features on simulated surface as host.

The ‘exponential relation’ property, predominantly quad-faced topology and uniform valence etc. are all consistent with prescribed ‘poly-modelling’ workflows and as such post-simulation, manual addition of features are easy and intuitive to the designer.

3.3.4. Export to down-stream applications.

As already explained in the workflow description, the ‘exponential relation’ property is also helpful in generating boundary curve-based NURBS patches for export to other downstream applications for further architectural production.



Figure 8. Propagated components model.

4. PROCEDURAL MODELLING OF DETAILED GEOMETRY

The advantages of working through low poly models are greatly amplified when the propagation of these models as repeated elements, such as the domes in Figure 8, is a requirement of the design process. See Figure 13 for additional examples.

4.1. Custom Syntax

To start we utilize a custom built syntax revolving around the users input of simple string instructions for the creation of a base set of points, this string command incorporates four simple instructions consisting of where to originate in the world space and what method of transformation to apply. This deployment system is abstracted into string segments wherein the position of a new point is defined as:

$v1 | v2 | tM | a$, where $v1$ and $v2$ represent pre-existing input points and tM is the transformation method which is denoted by Z for movement of $v1$ along the Upvector and V for movement between a vector defined by point $v1$ and $v2$ and N which is movement along the vector defined by $v1$ s normal.

These string commands generate a new set of points which are then used to generate geometry through simple connection commands. The syntax is designed to enhance the ease of its use and portability in the sense that all that is now required to construct the components is a few simple lines of the string commands.

4.2. Color Sculpting

The modification of large numbers of geometric elements usually requires designers to interactively adjust a large number of points on a control net to achieve their desired shapes, which can often be a laborious process. Deforming the geometry may also be necessary if the process is not carried out meticulously. On the other hand the use of a color sculpting system presents an intuitive and flexible alternative. The method relies on a ‘color mesh’ that is colored and the propagated geometry responds according to the color value of its assigned or closest vertex on the color mesh. This results in a system where designers do not need to concern themselves with the point-by-point modification of a control net but rather the modification of color swatches and blending between them. This is easily carried out in the majority of available design platforms.

This also lets the designers decide what transformations are applied to the geometry as a result of the color modification. For example, certain color values could control the z axis of the geometry while others control rotation of the geometry, allowing the designer to have either one-to-one control of the deformation or a weighted process of transformations being applied due to the mixing of color values.

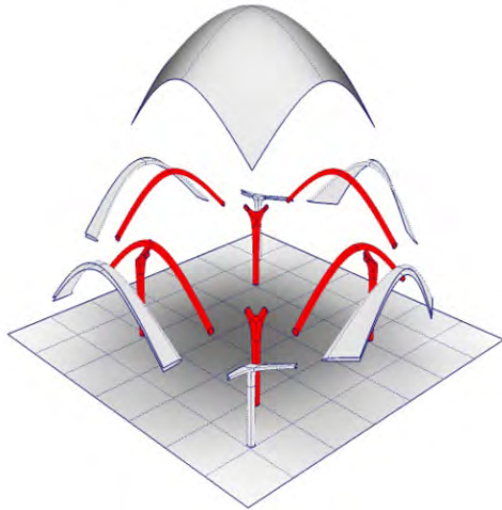


Figure 9. Main components vs. subcomponents and their proportional integrity.

4.3. Proportional Integrity

This use of custom syntaxes and color sculpting maintains the interactivity of a control net while preserving the proportional system built into the components, leading to a cohesive rational transition between them. Also, since each individual component consists of various other components itself, the component's parts also retain their proportional integrity. This level of control is helpful when coupled with the designer's ability to decide which subcomponents incorporate the deformations and which remain unchanged (see Figure 9). This creates an extremely flexible system of control and expression within a rational and easily discretized framework.

5. PRE-ENGINEERING RATIONALISATION OF SHELL STRUCTURES

At architectural scale, complex, continuous and free form topologies need to be physically manifest as an assembly of simple discrete elements. As such, it is desirable to form a rough estimate of the number of types of elements and their quantities at an early stage of a project (see Figure 10).

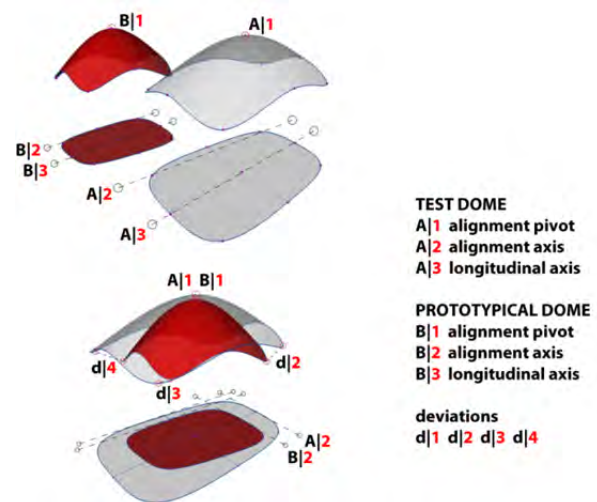


Figure 10. Simple evaluative metric used to estimate number of types of domes.

Furthermore, currently there is a great amount of effort spent in post-rationalising free form geometries to reduce the number of types and increase the quantities of repeating elements to ensure reasonable cost of construction. Common methods of rationalising include reducing complex high-resolution geometries to their lower resolution control nets (Ohmori et al. 2009, Veltkamp 2009). Subsequently a suitable mechanism is chosen (e.g. iteration, Genetic Algorithm, etc.) to search through various configurations of the control net for an optimal solution. Additionally, it is desirable to identify points of (curvature) singularity in some of the optimization procedures. None of these are trivial and are tasks often outsourced to specialists. In the workflow presented here, the control-net already exists as the lowpoly mesh and, more importantly, already embodies the design-intent of the designer, including the placement of singularity points. Simple evaluative metrics can also be established in early conceptual design to form a rough estimation of quantities (see Figure 11).

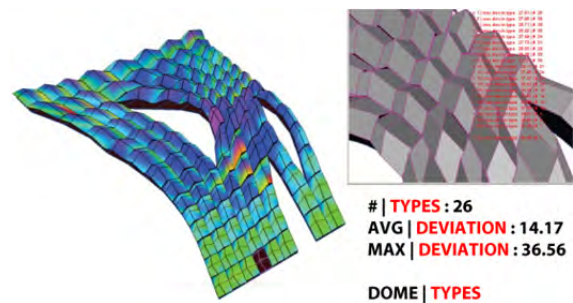


Figure 11. Simple evaluative metric used to estimate number of types of domes.

6. FABRICATION AND SIMULATED MODELS

The low poly methodology of design affords the designer an intuitive workflow to integrate fabrication logics into a cohesive design process from the conception stages onwards. This is due to the inherent and intuitive readability of low poly meshes which allow designers to begin to integrate things such as seams and connection edges as well as allowing the designer(s) to keep track of these edges throughout the evolution of the design and geometry.

6.1. Case Study Introduction

In the interest of clarity this section will use as a case study the workflow involved in the execution of a design-build exercise carried out over the summer of 2010. The workshop proceedings revolved around the design and construction of an interior installation within a two storey warehouse acting as the main architectural studio within the school, the warehouse volume was used to deploy a space filling diagram in the form of a low poly base mesh which was later used in a particle spring simulation to create a light and transparent tensile structure within the warehouse.

6.2. Inherited Quad Division

A key quality inherited from the base low poly mesh is its underlying quad division that can be remapped onto the higher resolution mesh required for particle spring simulations or a counterpart triangulated mesh created due to the triangulation requirement of various physical output methods such as stereolithography. The key difference between retaining the initial edge set is that they do not operate as a control net as is the case in various subdivision surface models nor are they mapped using projection onto a host surface, instead due to the recursive refinement structure of subdivision surfaces, the newer higher resolution edge set is used to recreate the initial set so that no new edges are introduced and that they all lie on the surface itself unlike in control net situations. Also, due to the “topological generality” (Xie and Qin 2002) of such a methodology the majority of the surfaces vertices are of equal valence which is ideal for the particle spring simulation, demonstrated by the surface employed in the China case study wherein there was an aforementioned single singularity point with a higher valence than the rest of the vertex set.

6.3. Seam Generation and Segmentation

Secondly, the initial edges of the base polygon could be stored and later used as seam lines on the final relaxed, high

poly mesh, which allows for two different workflows. On the one hand, the edges could be used as constraints in the simulation. This initial edge set also functions well for the placement of spring constraints on the surface, with a higher strength value than the other constraints. This methodology facilitates the construction of the entire surface as a series of patches. On the other hand, the simulated mesh could be split along these edges allowing for quad patches that are easily tracked from the low poly stage of design to the post simulation stage.

6.3.1. Cross Patch Continuity

Using the low poly edges as a source of spring constraints is a methodology that has clear implications on the surface’s cross patch continuity. Clearly this method retains positional continuity (G^0) but previous tangential (G^1) and acceleration (G^3) continuity created by the simulation is no longer possible due to the edge constraints acting as breaks in the surface’s curvature rate of change resulting in a surface with a C^0 value of continuity. Although this usually is a negative result since it is usually considered “important to achieve visual smoothness over the entire model while manufacturing physical forms” (Daniels II, J 2005); for our purposes this actually plays into our advantage since there now exists an implicit relationship between the fabrication and surface generation methodologies. This is due to the fact that the surface was constructed as a series of patches that were later stitched together and once tension was applied similar continuity was exhibited in the full scale physical model as that of its simulated, digital counterpart.

6.3.2. Seam Stitching

This behavioural similarity is due to both the materials and methods employed in the physical model being informed by the factors integrated into the digital simulation. In order to replicate the effects of the spring constraints placed using the low poly edges a thicker, less elastic gauge of threaded metal cable was used as patch boundaries while in the inner edges were described using thinner, more elastic cable with the inner lattice being scaled down to pull the entire patch into tension once the inner grid was attached to the boundary edges. This lead to several tension systems within the completed model, the boundary curves where placed in tension by both their respective inner grids (local tension) as well as being stretched into position in the space (global tension). See Figure 12.

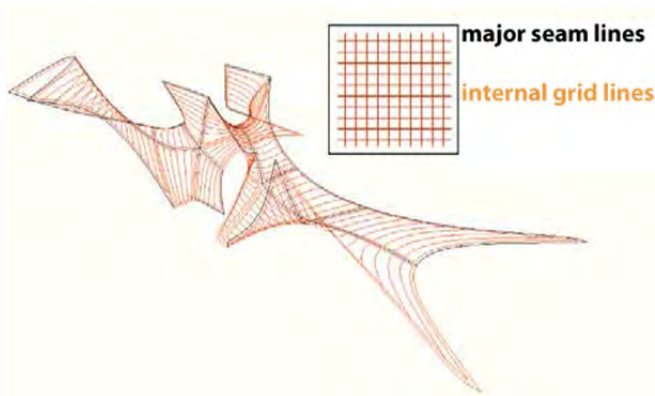


Figure 12. Global vs local tension.

7. CONCLUSION

The paper presented and argued for the benefits of using a sub-division surface based work-flow to aid the process of architectural form-finding. As mentioned previously, the paper attempted to present a proof of concept. Future scope of the research includes development of custom-tools to procedurally develop the low-poly mesh in the presimulation stage. Currently, this stage is manually carried out by the designer. Subsequently, the insimulation visualisations can also be modified to better correlate with FEA analysis. Lastly, postsimulation workflow can also improved by computing the placement and properties of such 'building-aware' elements as structural ribs, apertures ornament etc.

References

BLOCK, P., OCHSENDORF, J. 2007. THRUST NETWORK ANALYSIS: A NEW METHODOLOGY FOR THREE-DIMENSIONAL EQUILIBRIUM. JOURNAL OF THE INTERNATIONAL ASSOCIATION FOR SHELL & SPATIAL STRUCTURES 48:3 P. 167-173.

BORGART, A., 2005. THE RELATIONSHIP OF FORM AND FORCE IN (IRREGULAR) CURVED SURFACES. PROCEEDINGS OF THE 5TH

INTERNATIONAL CONFERENCE ON COMPUTATION OF SHELL AND SPATIAL STRUCTURES.

CHENG, K., WANG, W., QIN, H., WONG, K., YANG, H., AND LIU, Y, 2004 FITTING SUBDIVISION SURFACES TO UNORGANIZED POINT DATA USING SDM. 12TH PACIFIC CONFERENCE ON COMPUTER GRAPHICS AND APPLICATIONS '04 PROCEEDINGS. P.16- 24.

DANIELS II, J, 2005 VISUALLY SMOOTH BLENDING SURFACES FOR FABRICATION OF PHYSICAL MODELS. UNIVERSITY OF UTAH, SCHOOL OF COMPUTING.

FRASCHINI, M., LO PRETE, M., LAURINO, N., 2010. ADVANCED ARCHITECTURAL WORKSHOP: IMPLEMENTING STRUCTURAL LOGICS INTO A DESIGN PROCESS FOR LARGE SCALE LIGHTWEIGHT COVERINGS. JOURNAL OF THE INTERNATIONAL ASSOCIATION FOR SHELL AND SPATIAL STRUCTURES. VOL 51. P 140.

HUI X , HONG, Q, 2002. A PHYSICS-BASED FRAMEWORK FOR SUBDIVISION SURFACE DESIGN WITH AUTOMATIC RULES CONTROL. PROCEEDINGS OF THE 10TH PACIFIC CONFERENCE ON COMPUTER GRAPHICS AND APPLICATIONS. P.304.

KILIAN, A., OCHSENDORF, J., 2005. PARTICLE- SPRING SYSTEMS FOR STRUCTURAL FORM FINDING. JOURNAL OF THE INTERNATIONAL ASSOCIATION FOR SHELL AND SPATIAL STRUCTURES. VOL 46.P 147.

OHMORI, H., KIMURAA, T., MAENEB, A., 2009. COMPUTATIONAL MORPHOGENESIS OF FREE FORM SHELLS. PROCEEDINGS OF THE INTERNATIONAL ASSOCIATION FOR SHELL AND SPATIAL STRUCTURES SYMPOSIUM.

SHEPERD, PAUL., RICHENSA, P., 2009. SUBDIVISION SURFACE MODELLING FOR ARCHITECTURE. PROCEEDINGS OF THE INTERNATIONAL ASSOCIATION FOR SHELL AND SPATIAL STRUCTURES SYMPOSIUM.

VELTKAMP, M.; 2009. STRUCTURAL OPTIMIZATION OF FREE FORM FRAME STRUCTURES IN EARLY STAGES OF DESIGN. PROCEEDINGS OF THE INTERNATIONAL ASSOCIATION FOR SHELL AND SPATIAL STRUCTURES SYMPOSIUM.

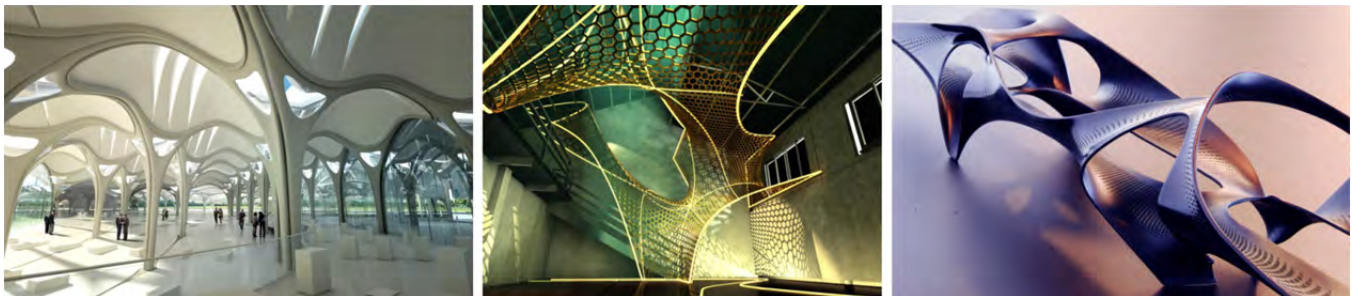


Figure 13. Images of proposals produced using the workflow described.

Irregular Vertex Editing for Architectural Geometry Design

Yoshihiro Kobayashi¹, and Peter Wonka²

¹Arizona State University
699 S. Mill Ave.,
Tempe, AZ 85281, USA
Dr.Yoshihiro.Kobayashi@gmail.com

²Arizona State University
699 S. Mill Ave.,
Tempe, AZ 85281, USA
pwonka@gmail.com

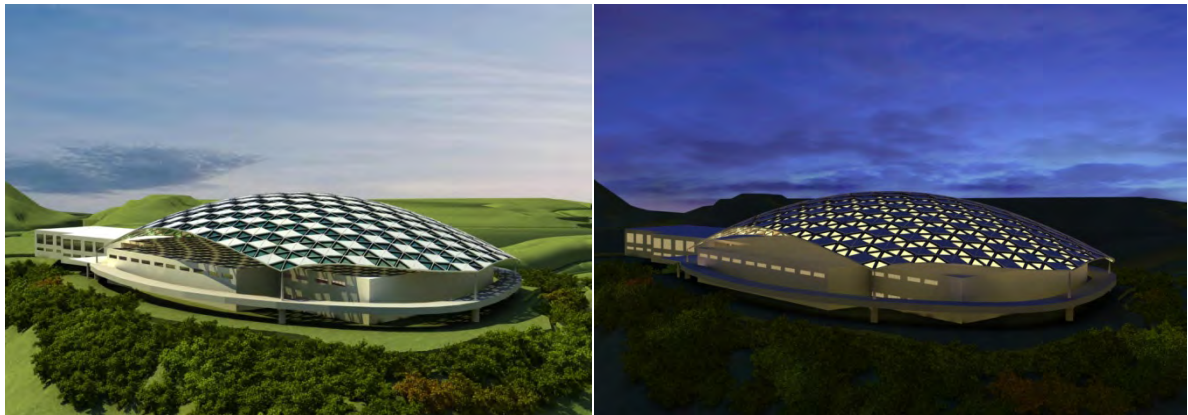


Figure 1: Daytime and nighttime scenes of designed roof by using the developed computational tools

Keywords: Architectural Geometry, Procedural Modeling, Mesh Editing, Computational Design, Computer Graphics.

Abstract

This paper introduces an innovative computational design tool used to edit architectural geometry by addressing the problem of irregular vertices. An irregular vertex is a special kind of vertex, which is connected with less or more edges than regular vertex on a mesh surface. Such vertices create problems with further surface rationalization, as well as structural analysis and constructability of the surface. Geometry created by other tools is remeshed upon import. Using the developed tool, the user is able to identify irregular vertices, and interactively change type, move or remove these irregular vertices. The background of our research and related work on architectural geometry and mesh editing are explained. A case study for a Building Information Modeling (BIM) competition is used to illustrate the workflow. The advantages and disadvantages of editing mesh topology on

architectural geometry design including the limitations are discussed at the end.

1. INTRODUCTION

In the last decade, various unique geometries and free-surface shapes have been applied to architectural design by using the technology of parametric modeling. Parametric modeling is a modeling method to generate geometry by specifying the procedure as an ordered set of parameterized components. Since the components are associated with each other, a designer more strictly controls the properties and attributes of the created geometry. For example, architectural geometry can be generated in three steps using parametric modeling: 1) specifying a location of control points, 2) generating a surface from section curves mathematically defined by using the control points, and 3) subdividing the surface as a set of units and assigning the element to each unit.

However, this approach has a limitation. It works well for the case when the surface is subdivided into a set of

units such as using UV mapping. Otherwise, some unsubdivided parts are generated. Designer and structural engineers have to find a solution to control these unexpected parts without degrading the design. Unfortunately there is no general solution for every kind of surface.

In addition, there are some inevitable geometrical problems. One such problem is an irregular vertex, in which the number of connected edges is less than or more than six (a regular vertex). A second problem is on the distortion of unit of size and shape. It is still a challenge to mesh the entire surface with a regular sized unit. As the result, it leads to the practical problems of structural analysis and construction management.

In order to solve these problems, we developed an approach to mesh the given surface using similarly sized faces and units interactively. It can be made not only step by step but also by editing (type-changing, moving, and removing) the irregular vertices and changing the mesh-topology even after the entire surface is re-meshed. We believe that this is the first direct control mechanism for irregular vertices and that this tool can be an important future design tool to create developable architectural geometry for any kind of free surface.

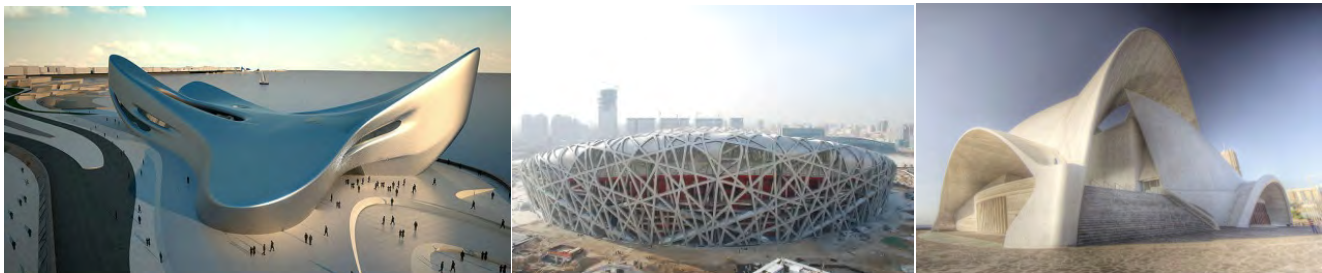


Figure 2: Architectural Geometry designed by well-known architects from Wikimedia: Museum of the Metiterranean in Reggio Calabria by Zaha Hadid (left), Beijing National Stadium by Herzog and de Meuron (middle), and Concert hall in Santa Cruz de Tenerife by Santiago Calatrava (right). [Wikimedia 2010]

2. RELATED WORKS

This paper relates to three main domains: digital architectural design, scientific research on architectural geometry, and professional structural analysis tools.

Recently, many unique architectural designs with free surfaces have been developed by using computational tools. Example buildings designed by well-known architects such as Zaha Hadid, Herzog & de Meuron, Santiago Calatrava, and Jean Nouvel are seen in the real world as depicted in Figure 2. Several books on digital architecture are used in academic programs in schools of architecture, and it is becoming more popular to learn these processes of design and fabrication among students. In most cases, they use the parametric modeling technology with off-the-shelf commercially available software tools such as Generative Component [Bentley 2010], Rhino 3D and Grasshopper [Grasshopper 2010], and other scripting tools within 3D modeling packages.

The scientific research field on computer graphics has also been at the forefront of many research projects with direct application in architectural design, including mesh editing [Hoppe et al. 1993; Surazhsky and Gotsman 2003], vector/tensor field design [Zhang et al. 2006; Zhang et al. 2007; Palacios and Zhang 2007; Fisher et al. 2007], and architectural geometry [Pottmann et al. 2008]. We have also developed some technology to design on a surface, integrating shape grammar and tensor field [Li et al. 2010] and to re-mesh on an architectural geometry surface [Li et al. 2010]. The theoretical background of our tool is briefly explained in the next section.

In a parametric architecture pipeline, after a design is defined, the geometry is sent to engineers to complete a structural analysis. In this process, several off-the-shelf analysis tools are used such as Multiframe and Advanced Steel. Multiframe, developed by FormSys (Formation Design Systems Pty Ltd.), is an integrated structural engineering software package to provide linear/non-linear,

static/dynamic, and buckling calculations. [FormSys 2010]. Advanced Steel, developed by Graitec Inc., is a plug-in tool for AutoCAD to automatically generate structural CAD drawings [Graitec 2010]. It creates the arrangement drawings, fabrication drawings, list of materials and NC files for steel structure design in architecture

3. METHODOLOGY

The objective of this paper is to introduce the developed computational design tool that re-meshes geometry for architectural design, and to show how the tool is integrated in a practical design workflow.

As explained in the introduction, current architectural geometry is created using the technology of parametric modeling. When a designed surface is not subdivided into a set of developable units, some unsubdivided parts are generated. A designer must find a solution to control them without degrading the design case by case.

Also, each unit/face may be distorted or ill-structured. In order to avoid these problems, we developed a theoretical framework to re-mesh a surface without parameterization [Li 2010]. However, it is not easy to control the irregular vertices on the surface because the vertices cannot be cancelled by using regular edge operations such as flip and subdivide edges. We developed 4 main operations: 1) type-change, 2) move, 3) remove, and 4) generate irregular vertices on mesh surface without degrading the mesh quality.

The followings show the terminology used in this paper.

- Regular Vertex: A vertex connected with six edges on a triangle mesh
- Irregular Vertex: A vertex connected with more than or less than six edges on a triangle mesh
- v5: Irregular vertex connected with 5 edges
- v7: Irregular vertex connected with 7 edges

3.1. Type Change

The most fundamental and common irregular vertices are v5 and v7 in a triangle mesh surface. By using three basic graph editing operations; vertex-splitting, edge collapsing, and edge flipping, any other irregular vertices such as v4 and v8 can be converted to v5 and v7. For example, a v4 vertex can be split into two v5 points connected/collapsed by two edges. In order to make the

editing process easier, it is better to make all the irregular vertices as v5 and v7.

3.2. Move

Moving a set of irregular vertices involves changing the position of the points while the other remained irregular vertices are not impacted. Analogous to previous work in vector, tensor, or higher order field design, we would like to design an interface that enables the user to select one irregular vertex and move it using drag-and-drop. Unfortunately, this is impossible for points whose valence is not a multiple of 6. This is somewhat surprising and makes irregular vertex control a challenging problem. The most fundamental operations that we provide are irregular vertex pair movement. The three most important movements are as follows: moving a v5-v7 pair, v5-v5 pair, and v7-v7 pair. The v5-v7 can move over the mesh similar to translation or parallel transport [Zhang et al. 2006]. The v5-v5 and v7-v7 pairs move in non-synchronized fashion, so that the user interface allows for selecting the movement of one of the vertices. The pair moves closer/further together, or circles around a fixed point in between them.

3.3. Remove

From Poincare index theory and Conley index theory we know that removing an single irregular vertex is not feasible. Canceling two irregular vertices is also not possible. This is another reason why valence control is difficult. While it is possible to cancel for irregular points, they have to be in a specific and uncommon configuration. The two most important removal operations operate on irregular vertex triples. A v5-v7-v7 triple can be removed while generating a new v7. A v5-v5-v7 triple can be removed while generating a new v5. By combining the pair-movement described above, it is possible to cancel the irregular vertices interactively.

3.4. Generate

Generation of an irregular vertex is the inverse operation of removal. We can therefore use the same mechanisms as removal. The only practical difference is that generating a quadruple from a regular point is always possible, while finding a quadruple in the right configuration on the existing mesh is difficult.

Because of the space limitation, we don't explain the details and mathematical proof here. The details are explained in the paper [Li et al. 2010]. The next section

illustrates the process to re-mesh on a surface using a sample mesh object, and also explains the framework to integrate the tool in a practical architectural design workflow.

4. WORKFLOW

The following diagram shows our proposed workflow. The first step is to design a free surface in the existing 3D CG modeling packages such as 3DS Max, Maya and Blender. The designed object is exported as an OBJ file, which is then imported into our developed tool. The second step is to re-mesh on the original surface in the developed tool. Several unexpected irregular vertices are generated in the initial re-meshing process, so the user needs to edit the points using the main operations explained in the previous section. Once the editing is completed, the re-meshed data is exported as an OBJ file to a 3D modeling package or structural analysis tool. The material patterns and details are created in the 3D package and geometry data is sent to the structural analysis tools to find the solution as shown in Figure 3.

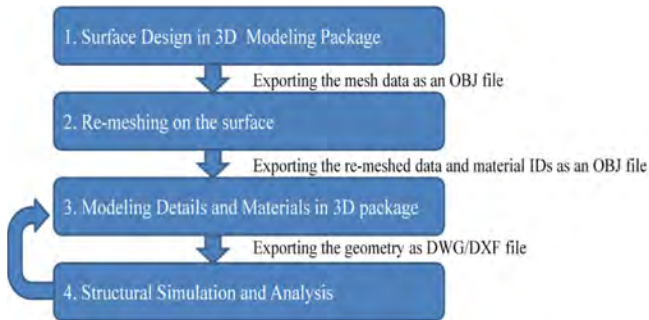


Figure 3: Workflow to Develop Architectural Geometry using our developed tool

4.1. Example Workflow

The following illustrates one example of the workflow described above. As shown in Figure 4, free surface objects designed in 3D packages have several problems. The size of faces is different. There are some too big faces and too tiny faces. The edges are not smoothly lined up. Quad and triangle faces are mixed. Therefore, it is difficult to parameterize the geometry like this. From here, the workflow is explained using the object shown in Figure 4.

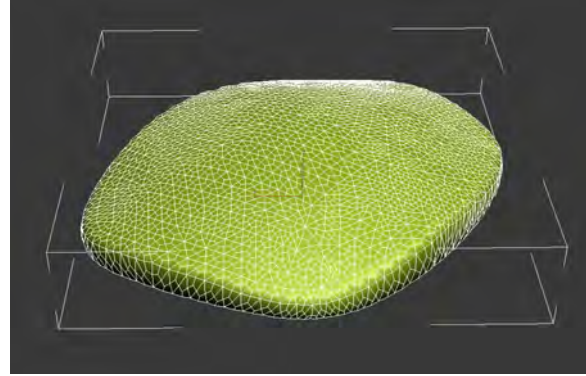


Figure 4: Input Mesh Object.

The left image of Figure 5 is the initial output of re-meshing for the input mesh object. Based on the topological invariant known as Euler characteristics (eq.1), it is mathematically impossible to avoid generating the irregular vertices, which are connected to 5 or 7 edges instead of 6.

$$\chi = V - E + F, \quad (\text{eq. 1})$$

where V is the number of vertices, E the number of edges, and F the number of faces. The Euler characteristic χ was classically defined for the surfaces of polyhedral, and any convex polyhedron's surface has Euler characteristic as 2. In short, any manifold triangle mesh has at least 12 irregular vertices.

The blue point shows the singular vertex with 5 edges, and the orange one with 7 edges. The middle and right images in Figure 5 show the changes of mesh topology before (left) and after (right) a move operation of v5-v7 path. The path is shown in green. We assume that the irregular vertices are expected to cancel as many as while possible keeping the geometry.

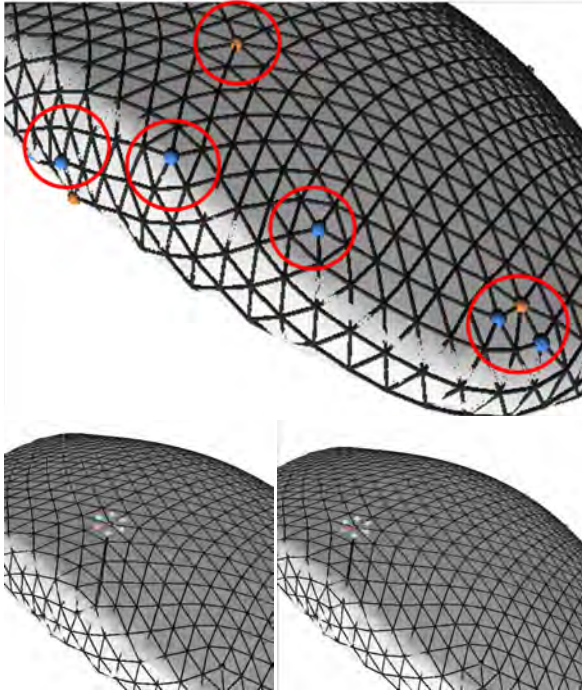


Figure 5: Move Operation of v5-v7 Pair: Irregular vertices on a triangle mesh. V5s are represented as a blue dot and Vs are as an orange dot (Top). A v5-v7 pair is defined (bottom left), and the pair is moved without degrading the mesh quality (bottom right)

First, the user picks up a pair of v5 and v7 points. The system has a function to find the shortest path between the points which is visualized with thicker green frames. Once the shortest path is defined, the system shows the six directions to move (v5 are shown in blue and one selected direction in pink). The user can select one of the directions by using up and down keys, and move the v5-v7 pair and path by typing the “G” key. For each step, the mesh topology is updated/changed. If the changed triangle faces become distorted, the smoothing function (by typing the “O” key) makes the faces more regular.

As explained in section3, a v5-v7 pair can be canceled when it hits another v5 or v7 point. Therefore, the main task is to find the v5 and v7 pair and the destination of the pair. Sometimes the topology changes in unexpected ways, so the user needs to pay attention to the topology, too. It is a little difficult to predict how the topology will be changed by editing at the beginning, but it can be done intuitively with very little practice.

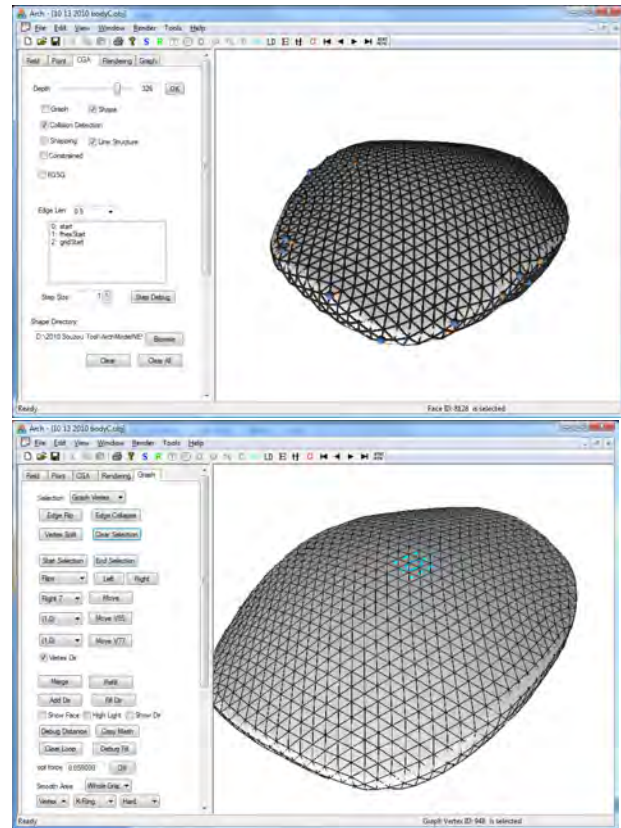


Figure 6: Screen shot images of initial re-meshing (Top) and after editing (Bottom)

Figure 6 shows the screen shot images before (top) and after (bottom) editing irregular vertices. The top image is the result of re-meshing on the original free surface, and several irregular vertices are shown on the top faces. All irregular vertices are moved to the bottom faces or are canceled by editing the v5-v7 pairs.

The tool is developed in C++ and OpenGL, and is operated in regular Windows PCs

5. CASE STUDY

Here we will discuss a case study to develop a free surface roof structure from re-meshed data using the workflow proposed in the previous section. This workflow is applied in a real building information modeling (BIM) competition called “Build Live Tokyo 2010”, in which participating teams design a building in 48 hours.

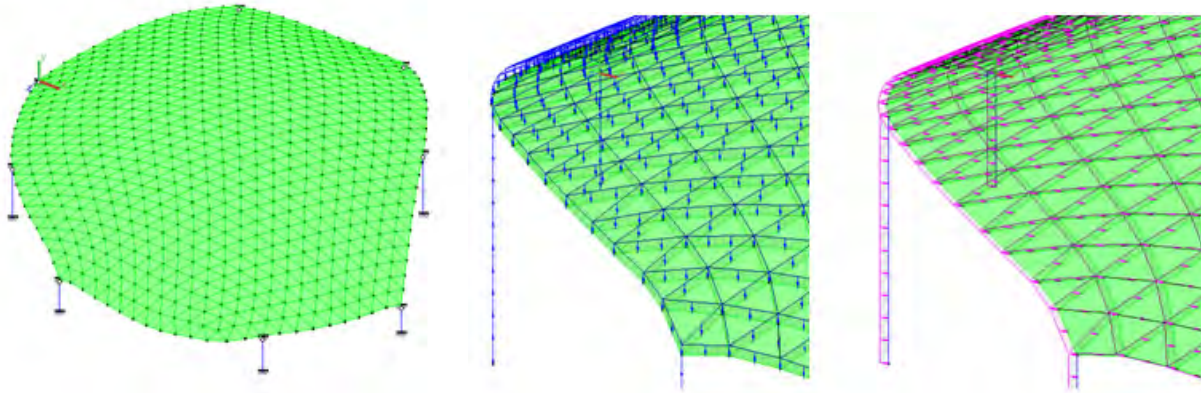


Figure 7: Data Input for Multiframe (left) and the loads for weight (middle) and for earthquake (right)

5.1. Structural Analysis

The re-meshed data is sent to Multiframe, which is an off-the-shelf 3D frame structural analysis tool. By assigning the loads, supports, and cross section dimensions, the structure is analyzed as an elastic beam element model. The structure has 2030 beams and 713 joints. The stiffness is ignored. The weight of frames, columns, roof panels are set as the dead loads, and the earthquake loads are set as the live loads. The wind loads or snow loads are ignored. The deformations, internal forces, and stresses are computed. Figure 7 shows the screen shot images of Multiframe and the loads settings. Figure 8 shows the result of displacement and the optimal location of support columns.

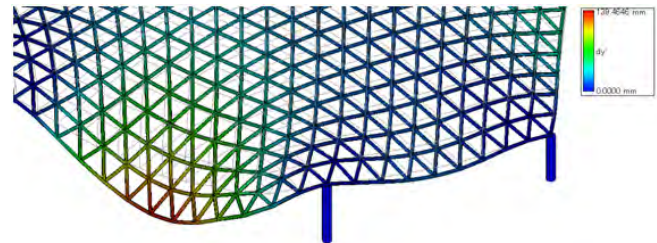


Figure 8: Results in Multiframe.

5.2. Joint Design

The next step is to determine the joint design. We tested several different joints using Advanced Steel, which is a plug-in tool for AutoCAD, to generate structural CAD drawings automatically. Figure 9 shows images of joint designs tested in this project.

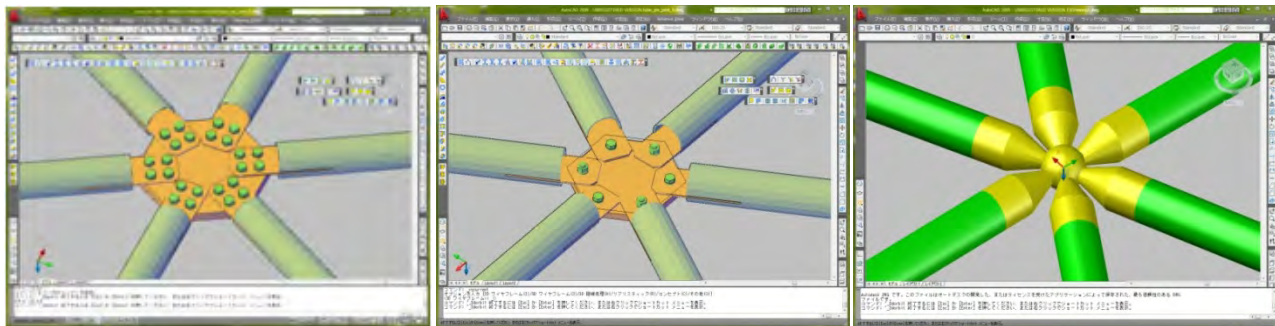


Figure 9: Screen Shot images of Advanced Steel for several joint designs

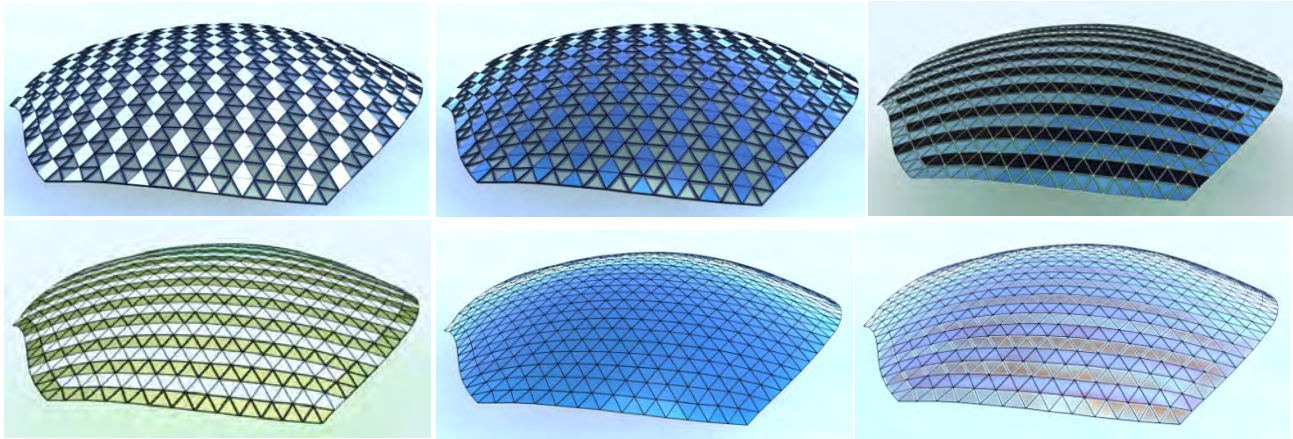


Figure 10: Several Roof Designs

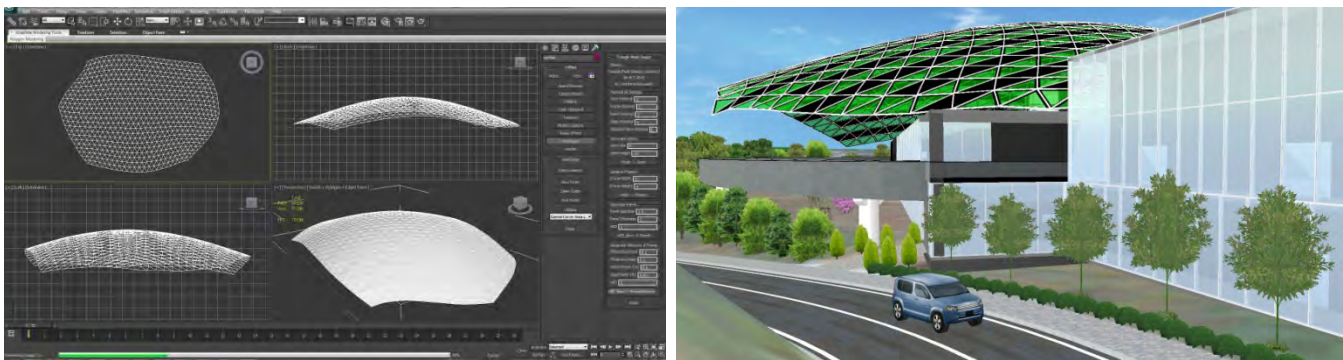


Figure 11: Screen Shot images of customized tool in 3D CG package (left) and VR tool (right)

5.3. Roof Design

Once the structure and the joint design are determined, the last step is to design the pattern and materials for the roof panels (see Figure 10). Several different patterns and materials are generated, visualized, and evaluated visually.

Several alternative designs are generated in 3D CG packages using customized tools in script language, and the designs are then visualized in a virtual reality (VR) environment [UC-win/Road 2010] shown in Figure 11. The customized tool can generate the geometry for joints, frames, and panels. Each part is parameterized, and the designer can change the size and materials interactively in the common 3D modeling environment

6. CONCLUSION AND FUTURE WORKS

We demonstrated the developed computational design tool to re-mesh a free surface object. An example workflow to design a lattice structure roof using the tool was illustrated. The pipeline to design architectural geometry, re-

mesh the surface, assign the details, and complete a structural analysis was explained. Through the case study, we demonstrated that this tool can be applicable for practical architectural geometry development. In fact, all of the process explained in the workflow was carried out in less than 12 hours during a real BIM competition.

The limitation of the current tool is that it can be used to generate only triangle mesh objects. The future goals are to extend this tool to quad mesh objects and to develop the pipeline to generate any kind of design/geometry on any kind of surface.

Acknowledgements

We would like to acknowledge the help of Eugene Zhang (Oregon State University) and Yuanyuan Li (ASU) for the tool implementation, Christopher Grasso and Michael McDearmon for rendering figures, and the support of Forum8 Co Ltd on using VR and structural analysis tools. This project was funded by NSF contracts IIS-0915990, CCF-0643822, CCF-083080, and IIS-0917308.

References

- BENTLEY 2010. ONLINE AVAILABLE ON 11.19.2010 AT [HTTP://WWW.BENTLEY.COM](http://www.bentley.com)
- FISHER, M., SCHRODER, P., DESBRUN, M., AND HOPPE, H. 2007. DESIGN OF TANGENT VECTOR FIELDS. IN SIGGRAPH 2007: ACM SIGGRAPH 2007 PAPERS, ACM, NEW YORK, NY, USA, 56.
- FORMSYS 2010. THE PRODUCT INFORMATION ON MULTIFRAME. ONLINE AVAILABLE ON 11.19.2010 AT [HTTP://WWW.FORMSYS.COM/MULTIFRAME](http://www.formsys.com/multiframe)
- GRASSHOPPER 2010. ONLINE AVAILABLE ON 11.19.2010 AT [HTTP://WWW.GRASSHOPPER3D.COM](http://www.grasshopper3d.com)
- HOPPE, H., DEROSE, T., DUCHAMP, T., McDONALD, J., AND STUETZLE, W. 1993. MESH OPTIMIZATION. IN SIGGRAPH 1993, ACM, NEW YORK, NY, USA, 19-26
- LI, Y., BAO, F., ZHANG, E., KOBAYASHI, Y., AND WONKA, P. 2010. GEOMETRY SYNTHESIS ON SURFACES USING FIELD-GUIDED SHAPE GRAMMARS," IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, 10 FEB. 2010
- LI, Y., ZHANG, E., KOBAYASHI, Y., AND WONKA, P. 2010. ACM TRANSACTIONS ON GRAPHICS (SIGGRAPH ASIA 2010)
- PALACIOS, J., AND ZHANG, E., 2007. ROTATIONAL SYMMETRY FIELD DESIGN ON SURFACES. ACM TRANS. GRAPHCS (SIGGRAPH 2007) 26, 3, 55.
- POTTMANN, H., ASPERL, A., HOFER, M., KILIAN, A., AND BENTLEY, D. 2007. ARCHITECTURAL GEOMETRY, BENTLEY INSTITUTE PRESS, ISBN 978-1-934493-04-5.
- SURAZHISKY, V., AND GOTSMAN, C. 2003. EXPLICIT SURFACE REMESHING. IN 2003 EUROGRAPHICS/ACM SIGGRAPH SYMPOSIUM ON GEOMETRY PROCESSING, 30.
- WIKIMEIDA 2010. ONLINE AVAILABLE ON 11.19.2010 AT [HTTP://COMMONS.WIKIMEDIA.ORG](http://commons.wikimedia.org)
- ZHANG, E., MISCHAIKOW, K., AND TURK, G. 2006. VECTOR FIELD DESIGN ON SURFACE. ACM TRANSACTION ON GRAPHICS (TOG) 25, 4, 1326
- ZHANG, E., HAYS, J., AND TURK, G. 2007. INTERACTIVE TENSOR FIELD DESIGN AND VISUALIZATION ON SURFACE. IEEE TRANSACTION ON VISUALIZATION AND COMPUTER GRAPHICS 13, 1, 94-107

Session 5: Design & Analysis

105 **Solar Zoning and Energy in Detached Residential Dwellings**

JEFFREY NIEMASZ, JON SARGENT and CHRISTOPH REINHART
Graduate School of Design, Harvard University



BEST PAPER AWARD

115 **A simple method to consider energy balance in the architectural design of residential buildings**

LAËTITIA ARANTES, OLIVIER BAVEREL, PASCAL ROLLET and DANIEL QUENARD
Ecole Nationale Supérieure d'Architecture de Grenoble, Centre Scientifique et Technique du Bâtiment and École des Ponts Paris-Tech

123 **A methodological study of environmental simulation in architecture and engineering. Integrating daylight and thermal performance across the urban and building scales.**

PETER ANDREAS SATTRUP and JAKOB STRØMANN-ANDERSEN
School of Architecture, Royal Danish Academy of Fine Arts and Technical University of Denmark

Solar Zoning and Energy in Detached Residential Dwellings

Jeffrey Niemasz, Jon Sargent and Christoph F Reinhart
Harvard University, Graduate School of Design, Cambridge MA, USA

Harvard University, Graduate School of Design
48 Quincy Street
Cambridge, MA, USA, 02138
jniemasz@gsd.harvard.edu

Keywords: Solar Envelope, Solar Zoning, Detached Single Family Residential

Abstract

The Solar Envelope is a three dimensional envelope on a site which ensures adjacent neighbors a specified minimum direct solar access time per day throughout the year. The solar envelope was developed as a tool to give buildings in an urban setting the mutual opportunity to employ passive and active solar design strategies. Parametric computer-aided-design (CAD) environments significantly ease the construction and visualization of solar envelopes across whole neighborhoods, facilitating its wider use as a prescriptive zoning tool. This study investigates the implications of a solar envelope zoning approach for the most common building type in the United States (US) with respect to energy use and developable density. The results indicate that solar zoning for this building type has a limited, and sometimes negative effect on energy use as well as a larger negative impact on developable density.

1. INTRODUCTION

Zoning rules can preserve access to daylight through a buildable envelope established by setbacks and/or height restrictions. These rules constrain development of neighboring sites or development on the site itself. The English “right to light” easements of the 1800’s are an example of the former while New York’s 1916 Zoning Resolution is an early example of the later. The Solar Envelope is an alternative approach to zoning (Knowles 1974). Like New York’s zoning envelope, the Solar Envelope follows a “good neighbor” approach by constraining development within the site. “I am my neighbor’s neighbor” when it comes to solar access (Knowles 1981). But instead of static setbacks to ensure daylight, the solar envelope is latitude and orientation specific. Any building contained within the solar envelope

will not cast a shadow outside the specified boundary for a specified amount of time during the winter solstice. In his early description of the solar envelope, Knowles stated a ‘minimum period of six hours a day [to be] practical’ (Knowles 1981, p. 56). Arguments in favor of ensuring solar access of a site include opportunities for active and passive solar design, aesthetics, and quality of life.

Traditionally, the generation of solar envelopes used to be a somewhat tedious process that required the use of paper charts and/or physical models combined with heliodons (Knowles 1981, Brown and Dekay 2001). This requirement probably slowed the adoption of the solar envelope as a zoning policy simply because the earlier described setback rules were easier to implement and enforce. The situation has changed in recent years with the widespread availability and use of parametric three-dimensional CAD environments that allow the implementation of the solar envelope. As an example the authors developed a Grasshopper component for the popular Rhinoceros/Grasshopper program (Rutten 2010). Based on street plans, site latitude, and required solar access time, the solar envelope component generates in real time a neighborhood of solar envelopes. In spring 2010, the use of the solar envelope component was tested on a group of first and second year architecture students taking an introductory class on environmental technologies (Figure 1). The assignment was mastered by all students showing how simple the generation of solar envelopes has become. The premise of the exercise was to demonstrate how, on an urban massing level, the solar envelope could be used to increase developable density while ensuring solar access compared with traditional height restrictions.

In principle, such computer-based tools now allow the wide-spread implementation of the solar envelope concept as a prescriptive zoning tool. Before embracing this approach, the authors asked whether this *should* indeed be

done and what solar access times should be recommended for different climates? A concern of the authors was that increasing solar access comes at the cost of a decreased envelope volume, particularly at higher latitudes, and hence lower developable densities which result in increased vehicle miles travelled (VMT) and thereby increased energy use. Research has demonstrated that land use and transportation patterns, especially the inclination of local zoning in the US towards automobile dependent and low density detached residential housing, play a major role in the rate of gasoline consumption (Newman and Kenworthy 1989).

The tradeoff between the transportation energy benefits of residential density on an urban scale compared to its reduction in a site's available solar energy has been explored for specific locations and housing types. A recent study showed that residential energy use in Toronto is closely tied to the transportation energy use of households living in neighborhoods of varying densities (O'Brien et al. 2010). For three building types of varying densities, the available solar energy was compared with household and transportation energy (O'Brien et al. 2010). Those living in the densest housing types used the least energy, even with the diminished potential for solar conversion. Interestingly, the medium density type, attached condominiums, showed the highest energy use because it combined reductions in solar availability without the offsetting benefits of reduced transportation energy typical of higher residential densities. However, in a scenario where every residence maximizes solar collector coverage and the entire transportation fleet is upgraded to plug-in hybrid electric vehicles, the model showed a reversal of the trend favoring density. (O'Brien et al. 2010). Under current and hypothetical future scenarios, the model suggests a significant tradeoff between solar access and transportation energy. Zoning tools like the solar envelope may be particularly useful in minimizing energy use through the maximization of solar access and developmental density if used in the right situations.

The magnitude of the effect on density depends on how the solar envelope is implemented in a neighborhood. Knowles proposed that the solar envelope concept can refer to different parts of a building site, i.e. it can be concerned with the whole site, the whole building, or the rooftop only (Knowles 1981). Providing solar access to the whole site obviously leads to the lowest urban density, whereas access to the rooftop results in the highest.

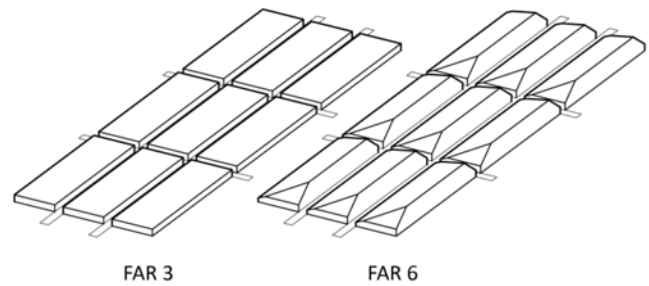


Figure 1. Solar Envelope exercise in the Environmental Technologies course at Harvard in spring 2010. On an urban massing model at 40 degrees latitude with a height limit of three stories, the solar envelope zoning approach of mixed height restrictions shows the capacity to double the developable density while ensuring 4 hours solar access to the street faces.

This study presents an attempt to quantify the energy and urban density implications of using the 'whole building' solar envelope as a zoning policy in a development consisting of 'typical' North American single family detached residential homes. The whole-building solar envelope was chosen since typical residential construction does not take advantage of solar collectors or photovoltaics and therefore opportunities for energy savings from solar design mainly stem from direct solar exposure of the walls and windows of a building. Of course, providing solar access to the whole building also ensures generous solar access to the rooftop for the future use of solar hot water and photovoltaics. The single family detached residential home was chosen as the building type for this study because about 60% of US homes are currently of this type (US Census Bureau 2005). As described below, the analysis in this study is based on annual whole building energy simulations using EnergyPlus. The simulations were conducted using another new Grasshopper component that exports and runs EnergyPlus models.

2. METHODS

2.1. Solar Envelope Component

As mentioned above, the authors built a Grasshopper component to facilitate the rapid construction and visualization of solar envelopes within design oriented software. Grasshopper is a parametric scripting tool and a plug-in for the three-dimensional non-uniform rational B-spline (NURBS) modeler for Windows Rhinoceros® (Rhino) (McNeel 2010). The component creates a series of

extrusions of the boundary which defines the adjacent sites requiring solar access. This two dimensional boundary is extruded along four reversed solar vectors for the cutoff times on the summer and winter solstice. A Boolean intersection of these four extruded volumes produces the maximal envelope which will not cast a shadow throughout the year between the cutoff times (Figure 2). Many three-dimensional modeling programs should allow the scripting of a solar position algorithm, curve extrusions, and intersection procedures.

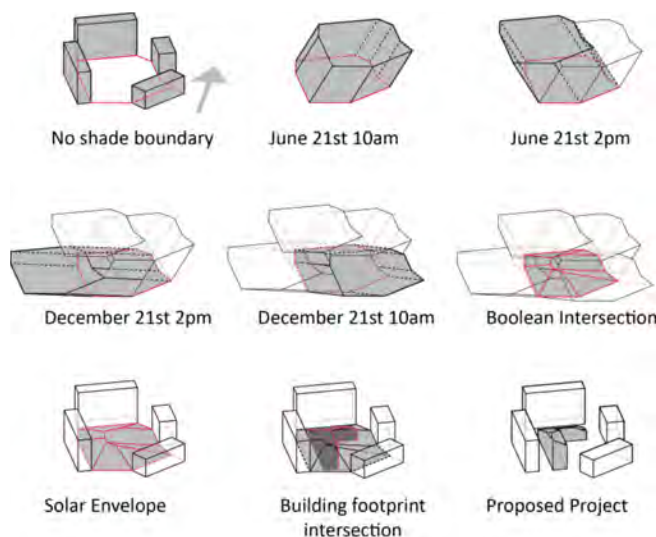


Figure 2. A method of Solar Envelope Construction at 38 degrees latitude for cutoff times of 10am and 2pm. A no shade boundary is constructed for the adjacent properties where solar access will be protected. This boundary is extruded along the reverse solar vector of the four cutoff times. A Boolean intersection solves for the maximal volume which will not cast a shadow between these times. All proposed building projects within this envelope will provide the adjacent properties a minimum of 4 hours direct solar access.

2.2. The Residential House

One of the Canadian Centre for Housing Technology's (CCHT) houses was modeled in Rhinoceros and Design Builder. DesignBuilder is a visual interface for the US Department of Energy's whole building energy simulation software EnergyPlus (EnergyPlus 2010). The CCHT house is a 210 m² two-story wood-frame design built in 2001. The building serves as a research facility and is supposed to represent typical new North American residential construction (Swinton et al. 2003). The house has been extensively used in physical and energy modeling research (e.g. Purdy and Beausoleil-Morrison 2001). An eight-zone model was constructed with one zone for each floor with a corresponding stairwell zone, attic space, and garage (Figure

3). The living spaces and basement zones were conditioned by the heating, ventilation, and air-conditioning (HVAC) system. The attic and garage zones were left free floating. Ground contact in the basement was modeled using the ground temperatures from the utilized Typical Meteorological Year 3 (TMY3) weather files. Seven weather files were selected for cities representative of the climate zones described in the 2009 International Energy Conservation Code (IECC) for residential energy efficiency. The cities are Anchorage, Denver, Los Angeles, Miami, Minneapolis, New York, and Phoenix. Factors such as urban heat island effect or local wind patterns are not accounted for in this approach, but are likely to be less significant with such a low density building type.

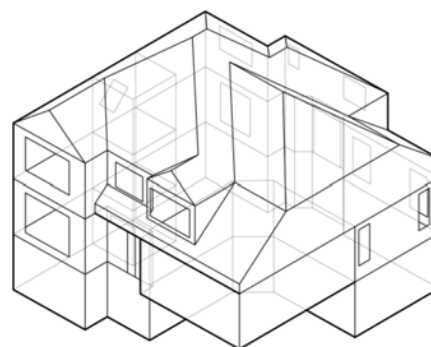


Figure 3. Eight zone model of the test house.

The surfaces were modeled with typical lightweight wood frame construction including gypsum board, oriented strand board (OSB), and extruded polystyrene (EPS). The thickness of these layers was adjusted to achieve the climate zone respective U factor recommended by the guidelines in the 2009 IECC for residential energy efficiency.

The house was placed in the smallest possible solar envelope which would contain it for a given solar access time period, thereby providing the maximal developable density. Following the CCHT house, the building was oriented to the South, with 16.2 m² of south facing glazing. The envelope was then multiplied to generate the zoning envelopes for the adjacent properties. A 14m right-of-way was carved from every other envelope to represent a minimum easement for automobile access and sidewalks typical of a North American residential neighborhood (Southworth and Ben-Joseph 1995) (Figure 4).

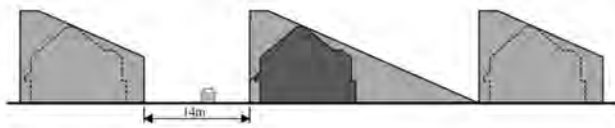


Figure 4. North-South section of neighborhood zoned for a minimum of 4 hours solar access in Denver. A 14 meter right-of-way truncates every other envelope for vehicular and pedestrian access.

Hypothetical communities were modeled according to solar envelopes which ranged from 0 to 8 hours at 6 minute increments. A separate set of envelopes were generated for each of the seven cities described above according to their varied latitudes.

The parameters for the solar envelope are the latitude and duration of ensured solar access. The model of the test house was surrounded by its adjacent solar envelopes for the given climate and minutes of solar access (Figure 5). The times of solar access were centered on noon, thereby maximizing the volume of the envelopes. Solar access could occur outside this time frame, but zoning according to a morning or afternoon time frame, such as 8 am to 11 am, dramatically reduces the volume of the solar envelope because of the respective profile angles of the sun on the winter solstice. In EnergyPlus these envelopes were modeled as shading blocks with a reflectance of 22%, representing the worst case shading scenario for the test house. Using the arrangement from Figure 5, neighboring properties can in principle completely fill their respective solar envelope. But in fact, each neighboring building is only guaranteed its own solar access if it remains within a footprint similar to the footprint of the test house. As mentioned above, guaranteeing solar access to the entire site, an approach called whole-site conversion, dramatically reduces developable density. Limiting solar access to the footprint of the house, or whole-building conversion, allows a greater duration of solar access than the envelope strictly predicts (Knowles 1981). In fact, the beam radiation received by the windows of the test house surrounded by increasing solar envelopes varies little once the first minute (the winter solstice noon angle) of solar access is ensured to the building footprint (Figure 6). The models of the home and their respective adjacent envelopes were exported from Rhinoceros/Grasshopper to EnergyPlus for an annual energy simulation at hourly time steps.

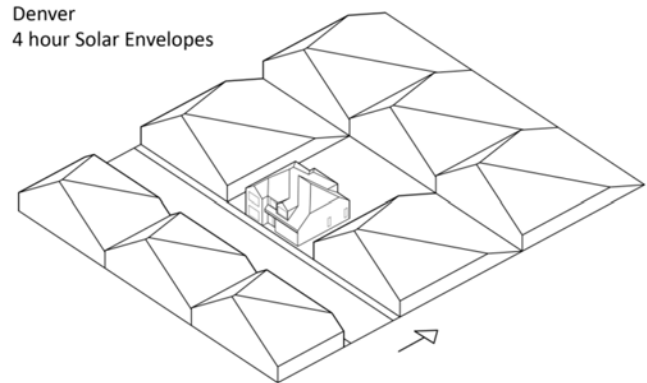


Figure 5. Test house with adjacent maximal building envelopes and public easement for sidewalk and street. The envelopes in this variant ensure a minimum of 4 hours solar access to the test house in Denver. (39.83 degrees latitude)

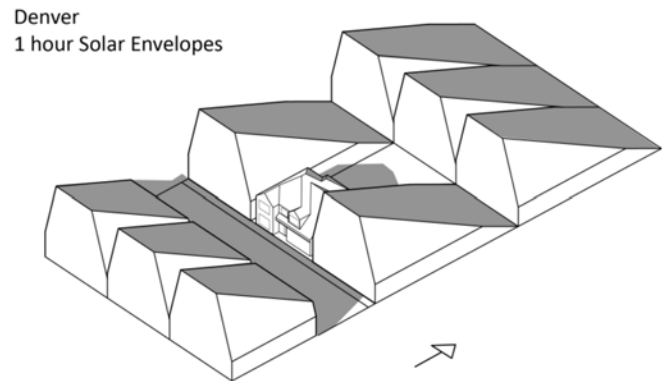


Figure 6. Test house zoned with envelopes to ensure a minimum of 1 hour solar access in Denver. The December 21st shadow line at 11:30am just misses the foundation of the test house; a small shadow will appear on the house at 12:30pm. The windows of the test house will clearly receive greater than 1 hour of solar access on the winter solstice, but the envelope boundary as a zoning tool cannot predict where windows will be placed so a whole-building approach is necessarily over cautious.

2.3. Developable Density

Once the solar envelope models were created, the number of dwelling units per acre (du/acre) was calculated. The no-shadow boundary of each solar envelope extended to the building footprint on the adjacent properties. The envelope itself was then truncated to allow right-of-way for a road and sidewalk and along the sides to evenly divide the property. Dividing the one dwelling by the acreage of the resulting site boundary for each zoning variant gave the respective du/acre.

3. RESULTS

3.1. Energy Use

The annual energy use of the test house was simulated for seven representative cities. For each location, the house was placed within a neighborhood zoned for solar access ranging from 6 minutes to 8 hours at 6 minute increments, for a total of 672 simulations. For example, Figure 7 shows the amount of solar radiation incident through all windows over the course of the year for different solar access times in Denver. As solar access time increases, annual solar gains also rise by 50% from 48 to 72 kWh/m²/yr. In contrast, Figure 8 shows that the onsite energy use for heating, cooling, lighting and equipment only decreases in Denver by about 5% for the investigated house, mainly because heating savings are balanced by increased cooling needs. If one were to use the recommended six hours of solar access, total annual energy savings would be 2.7%.

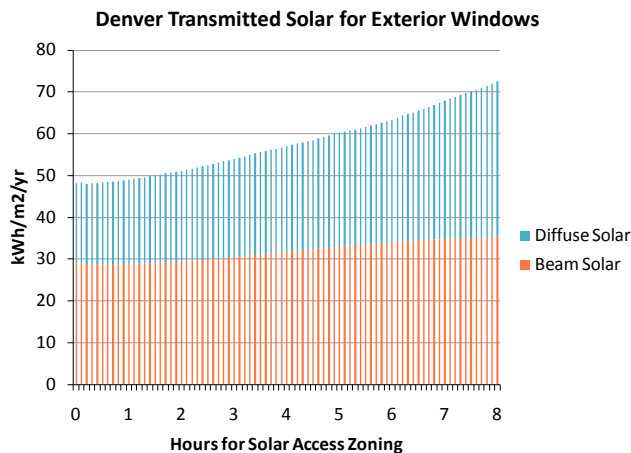


Figure 7. From the first minute of solar access to the building footprint in a whole-building solar envelope strategy the beam solar radiation varies less than the diffuse component. In effect, the test house is nearly saturated with beam radiation by the time the entire footprint receives sunlight at noon on the winter solstice.

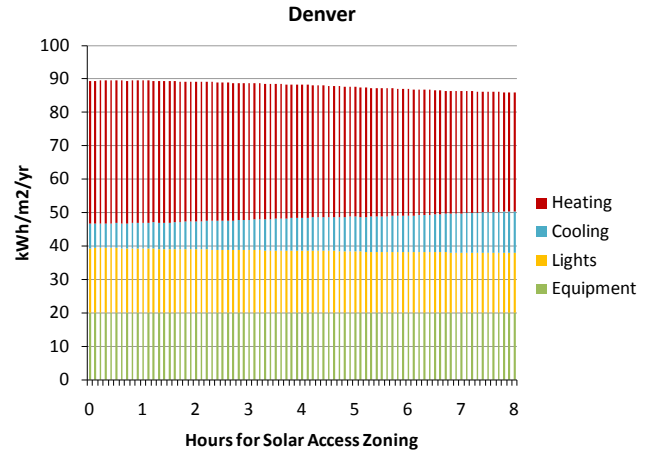


Figure 8. Annual energy use for the test house in Denver zoned from 6 minutes to 8 hours of solar access.

Figure 9 shows total energy use in relation to solar access time for all cities tested. Since no optimized or seasonally adaptive shading devices were part of the model, increasing solar access actually increases on site energy use in cooling dominated climates such as Phoenix and Miami.

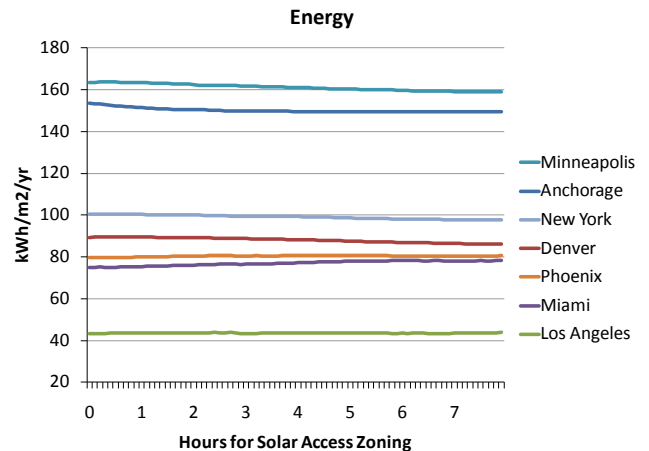


Figure 9. Annual energy use for the test house in seven representative cities zoned from 6 minutes to 8 hours of solar access.

Figure 10 is equivalent to Figure 9 and shows the annual carbon dioxide equivalents (CO₂e). In accordance with section 7.5.3 of ASHRAE Standard 189.1, a CO₂e emission factor of 0.758 kg/kWh was used for electricity and 0.232 kg/kWh for natural gas heating.

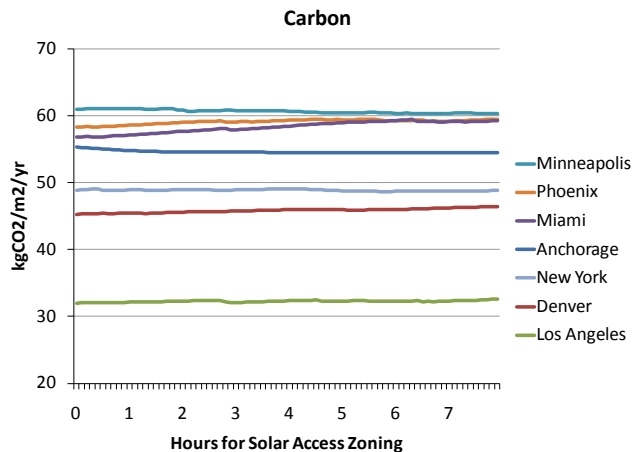


Figure 10. Annual carbon dioxide equivalent for the test house in seven cities zones for 6 minutes to 8 hours solar access.

Figure 11 is also equivalent to Figure 9 but adjusts the cost of energy to the 2009 cost of electricity and natural gas for each location as reported by the U.S Energy Information Administration of the Department of Energy (www.eia.doe.gov). This data varies by region and time, but would be an important component for convincing a local government to incorporate solar access into their residential zoning. The lower cost of natural gas compared with electricity helps Minneapolis, the city with the highest energy use, enjoy the lowest energy cost. On the other hand, increased solar access in Denver drives down energy use but with increased CO₂e and cost because the shift in natural gas heating is not offset by the increased electricity for cooling.

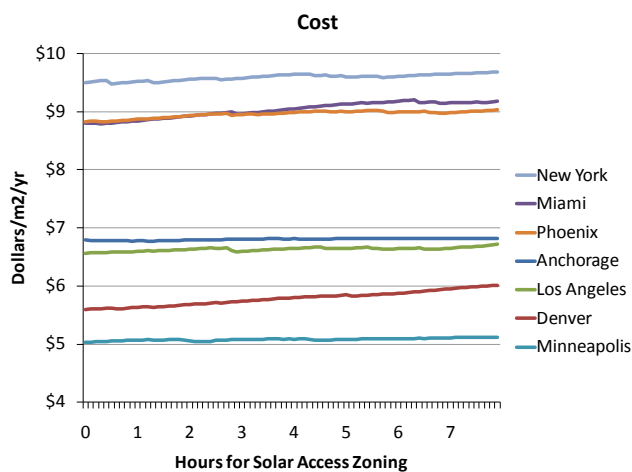


Figure 11. Annual energy cost for the test house in seven representative cities zoned from 6 minutes to 8 hours of solar access.

Figure 12 shows the change in energy, cost, and CO₂e for each location under a six hour solar access regime compared to a 1 minute envelope (maximum urban obstructions). All locations show an increase in cost with 6 hour solar access zoning. Cooling dominated climates such as Phoenix and especially Miami show a consistent increase in energy, cost, and CO₂e. Energy reduction is largest in Denver but comes at a cost increase of close to 5% and an increase CO₂e because reduction in heating loads were offset by a less efficient and regionally more costly cooling load.

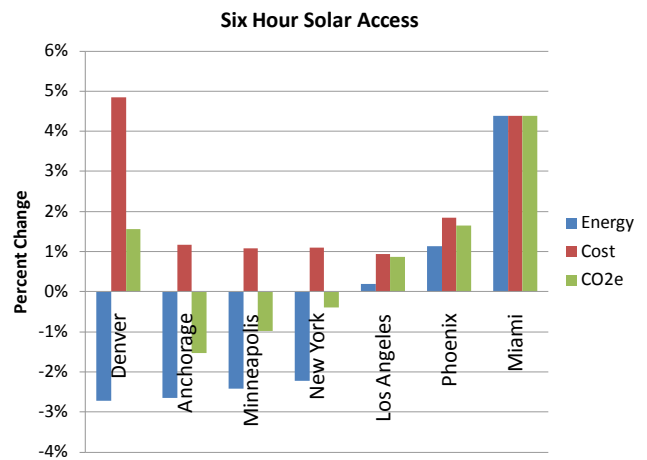


Figure 12. Change in energy, cost, and carbon dioxide equivalent (CO₂e), under a six hour solar access zoning compared with 1 minute solar access.

3.2. Developable Density

Figure 13 shows the developable density in dwelling units per acre (du/acre). The chart highlights those locations and solar access zoning hours which would achieve developable densities greater than 7 dwellings per acre (du/acre) which is the 2009 Leadership in Energy and Environmental Design (LEED) for Neighborhoods cutoff value for sites not in walking distance to a public transit service. The goal of this metric is to promote compact cities. None of the locations, even when zoned to a minimum of one minute of solar access, would meet the 12 du/acre LEED for Neighborhoods cutoff for sites within walking distance to public transit. Only the Miami location, when zoned for less than 1 hour of solar access, would receive credit for compact development by exceeding 10du/acre. Combing Figures 12 and 13 shows that solar access zoning for this single family detached housing achieves the highest

developable densities where it is not effective in reducing energy use.

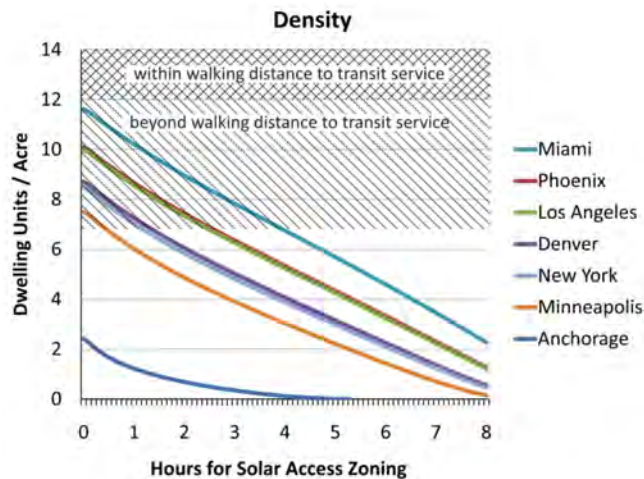


Figure 13. Developable density for neighborhoods zoned from 6 minutes to 8 hours of solar access with test house type. In this case, density is only a function of latitude and hours of solar access. 7 du/acre is the lower limit of density for 2009 LEED Neighborhood for sites outside of walking distance to a public transit service and 12 du/acre is the lower limit for sites within the walking distance of ¼ mile.

4. DISCUSSION

For the investigated building type, applying the concept of the solar envelope has a limited benefit on energy use in heating dominated climates and no benefit in cooling dominated climates. The solar envelope has a marked negative impact on developable density, especially for more northern latitudes. One may learn from this that regular infrastructural requirements such as vehicular access already provide an urban spacing which ensures an acceptable degree of solar gain for typical detached single family residences. Other building types and zoning scales, such as the urban massing in Figure 1, may show increased benefit in terms of density and energy in certain locations.

The limitations of the solar envelope as a prescriptive residential zoning tool for detached residential housing do not detract from its use as a site and context-specific design tool. In these situations, the envelope can be calibrated to ensure solar access to existing windows, solar collectors, or a selected open space, while maximizing the buildable volume and design options. A variation of the solar envelope called a landscape envelope, allows visualization of optimal locations and sizes for deciduous trees (Knowles 1981). As seasonally dynamic shading devices, properly

located deciduous trees would mitigate the cooling load in some climates.

Our approach focused on discerning the energy effects of a solar zoning approach for typical US residential construction. Homes were oriented along east-west streets to provide a south facing best case scenario. Changing the design of the house, adding solar collectors, incorporating optimized shading devices, planting trees, and varying the street pattern are all variables which would have an impact on energy use. By holding these variables constant, we were testing the feasibility of a solar zoning approach without additional guidelines. Of these variables, the energy from solar collection would vary least for this building type at its typical densities. This is because a whole building approach was used for solar zoning, meaning that solar access was ensured for the building footprint. Except for the extreme minimums of solar access zoning, rooftop solar access is not compromised.

On a general note, the problems with achieving an acceptable density are essentially a consequence of the residential type chosen. Taken together, the energy and density results argue that typical American residential construction faces serious challenges in significantly reducing its carbon footprint. In fact, per capita energy consumption and greenhouse gas (GHG) emissions are 2 to 2.5 times higher in low-density developments than in high-density areas (Norman et al. 2006). The small magnitude of the effect of solar zoning on CO₂e compared with a potential 2 fold decrease in GHG emissions from increasing residential density suggests that the most effective intervention might be to abandon or significantly change the dominant housing type. The US is not stuck with its current housing stock; nearly half of the built environment is projected to be constructed in the next 25 years (Nelson 2006). The study suggests that current trends towards infill development in urban neighborhoods have the potential to significantly reduce overall energy use and carbon emissions. The form and type of housing development is largely controlled by local governments through zoning and land use regulation. Municipalities should hence concentrate on providing zoning laws that facilitate the generation of infills. The use of the solar envelope may serve in this context as a tool to ensure some minimum level of solar access is maintained in urban neighborhoods for aesthetics, comfort, and health.

The uniformly negative impact of increased solar access on energy, carbon, and cost in cooling climates like Phoenix and Miami can be extrapolated to other building types in a wider range of climates (Figure 12). From an energy standpoint, tall, internally loaded buildings might benefit from a zoning tool which would ensure that sites around a given building are instead built up to a minimum height, opposite of the solar envelope strategy. A study of an eight story office building model placed in varying street configurations in a Chicago climate showed that compact geometries with high developable densities are associated with significant energy savings potential (up to 37%) by reducing cooling load (Bhiwapurkar and Moschandreas 2010). Importantly, this study did not account for urban heat island effects, which would have moderated the energy savings.

The cost of energy is disconnected from its carbon footprint. The development of more effective zoning tools for reducing energy use face an uphill battle when the low cost of energy in the United States is disconnected from its carbon consequences. In this study, the discrepancy between the pricing of energy and its carbon equivalent led to several contradictory results. In Denver, increased solar access is associated with reduced energy use, but an increase in CO₂e and cost. Minneapolis has the highest energy use and associated carbon emissions of the seven cities but also the lowest operational energy costs.

In summary, it is time to moderate some of the generic claims for the energy benefits of solar exposure. Current sustainability focused guidelines continue to perpetuate a “South facing” façade bias. LEED 2009 for Neighborhood Development awards 1 point for Solar Orientation, meaning a building’s longer east-west axis in within 15 degrees of geographic east-west, regardless of the buildings climate, program, or construction (US Green Building Council et al. 2010). As this study suggests, the effects of direct solar exposure on energy vary significantly with climate and construction. We have adopted the LEED Neighborhood recommendations for residential density in our assessment, a guideline which draws a relationship between developable density and proximity to public transit. As discussed above, there is evidence that this relationship is meaningful for residential energy use in Toronto (O’Brien et al. 2010). Our findings suggest that for single family detached residential buildings, the strength of the tradeoff between solar access and density may vary significantly based on climate. There

is obviously a need for more climate-specific guidelines and standards that holistically integrate the concerns of solar access and developable density.

5. CONCLUSION

Effective zoning tools would produce buildable envelopes with advantageous energy properties at sufficient developable densities (Knowles 1981). How generalizeable such zoning tools are across climates and building types is the question. For the most common residential building type in the US, increased solar access through solar zoning comes at a high price in developable density and has small to negative energy, carbon, and cost implications. The task of designing sustainably resists interventions which are not integrated into the larger built environment and site specific conditions. Urban form, transportation patterns, building organization, the technology of building parts, and occupant use can interact in complicated and counterintuitive ways. Multi-scalar simulations offer the opportunity to target energy saving interventions and free design from outdated rules-of-thumb or overly generalized principles. Future advances in simulations on an urban scale, such as urban heat island effect, hold the promise of increased integration between urban, building, and landscape designers to make meaningful and coordinated reductions in energy use. This study contributes a spatial and thermal simulation method which assesses the top-down aspect of solar zoning with the bottom-up energy analysis of a typical detached residential house. Such an approach may help identify effective zoning tools for reducing energy use which are climate and building type specific.

Acknowledgements

The first author is thankful to Matthias Sauerbruch and Louisa Hutton for guidance and critique during his initial explorations of the solar envelope concept as a zoning tool during their “Language of Sustainability” design studio at Harvard’s Graduate School of Design in Spring 2010. The EnergyPlus Grasshopper component of the CCHT house was developed with support from Harvard’s Wyss Institute for Biologically Inspired Engineering. The writing of this manuscript was supported by the “Sustainability in Design” GSD Research Lab.

References

- BHIWAPURKAR, P., MOSCHANDREAS, D. 2010. Street Geometry and Energy Conservation of Urban Buildings in Chicago. *Intelligent Buildings International*. 2:233-250.

- BROWN, G.Z., DEKAY, M. 2001. *SUN, WIND & LIGHT*. NEW YORK: JOHN WILEY & SONS. 89-98.
- CONDON, P.M., CAVENS, A., MILLER, N. Urban Planning Tools for Climate Change Mitigation. The Lincoln Institute of Land Policy. www.lincolnst.edu accessed 11/14/2010.
- CONGRESS FOR THE NEW URBANISM, NATURAL RESOURCES DEFENSE COUNCIL, U.S. GREEN BUILDING COUNCIL 2010. LEED 2009 FOR NEIGHBORHOOD DEVELOPMENT. WWW.USGBC.ORG ACCESSED 10/20/2010: 42,53.
- KNOWLES, RALPH L. 1974. *ENERGY AND FORM*. CAMBRIDGE, MASSACHUSETTS: MIT PRESS.
- KNOWLES, RALPH L. 1981. *SUN RHYTHM FORM*. CAMBRIDGE, MASSACHUSETTS: MIT PRESS.
- MCNEEL, R. 2010. RHINOCEROS: NURBS MODELING FOR RHINO. (VERSION 4.0 SR 8) [COMPUTER SOFTWARE]. ROBERT MCNEEL AND ASSOCIATES. AVAILABLE FROM WWW.RHINO3D.COM
- NELSON, A.C. 2006. Leadership in a new Era. *Journal of the American Planning Association*. 72(4): 393-409.
- NEWMAN, P.W.G., KENWORTHY, J.R. 1989. Gasoline Consumption and Cities: A comparison of U.S. Cities with a Global Survey. *Journal of the American Planning Association*. 55:24-37.
- NORMAN, J., MCLEAN, H.L., KENNEDY, C.A. 2006. Comparing high and low residential density: Life-cycle analysis of energy use and greenhouse gas emissions. *Journal of Urban Planning and Development*. March:10-21.
- O'BRIEN, W.T., KENNEDY, C.A., ATHIENITIS, A.K., KESIK, T.J. 2010. The Relationship Between Net Energy Use and the Urban Density of Solar Buildings. *Environment and Planning B: Planning and Design*. 37:1002-1021.
- PURDY, J., BEAUSOLEIL-MORRISON, I. 2001. The Significant Factors in Modeling Residential Buildings. Seventh International IBPSA Conference. 9th Canadian Conference on Building Sciences and Technology. 207-214.
- RUTTEN, D. 2010. GRASSHOPPER: GENERATIVE MODELING FOR RHINO. (VERSION 0.7.0045) [COMPUTER SOFTWARE]. ROBERT MCNEEL AND ASSOCIATES. AVAILABLE FROM [HTTP://WWW.GRASSHOPPER3D.COM](http://WWW.GRASSHOPPER3D.COM)
- SWINTON, M.C., ENTCHEV, E., SZADKOWSKI, F., MARCHAND, R. 2003. Benchmarking Twin Houses and Assessment of the Energy Performance of Two Gas Combo Heating Systems. 9th Canadian Conference on Building Sciences and Technology. 365-381.
- SOUTHWORTH, M., BEN-JOSEPH, E. 1995. Street Standards and the Shaping of Suburbia. *Journal of the American Planning Association*. Winter 1995: 65-81.

A simple method to consider energy balance in the architectural design of residential buildings

ARANTES Laëtitia^{1,2}, BAVEREL Olivier^{1,3}, ROLLET Pascal¹, QUENARD Daniel²

¹ UR Architecture, Environnement
et Cultures Constructives (AE&CC)
ENSAG (Ecole Nationale Supérieure
d'Architecture de Grenoble)
60 avenue de Constantine, BP 2636
38036 Grenoble Cedex 2

² CSTB – Grenoble
(Centre Scientifique et Technique
du Bâtiment)
24 rue Joseph Fourier
38400 Saint Martin d'Hères

³ UR NAVIER
École des Ponts Paris-Tech
6-8 avenue Blaise Pascal
Cité Descartes
Champs-sur-Marne
F-77455 Marne-la-Vallée cedex 2

Keywords: Sustainable design, Urban forms, Towers, Energy-saving.

Abstract

This paper describes a “morpho-energetic” study that is part of a broader study that considers the morphology of energy-saving cities. Nowadays, French urban planners recommend denser cities. Are dense and compact cities really sustainable? How can dwelling designs balance the environmental issues with the inhabitants' wishes? This study is focused on two urban typologies: the “nanotours” (little tower concepts made up of houses) and the refurbishment of 1960s-70s towers. In order to check the energy performance of these high-rise forms, this research proposes a simple method to consider an energy balance in the architectural design of buildings. In particular, this paper examines the link between the size of a building and its whole energy consumption (heating and cooling, lighting, hot water, building energy and the inhabitants' consumption). The input data is the primary energy consumed during both the construction and the use of a building. The output parameters are the dimension ranges (size) of an energy-saving building. In conclusion, the paper explains the prospects of this study and its possible implementation for energy-saving cities.

1. INTRODUCTION

Nowadays, in France, as in other European countries, politicians and urban planners are facing a myriad of challenging issues, such as social diversity, access to urban services, land cost control, and sustainable development. Considering these facts, they recommend denser and more compact cities. Their watchword is to densify cities and to fill urban waste grounds. Several questions still remain: are

dense and compact cities really sustainable? How can one design dwellings that suit both the environmental issues and the inhabitants' wishes? This study is focused on two urban typologies that could propose satisfying dwellings in urban centers: the “nanotours” and the refurbishment of 1960s-70s towers. Designed in 2005 by French architecture students in Grenoble (France), the “nanotours” are little towers less than 50 meters in height. Made up of stacked individual units, they include privacy features like outdoor spaces, appropriation and internal layout flexibility. The “nanotours” are built in urban centers near urban equipment and services. “Tour d'EauRizon” (Figure 1) is an example of a “nanotour” in Grenoble, designed by architecture students in 2009. Today, these “nanotours” are building concepts that are still being evaluated.

In order to check the energy performance of these high-rise forms, this paper proposes a simple method to consider an energy balance in the architectural design of buildings. This simple method is a way to study the link between the size of a building and its whole energy balance, including all construction aspects and energy flows.



Figure 1. Tour d'EauRizon, in Grenoble (Arantes *et al.* 2009).

2. FORM AND ENERGY: LITERATURE REVIEW

Research to assess how much a building's form affects its energy performance has been the topic of many publications. Most of these papers study only one geometric parameter: the "compactness" (weak shape coefficient). In the early 2000s, (Depecker *et al.* 2001) studied 14 buildings conceived from the same basic cell. Their results established a linear dependence between the compactness and the heating loads in the context of a Paris's climate. At the same time, by working on the building's form and structure optimization, (Jedrzejuk and Marks 2002) showed that compactness is a good market indicator of energy-performance. Yet, in 2003, (Pessenlehner and Mahdavi 2003) detailed the limits of the compactness parameter and highlighted the influence of another one: the transparency, including glazing area and orientations. Recently, (Catalina *et al.* 2008) completed these works by coupling three parameters (compactness, inertia and glazing distribution) in order to assess their combined influence on heating loads. To pursue these conclusions, (Ourghi *et al.* 2007), and more recently (Al-Anzi *et al.* 2009), tried to find a relationship among all these parameters and energy usage. Thanks to a linear regression, they proved that there is a direct dependence between the heating loads, the compactness, glazing area and glazing type. Some papers provide a more complex analysis. For instance, (Adamski and Marks 1993, Jedrzejuk and Marks 1994) simultaneously considered several geometric variables in their studies, like wall lengths, building height, walls angles, window sizes and the thermal resistance of external walls. In 2008, (Zsalay 2008) used geometric height parameters (including floor area, the number of levels, the perimeter-to-floor area ratio, etc.) to generate a large building sample considering different building typologies. More recently, (Yi and Malkawi 2009) did not restrict their study to bow or simple forms. Thanks to the integration of advanced simulation and optimization algorithms, they developed a performance-based form making optimization method.

Studies trying to find the optimal energetic form of a building are numerous. Yet, most of these consider only heating loads (sometimes cooling energy use). The model we propose considers all of the energy flows due to construction and usage of a building.

3. BUILDING ENERGY MODEL

This research aims to establish the impact of a building's size on its whole energy consumption. The input data is the

primary energy consumed during both the construction and the use of a building. The output parameters are the dimensions - or rather the dimensional ranges - of an energy-saving building. In this study, only residential buildings are considered.

3.1. The basic building

For this study, the basic building has a simple plane-parallel form (Figure 2). Amongst the three building's dimensions, the depth p is fixed to 7 meters, so that the central part of the building is illuminated with daylight. This value has been chosen from daylight studies conducted by (Avouac *et al.* 2009), illustrated with Figure 3. The length L and the height h are the output variable parameters. The height depends on the number of levels n . The average floor-to-ceiling height is 2.5 meters. For optimal daylight comfort, the window-to-floor area ratio is around 17%, in accordance with French energy policy. An increased ratio can be evaluated with a sensitivity analysis. Concerning the thermal performance of the model envelope, the floors are made up of 200 mm heavy weight concrete that provides thermal inertia. The external walls are made up of wooden panels insulated with 240 mm of wood fiber insulation, while the roof is made up of 200 mm heavy weight concrete (like the floors) lined with 240 mm of wood fiber insulation. The overall U -values are 0.16 W/m².K and 0.235 W/m².K for, respectively, the external walls and the roof. The windows are double-glazed, with a U -value of 2.68 W/m².K and a solar heat gain coefficient of 0.5.

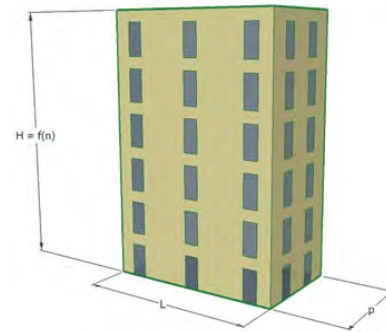


Figure 2. The basic building and its three dimensions.

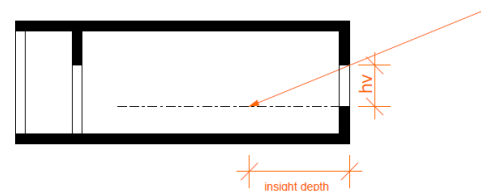


Figure 3. The daylight insight depth.

3.2. Methodology

The model draws up the energy balance of a building: it includes its gains, its loads and also the storage phenomenon, during both the construction and the use of a building. These flows are shown in Figure 4. To improve its energy balance, the building supports photovoltaic panels that produce electricity, and thermal panels that produce thermal energy. As shown in Figure 5, active panels are vertically and horizontally installed on the South front and on the roof.

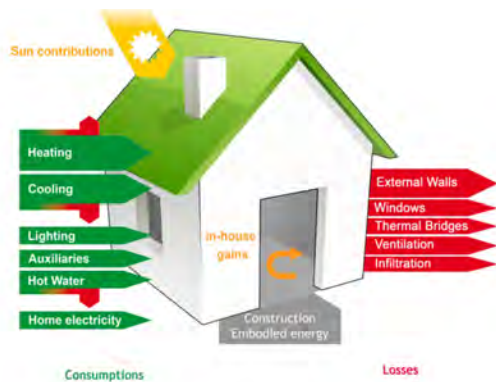


Figure 4. The building's energy flows.

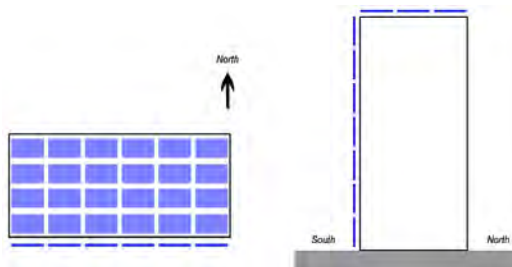


Figure 5. The solar panels distribution in plane view (on the left) and in side view (on the right).

3.2.1. Three case studies

The balance is performed monthly, in the case of a Grenoble climate. Three cases are considered:

- The regulatory case: the balance includes the five energy loading posts defined by the 2005 French energy policy (heating, cooling, hot water, lighting, and auxiliary power unit electricity). The auxiliary power unit includes electric appliances that provide ventilation, heating and cooling;
- The building and its own consumption: the model estimates every consumption flow that is mainly induced by the design of the building, from its construction energy costs to its lifetime energy consumption: it includes the materials embodied

energy, heating, cooling, lighting, auxiliary power unit electricity and elevator electricity;

- The building and its users: on top of the previous case, the model assesses the consumption due to the inhabitants' behavior (hot water, appliance electricity, etc.). Thus, the model aims to evaluate the users' influence on the global energy balance of a building.

For each case, the model includes both solar contributions (active and passive) and in-house gains (due to appliance utilization and metabolic activities).

3.2.1. Appliances and systems hypotheses

The model utilizes typical residential space occupancy patterns and schedules, given by the French energy policy and the *INSEE* statistics. These patterns concern the occupancy scenarios and the appliance and systems functioning. They are listed in Table 1. The model building and its whole components shelf life are theoretical: 50 years for the structure, 30 years for the façade, and 20 years for the photovoltaic panels.

Household size	4 persons/apartment
A dwelling size	66 m ²
Lighting power density	8 W/m ²
Heating system yield	0.81
Ventilating system yield	0.8
Air renewal flow	75 m ³ /hour.apartment
Thermal panel power density	450 kWh/m ² _{pan} .year
Photovoltaic panel power density	135 kWh/m ² _{pan} .year
(South oriented, 30°-tilted against the horizontal position)	

Table 1: Building model basic features

3.2.2. Energy and surfaces: reference units

Concerning the reference surface, the model first considers the benchmark surface used by the French building energy performance diagnosis, that is to say the *SHON* ("Surface Hors Oeuvre Nette"), whose equivalent is the net floor area. This surface counts, on top of the living surfaces, the surfaces filled by the external walls. In a later study, the model will weigh the livable surface, which represents the heated volume and does not count external walls (that can have various thicknesses). Moreover, to be more relevant, the model uses primary energy (E_p) instead of final energy (E_f): contrary to final energy, primary energy considers the way the energy is produced. The model uses E_p -to- E_f conversion ratios used by the French energy policy (electricity: 2.5; fossil fuel: 1; wood: 0.6).

3.2.3. Description of the calculation method

The energy balance of the model is drawn up from static equations (*ie* energy equations in steady state), varying according to the length and the height of the building. These equations are deduced from French and European norms, and from the French energy policy (CSTB 2007). These standards (available on Afnor Website) and their calculation methods are the following: for heating loads, the European and French norm (NF EN 832 1999) calculates the building's heating flux, from thermodynamics equations. These equations calculate the building conductive heat transfer coefficient L_D as follows: $L_D = \sum_i A_i \cdot U_i + \sum_k l_k \cdot \Psi_k$, where A_i is the external walls surface, U_i is the wall's thermal transmittance, l_k is the length of the thermal bridges and Ψ_k their thermal transmittance. For lighting, we use the European and French standard (NF EN 15193 2007), which considers two rates that weigh lighting consumption according to the switch type and the orientation of glazing. Finally, for hot water loads and auxiliary power unit electricity, we use the French Energy policy. Considering static equations induces a huge simplification of the model, and thus a margin of error, in particular concerning dynamic phenomena like energy storage. However, inertia is taken into account thanks to the calculation of the time constant, according to the standard (NF EN 832 1999). Moreover, in a final part, this calculation method is compared to two energy simulation software packages (PHPP and Comfie-Pleiades), in order to assess the error induced by the simplification and to validate the model.

3.3. Discussion of results

In this section, selected results concerning either the building's energy consumption or its energy balance are presented. Four cases are presented: the monthly unit regulatory consumption, the monthly unit regulatory energy balance, the monthly unit whole building and its users' energy balance, and finally the annual unit whole building and its users' energy balance. For each case, the unit primary energy (on the ordinate axis) varies according to the building's length L (on the right abscissa axis of the graphic, from 7 meters to 400 meters) and to its level number n (on the left abscissa axis, from 1 to 100). The unit of this energy is kWh_{ep}/m²_{SHON}. When negative, energy is consumed. When the balance is positive, the building produces more energy than it consumes. In the first three parts, graphs show monthly results, for January and July, in order to

compare winter and summer energy balances. The other monthly results are also available. Thanks to detailed monthly results, designers have enough information to design energy-saving buildings: by meeting December exigencies (that is the coldest month in the year), the building is inevitably energy-saving during the other months. However, considering only the annual energy balance does not show the monthly energy efficiency of a building. Annual results are presented in the fourth part of this section, for the whole building and its users' case. Thus, monthly and annual results give designers needed information to design an energy-saving building.

3.3.1. The monthly regulatory consumption

The first results concern the regulatory case: they present the unit regulatory consumption (primary energy), according to the length L and the height n of the building. The five loading posts defined by the French energy policy are considered (heating, cooling, hot water, lighting, and auxiliary power unit electricity). The model also takes into account the passive gains (from sun, appliances and metabolism) in order to reduce heating needs. No active contribution (thermal or photovoltaic) is taken into account.

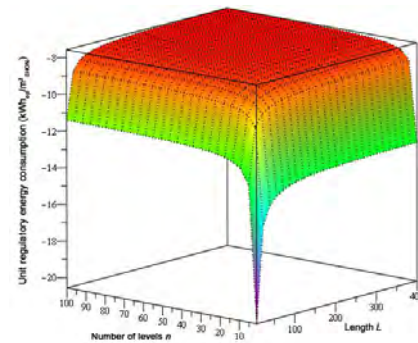


Figure 6. Evolution of the building's unit energy consumption according to its levels number n and its length L , for January.

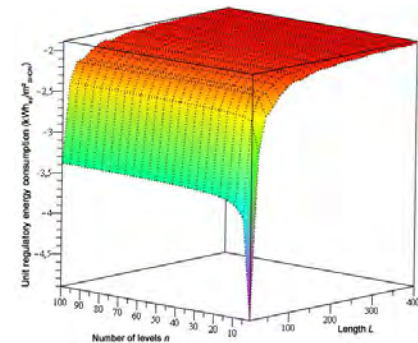


Figure 7. Evolution of the building's unit energy consumption according to its levels number n and its length L , for July.

Figures 6 and 7 illustrate the building's unit regulatory consumption (primary energy divided by the net floor area, on the ordinate axis) according to its number of levels n (on the left abscissa) and its length L (on the right abscissa) for January and July respectively. These two figures show that the bigger a building is, the more reduced its unit consumption is. This concurs with the literature review conclusions: the more compact a building is, the less it consumes. That is why the curve sharply decreases nearing the littlest building ($n = 1$ level and $L = 7$ meters). Moreover, the difference between the January and July graphs is due to passive solar gains that are soaked up by the windows (situated on the longer South-oriented façade). Since solar radiation increases during the summer, passive solar gains are higher during July than in January.

3.3.2. The monthly regulatory energy balance

When the model adds active gains, there is a trend reversal. Figures 8 and 9 show the evolution of the building's unit energy balance (primary energy divided by the net floor area, on the ordinate axis), according to its number of levels n (on the left abscissa axis) and its length L (on the right abscissa axis) for January and July respectively. This balance includes the five regulatory loading posts (heating, cooling, hot water, lighting, auxiliary power unit electricity), corrected by passive gains (from sun, appliances and metabolism). On top of this, it also takes into account active thermal and photovoltaic gains.

Contrary to the previous case (where only consumption was considered), in order to have an optimal energy balance, the building should not be huge. In particular, it should not be higher than 15 levels, *ie* 40 meters. Beyond this, its energy balance is negative and the building consumes more energy than it produces. However, the length of the building has little impact on its energy balance. The explanation of these facts is the following: active solar panels are installed on the roof and the South façade of the building. When the length of the building increases, its roof surface increases too and can carry solar panels that produce energy. On the contrary, the roof surface is the same however high the building is. Thus, despite the increase of the South façade surface, the energy production is not enough to balance the consumption.

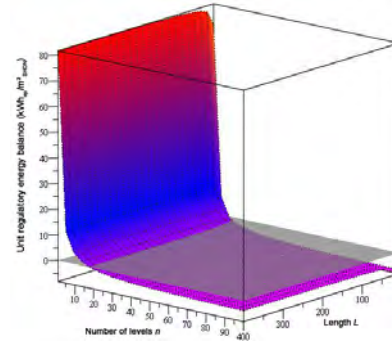


Figure 8. Evolution of the building's unit energy balance according to its levels number n and its length L , for January.

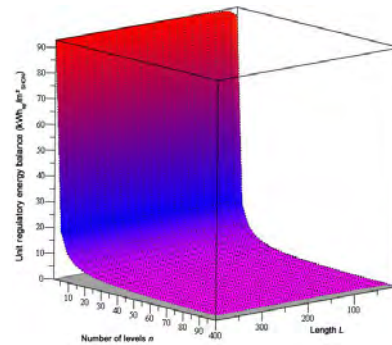


Figure 9. Evolution of the building's unit energy balance according to its levels number n and its length L , for July.

3.3.3. The monthly whole building and its users' energy balance

This trend increases when the model considers the monthly whole consumption of the building and its inhabitants. Figures 10 and 11 show the building and its users' energy balance (primary energy divided by the net floor area, on the ordinate axis), according to its number of levels n (on the left abscissa axis) and its length L (on the right abscissa axis) for January and July, respectively.

In this case, the whole energy loading posts are taken into account, from the construction (materials embodied energy) to the users' consumption (hot water and appliances electricity) including the building's own consumption (heating, cooling, lighting, auxiliary power unit electricity, and elevators electricity). Passive gains (due to solar, appliance and metabolic activities) are taken into account to reduce heating needs. Active thermal and photovoltaic gains are also considered.

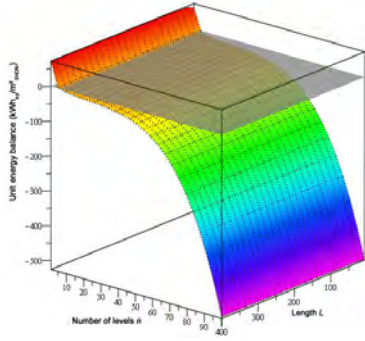


Figure 10. Evolution of the whole building and its users' energy balance according to its levels number n and its length L , for January.

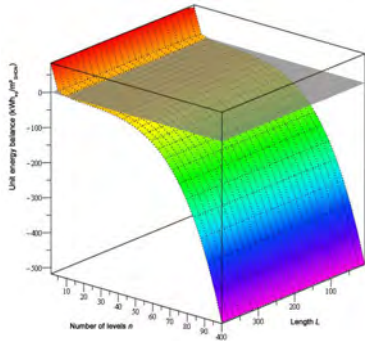


Figure 11. Evolution of the whole building and its users' energy balance according to its levels number n and its length L , for July.

In this last case, the trend is the same as previously, with a staggering decrease concerning higher buildings, from 40 to 100 stories. This is due to the materials embodied energy: high-rise buildings need more materials to withstand wind and earthquake forces. According to the graphs, to have a zero energy balance, the building should not be higher than 8 levels, *ie* 22 meters. As seen previously, the length has little influence on the building and its users' energy balance.

3.3.4. The annual whole building and its users' energy balance

The annual energy balance is given by adding up every monthly energy balance from January to December. Figure 12 illustrates the building and its users' energy balance (primary energy divided by the net floor area, on the ordinate axis), according to its number of levels n (on the left abscissa axis) and its length L (on the right abscissa axis), for the whole year. As in previous calculations, every energy flow is taken into account: consumption (materials embodied energy, hot water, appliances electricity, heating, cooling, lighting, auxiliary power unit electricity, elevators electricity), passive gains (solar, appliance and metabolic activities), and active thermal and photovoltaic gains.

The annual result presents surface curves similar to those of the monthly results. Besides, the “optimal” stories limit of an energy-saving building is quite the same as seen previously. From 1 to 8-10 levels, the building is energy-saving. From 10 to 40 levels, its unit consumption is higher than its unit energy production, but with a very slight decrease. Beyond 40 levels, the consumption decreases become staggering, because of the materials embodied energy. Moreover, the length has little influence on the building and its users' energy balance.

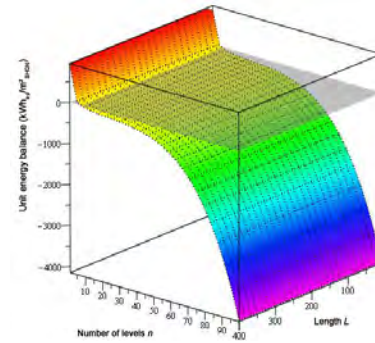


Figure 12. Evolution of the annual whole building and its users' energy balance according to its levels number n and its length L .

Thus, considering these four analyses, this static model established a link between the dimensions of a building and its whole energy balance (gains – consumption, including storage). From these monthly and annual results, designers have two possible objectives: either they want to design a monthly energy-saving building or they plan to design a building that is energy-saving the whole year. In the first case, designers need only the results of December that is the coldest month in the year: by answering December exigencies, the building is thus energy-saving during the other months. In the second case, designers need monthly detailed information. However, considering this study's results, there is little difference between the annual and monthly “optimal” levels limit. This is probably due to the calculation method of the annual energy balance (by adding up monthly results, instead of doing an annual calculation). The next step of this study is to validate this simplified model, based on static equations.

4. VALIDATION OF THE MODEL

To assess the model and its calculation method, it is necessary to check its results and to compare them to other energy simulation software results. Two software applications are chosen: *PHPP* (Passive House Planning

Package) and *Comfie-Pleiades*. *PHPP* is an *Excel* spreadsheet thermal calculation tool, based on the *PassivHaus* label's demands. It only applies to houses. Like the model, it uses steady state equations and calculates the house time constant to consider dynamic phenomena. *Comfie-Pleiades* (Salomon *et al.* 2005) is a thermal simulation software that calculates the thermal behavior of a building and its different thermal zones. Applied to houses and collective buildings, it uses dynamic equations that include inertia phenomena.

4.1. Validation methodology

To compare the model's results to those from the two software applications, we propose to apply them to a case study: a house whose characteristics (geometry and occupancy) are detailed in Table 2. The house is situated in Grenoble in France. As *Comfie-Pleiades* mainly gives thermal results about heating, cooling and passive gains, only these three energy balance components are presented here.

Number of levels	1 level
Length	10 meters
Depth	7 meters
Size	66 m ²
Household size	4 persons

Table 2: The case study: characteristics of the house

4.2. Discussion of results

Figure 13 illustrates the comparison of the simplified model to the *Comfie-Pleiades* and *PHPP* software, applied to the previously defined house. These results concern annual heating and cooling loads, as well as annual useful passive gains. Useful passive gains are internal (appliances and metabolism) and passive solar gains, corrected by the storage phenomenon. This graph shows that concerning heating and cooling loads, the model's results are close to those from *PHPP* and *Comfie-Pleiades*. Concerning heating loads, there is 3.04% difference between the model and *Comfie*, and 8.26% between the model and *PHPP*. Concerning cooling loads, the respective errors are 7.6% and 8.85%. These first results validate the simplified model concerning the calculation of heating and cooling thermal loads.

Concerning annual useful passive gains, there is little difference between the simplified model results and those from *Comfie-Pleiades*. The difference is 4.06%. However, between the model and *PHPP*, the difference is close to

80%. The explanation could be the following: *PHPP* uses an all-inclusive method for the calculation of internal gains (due to metabolic and appliance activities) and considers only global solar radiation for the calculation of passive solar gains. On the contrary, the model calculation methods for internal and passive solar gains are quite a bit more detailed (even if they do not consider dynamic phenomena). Concerning internal gains, the model separately considers the appliances and the metabolism contributions; and in the case of solar gains, it separates direct, diffuse and reflected radiation. Considering the little difference between *Comfie-Pleiades* and the model's results, the model is considered valid concerning useful passive contributions.

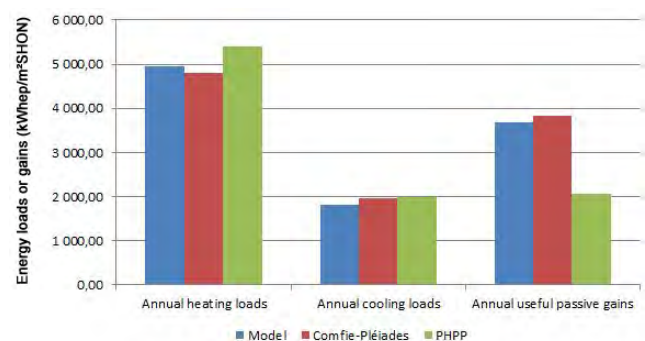


Figure 13. Comparison of the three calculation methods' results concerning heating, cooling and passive gains

This first comparison is quite positive concerning the relevance of the simplified calculation method developed in this paper. However, a next step proposes to complete this first analysis by two additional comparisons on two collective buildings: a high-rise building ($n = 15$ levels and $L = 30$ meters) and a long-spread building ($n = 5$ levels and $L = 56$ meters). These two additional studies will strengthen (or not) the conclusions of this paper. They will be published in a later paper.

5. CONCLUSIONS

This paper presents a simplified but complete calculation method, aiming to assess the whole energy balance of a building, according to its length L and its number of levels n . The results of this detailed model validate the conclusions of the literature review concerning the link between compactness and energy: the more compact a building is, the less unit energy consumption it exhibits. However, when considering the whole energy flows due to the construction and the functioning of a building and its inhabitants' behavior, there is a trend reversal. In this case, the curve

shows staggering decreases for higher buildings due to the materials' embodied energy. Thus, to be energy-saving, the building should be lower than 8 stories, *ie* 22 meters. Another remark is the following: the length has little impact on a building's whole energy balance in this study. This is due to active solar production, installed on the roof and the South façade of the building. When the length increases, the roof surface increases and can carry solar panels that produce energy. On the contrary, the roof surface is the same however high the building is. Finally, these results and the model have been preliminarily validated based on a comparison to two energy simulation softwares (*PHPP* and *Comfie-Pleiades*), in particular concerning heating and cooling loads in the case of a single house. Additional comparisons are needed to further validate these conclusions.

Moreover, as things stand at present, this paper cannot give a ruling on the presumption that density might not be sustainable and entirely positive for occupants. From this perspective, this “morpho-energetic” study is part of a broader study considering the morphology of energy-saving cities.

Acknowledgements

The simplified calculation model presented in this paper benefits from comments and advice from CSTB engineers and researchers, from UR AE&CC researchers and from ExnDo engineers. We would like to acknowledge them.

References

- ADAMSKI M., AND MARKS W. 1993. MULTICRITERIA OPTIMIZATION OF SHAPES AND STRUCTURES OF EXTERNAL WALLS OF ENERGY CONSERVATION BUILDINGS. ARCHIVES OF CIVIL ENGINEERING. VOL.39. ISS.1. 77-91
- AL-ANZI A., SEO D., AND KRARTI M. 2009. IMPACT OF BUILDING SHAPE ON THERMAL PERFORMANCE OF OFFICE BUILDINGS IN KUWAIT. ENERGY CONVERSION AND MANAGEMENT. N°50. 822-828
- ARANTES, L., CHALENCON, E., AND HADJ-HASSINE, Y. 2009. TOURS ECONOMES, NOUVELLES FORMES D'HABITER EN CENTRE URBAIN DENSE. FINAL YEAR REPORT. ENSAG, GRENOBLE, FRANCE, 189 P.
- AVOUAC P., FONTOYNONT M., AND PERRAUDEAU M. 2009. GUIDE PRATIQUE DE L'ÉCLAIRAGE NATUREL. 150 P.
- CATALINA T., VIRGONE J., AND BLANCO E. 2008. DÉVELOPPEMENT D'UN OUTIL POUR L'OPTIMISATION DES BESOINS D'UNE HABITATION ET LE CHOIX DES SYSTÈMES À ÉNERGIE RENOUVELABLE. CONGRÈS IBPSA. LYON, FRANCE, 8 P.
- CSTB (CENTRE SCIENTIFIQUE ET TECHNIQUE DU BÂTIMENT), 2007, RÉGLEMENTATION THERMIQUE 2005, GUIDE RÉGLEMENTAIRE
- DEPECKER P., MENEZO C., VIRGONE J., AND LEPERS S. 2001. DESIGN OF BUILDING SHAPE AND ENERGETIC CONSUMPTION. BUILDING AND ENVIRONMENT, 36, 627-635.
- JEDRZEJUK H., AND MARKS W. 1994. ANALYSIS OF THE INFLUENCE OF THE SERVICE LIFE AND SHAPE OF BUILDINGS ON THE COST OF THEIR CONSTRUCTION AND MAINTENANCE. ARCHIVES OF CIVIL ENGINEERING, VOL.40. ISS.3/4. 507-518
- JEDRZEJUK H., AND MARKS W. 2002. OPTIMIZATION OF SHAPE AND FUNCTIONAL STRUCTURE OF BUILDINGS AS WELL AS HEAT SOURCE UTILIZATION. BASIC THEORY, BUILDING AND ENVIRONMENT. VOL. 37. ISSUE 12. 1379-1383
- NF EN 832. 1999. PERFORMANCE THERMIQUE DES BÂTIMENTS, CALCUL DES BESOINS D'ÉNERGIE POUR LE CHAUFFAGE, BÂTIMENTS RÉSIDENTIELS. NORME EUROPÉENNE, NORME FRANÇAISE, ISSN 0035-3931. 62 P.
- NF EN 15193, 2007, EXIGENCES ÉNERGÉTIQUES POUR L'ÉCLAIRAGE, NORME EUROPÉENNE, NORME FRANÇAISE, ISSN 0335-3931. 69 P.
- OURGHI R., AL-ANZI A., AND KRARTI M. 2007. A SIMPLIFIED ANALYSIS METHOD TO PREDICT THE IMPACT OF SHAPE ON ANNUAL ENERGY USE FOR OFFICE BUILDINGS. ENERGY CONVERSION AND MANAGEMENT. 48. 300-305
- PESSLENHNER W., AND MAHDABI A. 2003. BUILDING MORPHOLOGY, TRANSPARENCY AND ENERGY PERFORMANCE. EIGHTH INTERNATIONAL IBPSA CONFERENCE. 1025-1032.
- SALOMON T., MIKOLASEK R., AND PEUPORTIER B., 2005, OUTIL DE SIMULATION THERMIQUE DU BÂTIMENT, COMFIE, SFT-IBPSA
- SZALAY Z. 2008. MODELLING BUILDING STOCK GEOMETRY FOR ENERGY, EMISSION AND MASS CALCULATIONS. BUILDING RESEARCH AND INFORMATION. VOL.36. N°6. 557-567

A methodological study of environmental simulation in architecture and engineering. Integrating daylight and thermal performance across the urban and building scales.

Peter Andreas Sattrup¹, Jakob Strømmand-Andersen²

¹Royal Danish Academy of Fine Arts
School of Architecture
Philip de Langes Allé 10,
DK-1435 Copenhagen K, Denmark
peter.sattrup@karch.dk

²Technical University of Denmark
Department of Civil Engineering
Brovej 118,
DK-2800 Kgs. Lyngby, Denmark
jasta@byg.dtu.dk

Keywords: Integrated design methodology, holistic design, architectural theory, design process, energy optimization, environmental simulation

Abstract

This study presents a methodological and conceptual framework that allows for the integration and creation of knowledge across professional borders in the field of environmental simulation. The framework has been developed on the basis of interviews with leading international practitioners, key theories of environmental performance in architecture and engineering, and a range of simulation experiments by the authors. The framework is an open structure, which can continuously be renewed and contributed to by any author.

The value of the framework is demonstrated, using it to map a series of simulation studies, emphasising the multidimensionality of environmental performance optimization. Clarifying the conceptual interconnectivity between architecture and engineering, - agency and physics, - not only enhances communicative power and the dissemination of knowledge, but becomes instrumental in pointing out the need for improving metrics, software and not least the performance of the built environment itself.

1. INTRODUCTION

Though environmental simulation software has been around for decades, developed and used mainly by engineers, it has only recently become widely available to

architects without an extreme specialization in physics and computation. Following this introduction of technology into the field of architecture, comes a stimulating shift of attention in terms of the aims of simulation research: that of studying the relations between different spatial scales, exploring form and material organization as means to produce desirable human environments, rather than the singular optimization of specific technical subsystems. Architecture can in itself be considered an open system of environmental technology that is not just technical but informational, social and cultural too.

A problem in current energy optimization in architecture and engineering is a certain blindness towards the multiple facets of performance of each part of the complex systems of built environments. Buildings failing to use form and materials to direct nature's forces for the benefit of the occupants get wrapped up in sub-optimized comfort and energy delivery systems to compensate for their lack of environmental qualities. Technical systems, that is, that have high embodied energy and a much shorter lifespan than the building structure and its skin, and as a consequence a higher detrimental impact on the environment.

There is a need for a holistic view and an integrated approach that emphasises that the layers, scales, components, materials, uses etc. of a building or a built environment plays multiple roles, and must be understood in temporal dimensions that include the day to day, seasonal,

yearly and lifetime dimensions. This holistic view should embrace the multidisciplinary of the design professions, and establish a common conceptual framework.

The research questions behind this paper are:

How can a conceptual framework be devised, that allows for the synthesis and integration of environmental performance information in the built environment across the architecture/engineering professions? How can the dual aspects of operational and embodied energy in architecture be linked? How can the spatial and temporal dynamics of the performance of the built environment be highlighted, to improve communication, software and metrics in environmental simulation?

To answer these questions, 3 principal approaches are used, including interviews with leading practitioners in architecture and engineering, a literature review and the simulation experience of the authors – an architect and an engineer. The framework devised can potentially be used to guide future research, mapping the impact of design variables on environmental performance, and act as a support when establishing the decision hierarchies necessary in any design project regardless of scale, to meet the demands of rapid decision making and efficiency of solutions required today.

2. METHODS AND METHODOLOGY

A distinction between methods and methodology needs to be made. Methods are ways of using tools or techniques using a prescribed procedure towards a certain aim. Following a tutorial of how to do a daylight analysis could be an example. Methodology instead considers multiple methods, critically examining the assumptions behind them and examines the interrelationships between output from different methods. Though architects rarely claim to follow specific design methodologies, a new attention to design methods and processes is emerging in order to deal efficiently with the realization that even the first sketches potentially carries a strong impact on the energy and environmental performance of the design. In particular engineers have begun to promote ‘integrated design processes’ in which the architect-engineer collaboration is shifted forward into the initial design phases, instead of the traditional process of engineers following up on designs already elaborated by architects. This could possibly result in some rivalry for position and influence on design among

the professions, but the view taken in this paper is that collaboration is necessary and beneficial for the overall good of the built environment. The paper is a result of a collaboration of an architect and an engineer, and the methodological framework proposed here is intended to clarify the basis for the use of environmental simulation in the early design phases as a common platform across the professions.

The methodological framework presented in this paper, is in itself the outcome of different research methods: The hypothesis that architectural scale is a key factor in energy efficient design, connecting both integrated design processes, operational efficiency and lifecycle analysis, was derived from interviews with leading practitioners in architecture and engineering from the offices of Foster and Partners (Behling and Evenden 2009), Baumschlager & Eberle Architekten (Eberle 2008) and Transsolar (Schuler 2008). The theory behind the hypothesis was developed through a literature review. A series of simulation studies using Copenhagen as a reference were carried out by the authors to test the hypothesis in regards to solar access, daylight, thermal and energy performance in urban and basic building design, some of which are presented in the demonstration case below.

3. ARCHITECTURAL PRACTICE

While their architectural interests and formal expression differ significantly, the architect offices of Foster and Partners and Baumschlager and Eberle share the notion based on their experience in design and built work, that the greatest environmental impacts come with design decisions taken at the scale of the city, and that impacts decrease with minor scale decisions. Argued this way, optimization of the basic formal and material properties of a design, takes priority compared to the optimization of technical service systems, - conveniently weighting the influence of the architect’s design responsibilities over those of the engineers. Both offices employ specialists working with simulation, and have developed tailor-made software applications to suit the offices’ workflows. Fig. 2 shows a diagram by Foster and Partners presented with the Masdar project, expressing the notion that design decisions taken at the largest scales of a project impact the environment the most, and are inverse proportional to the costs of the solutions. While ‘environmental gain’ includes other factors than energy, - it could be interpreted aesthetically, - energy plays the dominant role in the diagram’s highlighting of

passive and active measures and the implementation of renewable energy systems.



Figure 1: Foster and Partners: Scale, systems, costs and environmental gain diagram.

In the design approach of Dietmar Eberle attention towards the durability of the different layers that constitute a building govern a hierarchy of design decisions. Each layer: place, load bearing structure, envelope, programme and materiality carry weight in the design process according to their relative permanence, so as to preserve the resources invested in them. Where Foster and partners employ advanced form to harness the benefits of nature’s forces, Baumschlager and Eberles approach highlights architecture’s generality and its adaptivity over time as a sustainable strategy. Eberle explicitly states the need for design methodology to integrate knowledge in the design process (Simmendinger and Schröer 2006).

But what lies behind these assumptions and notions?

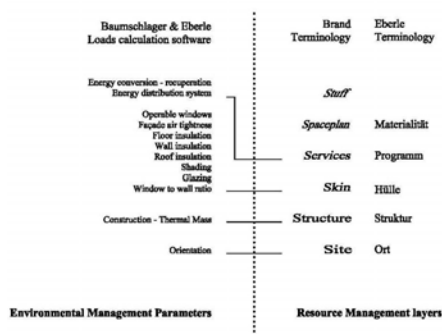


Figure 2: Comparison of design variables in environmental management to shearing layers and the layers of the design theory by Dietmar Eberle of Baumschlager & Eberle Architekten.

4. ARCHITECTURAL THEORY – RESOURCE AND ENVIRONMENTAL MANAGEMENT

Two key references seem to have spawned several subsequent architectural research enquiries. While research

has developed since then, it is nevertheless the original concepts that remain fundamental to sustainability in architecture. Taking a step back does not ignore the progress of knowledge since then, rather it allows us to identify design issues at the macro level of sustainable design that must be addressed simultaneously and coherently by designers, including specialists working with environmental simulation.

In the book ‘The Architecture of the Well-tempered Environment’ (Banham 1984) ‘structure’ and ‘power’ solutions are defined as the two fundamental ways to mediate the environment through the use of resources. Structure solutions are resources invested in built space that is able to ‘conserve’ energy (eg. heat). Power solutions are energy resources used to ‘regenerate’ environmental condition artificially, as when burning a timber resource to provide heat rather than burning it. The ‘selective’ mode in between, is the building features that allows the occupant to choose among environmental stimuli, natural or artificial. The distinctions are not exclusive, - many building components perform and can be used in different modes. Banham was able to set the stage for later introduction of the technical distinctions between embodied and operational energy, and the environmental performance associated with the building fabric, user behaviour and energy use through service systems.

The second key reference is ‘How Buildings Learn’ (Brand 1997). Drawing on the theory of the shearing layers, originally developed in forestry and ecology studies, Brand establishes the idea that buildings have metabolism, and that the rate of metabolism is connected to layers of scale and activities that change a building over its lifetime. The layers that Brand identify are Site, Structure, Skin, Services, Spaceplan and Stuff, - their sequence referring to their durability and expected lifetimes, - Site being the most durable, almost permanent condition governing a building and Stuff, - furnitures and the like - being the most ephemeral with the highest metabolic rate (Fig. 3). It is perfectly possible for parts of buildings to fulfil more purposes, though it should preferably be avoided to allow better adaptability in the long term. Similarly at the urban scale spatial, legislative, regulative and ownership layers with different permanence can be identified that frame the evolution of the city. (Fig. 4)

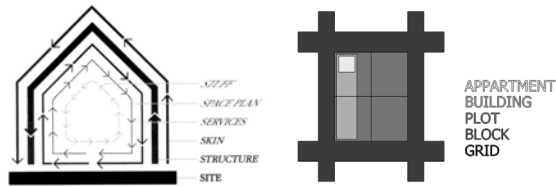


Figure 3: Brand: Shearing Layers. Organizing a building according to the permanence of its different functional layers becomes instrumental in the resource management of buildings' material lifecycle.

Figure 4: Sattstrup: At the urban scale, regulatory layers can be identified on the basis of the spatial, property and planning framework that governs the development of cities over time.

Brand identifies two ways of ensuring that a building achieves a long life – thus ensuring the maximum benefit of the resources and energy invested in its construction and maintenance – the ‘high’ way of investing a high cultural value in a building, and the ‘low’ way of ensuring the practicality of adapting the building to changing uses, by consciously using the shearing layers as a way to organize the building functionally and tectonically. This has spawned subsequent research enquiries in architecture aimed at minimizing the environmental impact of waste associated with buildings' materials and the embodied energy invested in them, through ecologically and lifecycle oriented approaches (Berge 2009), (Braungart and McDonough 2009).

5. A METHODOLOGICAL FRAMEWORK FOR ENVIRONMENTAL SIMULATION

What does Banham's Environmental Management and Brand's Shearing Layers have in common, how are they differentiated, - and how can they be linked?

Brand's Shearing Layers primarily addresses the long term use of resources – what Banham terms solutions of Structure. But the layers also differentiate between different building scales, and the uses associated with them, opening up a connection to Banham's secondary concepts of the conservative, selective and regenerative modes of environmental management. Shifting Banham's definition of the selective mode slightly, so as to specifically describe the selective behaviour of the occupant rather than the properties of building components, Brand's layers: Site, Structure and Skin can be specifically linked to the Conservative mode, and the Selective mode used to describe the occupants' behaviour regarding the operation of the Skin and the Services layers. Now several frameworks can be

defined that link the different scales surrounding a building project (Fig. 5).

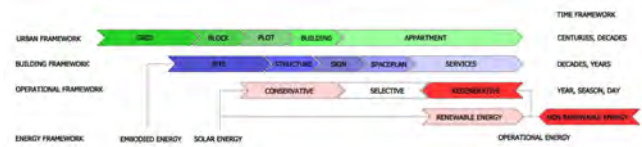


Figure 5: Connection of frameworks: Urban, Building, Operation, Time and Energy. The relative position of layers imply their relatedness.

Within the regulatory layers of the Urban Framework, we can use Brand's layers to describe the Building Framework, which again frames the Operational Framework, which we can describe using Banham's terms. Each of these operate at different time scales, so the Time Framework indicates the rate of change of the others: from the Urban Framework that can potentially last for centuries, to the daily rhythms of people in the Operational Framework. The Energy framework describes how embodied energy is stored in the fabric, solar energy potential for heating and lighting is mediated through the urban and building layers, and how operational energy is dispersed through the service systems. By organizing these visually it is made clear how the Frameworks influence each other, so as to create an awareness of the multiple aspects of the built environment that designers need to navigate to create truly environmentally and culturally sustainable buildings. The Spaceplan layer involves the organization of the building's programme, and is connected to the patterns of occupation and operation. The Services layer is associated with the energy loads for heating, cooling and lighting and the process of optimizing the plant and distribution systems. This categorization allows us to identify six domains of performance optimization: *Form, Material, Programme, Operation, Loads* and *Service Systems*. Each has different design variables that interact as complex systems and sometimes overlap between domains (Fig 6 & 7).

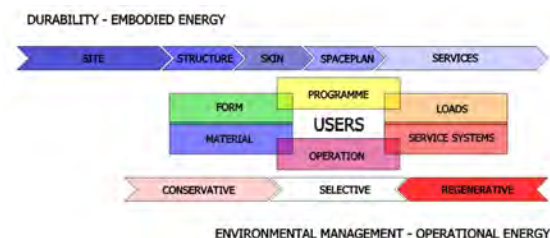


Figure 6: Design domains in between the Building and Operational framework. The domains can be described using specific design variables.

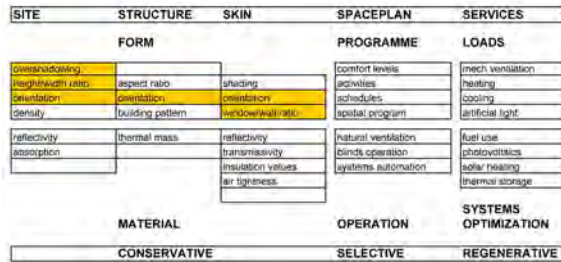


Figure 7: Design domains with detailed design variables. Variables can be added according to design or simulation interest.

6. DEMONSTRATION CASE – INTEGRATING DAYLIGHT AND THERMAL PERFORMANCE ACROSS THE URBAN AND BUILDING SCALES

In the following demonstration case Site, Structure and Skin layers are investigated for the impact of Form on energy use, differentiating thermal performance according to Conservative, Selective and Regenerative modes of operation. The framework is used to map and interrelate a series of simulation studies undertaken by the authors. The aim of the studies is to clarify the following:

- 1) The impact of Form on the energy performance, investigating orientation and window size design variables of the Site, Structure and Skin layers.
- 2) Using the Conservative, Selective and Regenerative modes as conceptual and analytical tools to pinpoint the influence of Form and Material properties on the daylight and thermal performance related to Building Skin.

The studies focus on the integration of daylight and thermal performance tracing the impact of generic formal design decisions from the urban to the building scale, investigating how the temporal and spatial dimensions of solar access in the urban environment affect thermal and daylighting performance of apartments with different window to wall ratios. The climatic context of the study is Copenhagen (N56,E12) in Northern Europe, a climate that is marked by the relative scarcity of sunlight due to high latitudes, a predominance of overcast skies and the low solar altitude in the winter months. Sketch-up was used to create the models, which were exported to IES-VE for thermal, artificial light and energy analysis. Ecotect was used to analyse and visualize the spatial and temporal distribution of

solar radiation in the urban environment exporting the model for daylight autonomy analysis using DAYSIM. In all studies a design reference year (DRY) weather file for the city of Copenhagen was used. The material specifications are equal to the minimum current requirements in Danish Building regulations. See appendix.

6.1. SITE, STRUCTURE and SKIN

A first step in understanding the conditions of the Site was an analysis of the temporal and orientational distribution of radiation. A ‘solar rose’ was invented to visualize the yearly and seasonal radiation on vertical surfaces compared to the global radiation on a horizontal surface, as passive solar energy usually is distributed through vertical facades. Using the solar rose both seasonal and daily variations can be grasped at a glance, as the intensities are also connected to the time of the day. As can be seen, the intensity differs greatly, but due to the angle of incidence, some surprising facts are found: the solar potential on facades in spring is equal to that of summer, and offers a potential to shorten the heating season as temperatures have not risen yet. In Autumn and winter the low inclination of the sun means that the intensity of radiation on south facades can rival those of the yearly average though the exposure times are shorter, and the sensitivity to overshadowing in urban contexts increases greatly.

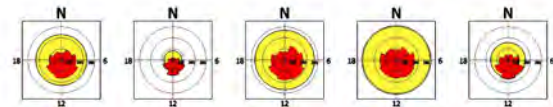


Figure 8: SITE: Solar roses, Copenhagen. From left to right: average hourly radiation on vertical (red) and horizontal (yellow) surfaces, - yearly, winter, spring, summer and autumn averages. Range 0-300wh/m2.

Previous studies by the authors examining solar envelopes (Knowles 1985) for the city of Copenhagen suggest that a maximum eaves height at 5 stories is advisable in dense urban districts at the same latitude, - a fact that corresponds very precisely to the actual densities of the inner city of Copenhagen. Above these densities, solar and daylight access are so restricted that denser urban patterns risk becoming unattractive, unless other attractions are associated with them.

To find out the seasonal intensity variations, the solar potential of the facades of a 5 story 50x50m urban perimeter

block was calculated. As can be seen (fig. 11) the patterns of overshadowing by the surrounding buildings are gradients of intensities with great directional and temporal variations.

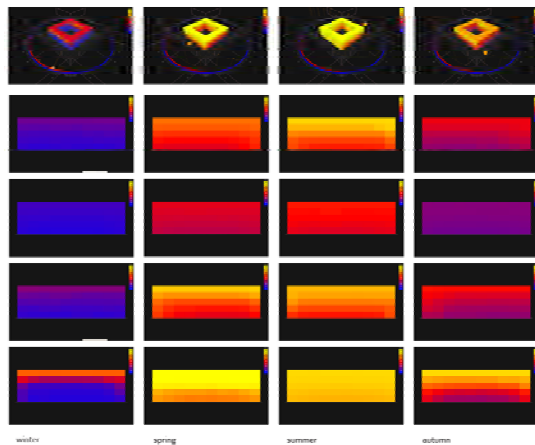


Figure 9 SITE+STRUCTURE & TIME: solar exposure on 5 story 50x50m perimeter block. Top to bottom: SE view, W, N, E & S facades. Left to right: winter, spring, summer and autumn exposure. Range 0-200wh/m2.

As the urban grid and planning regulations often limit the formal exploration on a given site, geometry and orientation can still be used by designers to increase or decrease the radiation intensity through working at the building Structure and Skin scales. Orientation is investigated as a design variable through either rotating the block 45 degrees or folding its skin, so as to increase solar intensity hitting glazed areas of the façade in the winter season /Fig. 10).

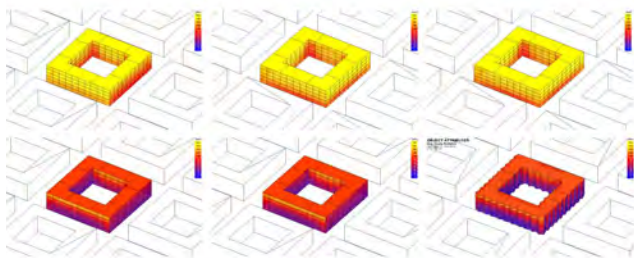


Figure 10: SITE+STRUCTURE+SKIN & TIME:

North/south, rotated 45degrees and North/South faceted façade block. Radiation levels during warm season April-Oct and cold season Oct-April. Radiation levels 0-200wh/m2.

The radiation levels are so low due to the high latitude and low sun angles in Copenhagen which cause overshadowing in winter, that only the top 3 stories can

pursue solar strategies for low-energy consumption with interesting local differences: South facing apartments have higher solar exposure in winter, but lower in summer than the others, due to the changing inclination of the sun-path. East-West facing apartments have high exposure in the summer and very little in the winter, which can be mediated using faceted facades, shifting the gains towards the season where they are needed. The rotated block has medium-high solar gains throughout the year when compared to the others. But changes to orientation carry very little weight on the overall energy demand, even given today's standard of construction. Heating demand for a 100m2 apartment with a window to wall ratio of 40% changes insignificantly when averaged over all 5 levels of the model, stabilizing at 44kwh/m2yr as the 3 bottom levels are totally overshadowed during winter at the urban density studied in this model. The 45 degree rotated block has a more even spread of the solar potential, more apartments benefit from the heat gains and a greater diversity of climatic situations and sunshine hours than in a north/south facing block, faceted or not.

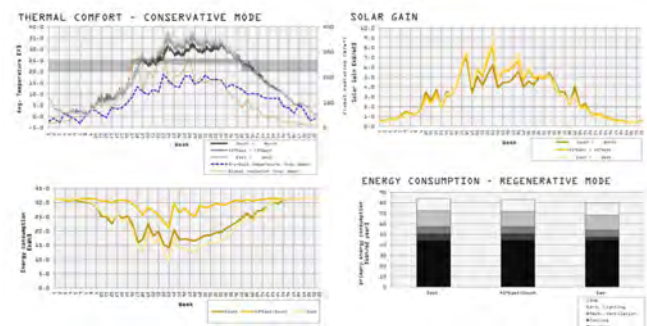


Figure 11 Thermal comfort, daylight, solar gains and Energy. North/south, rotated 45degrees and East/West facing buildings.

Surprisingly the rotated block does not get the energy savings for daylight that the high radiation levels would make one think, it performs much worse than the north/south and east/west oriented buildings. A careful examination using a sunpath tool reveals why: As the buildings are used for housing, the occupants are not at home during weekdays at the hours where the sun delivers its energy. In the morning and afternoon the sun-angles are so low that the main bulk of the building lies in shadows.

Linking the energy use calculation for artificial light to climate based daylighting metrics such as the Daylight Autonomy is not so straight-forward. Using IES-VE

radiance to set up an artificial light control system that switches on light when natural light levels fall below 200lux in the occupied hours applying a 30% switched-on percentage, is not quite the same, though it is climate based, as it is linked to the climate file's radiation data, converted to lighting. IES-VE automatically places the sensor point in the middle of the zone, (if one uses the thermal engine's control system for switching as is done here) when generic models such as the ones presented here are studied.

The true benefit from working with the orientation lies in the temporal dimension of the solar exposure seen in accordance with the building's rhythms of occupation. But as minimum insulation values will increase over the next decade, even the small increases in average radiation observed in this simulation are likely to carry a larger weight on comfort levels and energy use in future construction. Returning to the Urban Framework, it may well be worth opting for an optimization of solar potential through orientation though it carries little weight in the energy budget today. In the long term perspective of the Site, a 10% better solar potential which is more evenly shared among neighbours can prove a valuable asset as cities develop, building technologies are upgraded and social patterns change.

6.2. CONSERVATIVE, SELECTIVE and REGENERATIVE mode analysis.

Further investigating the performance of the Building Skin, the influence of different Window to Wall ratios was defined, using the same model properties as in the previous study. To be able to identify precisely the influence of the building fabric, the behaviour associated conditions and the systems energy loads on the thermal performance, the settings of the model were varied using the Conservative, Selective and Regenerative mode:

In the **Conservative** mode the empty building envelope is simulated. This allows a very accurate analysis of the influence of the Form and Material design domains on the thermal performance, as the influence of user patterns is excluded.

In the **Selective** mode the building is basically free-running, including internal gains from occupants and equipment and natural ventilation in summer. This adds the probabilistic user patterns of the Programme and Operation design domains to the model. No climatization is included.

In the **Regenerative** mode the building is fully conditioned, adding the influence of all three modes. Total primary energy use is calculated using fully dynamic IES-VE radiance climate based thermal and lighting simulation.

The practically unobstructed apartments at the 5th floor were subjected to a comparative study using the conservative, selective and regenerative modes to analyze their thermal performance. Some interesting facts emerge: The building fabric alone (conservative) is able to shorten the heating season by 6 weeks in spring and delaying it in autumn by 4 weeks totalling 2½ months when comparing the 20% window to wall ratio with 80%, though this comes with the risk of serious overheating unless measures are taken to limit summertime heat gains, as natural ventilation (Selective mode) is not sufficient, or cooling (Regenerative mode) will be necessary (Fig. 11).

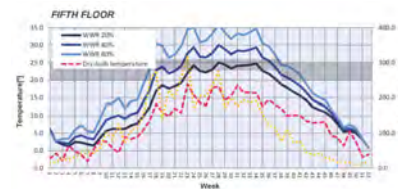


Figure 11 CONSERVATIVE: 5th floor apartment types with 20, 40 and 80% window to wall ratio. Thermal performance of the empty building envelope.

During winter, the average temperature performance favours large windows, showing that the higher conductive losses from larger windows can be balanced by the heat gains from the solar radiation, even though it should be noted that daily temperature swings are much more pronounced the larger the glass areas, a fact that is masked by the weekly averages shown by the graphs. The 40% wwr apartment is better balanced, and the selective mode shows, that the heating season can be shortened equally to 80% wwr, when the internal gains from the occupants are included in the energy balance (fig. 12).

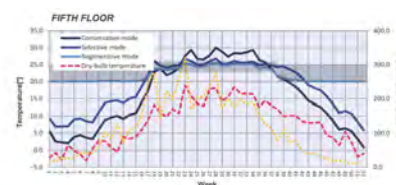


Figure12 CONSERVATIVE+SELECTIVE+REGENERATIVE: 5th floor apartment with 40% window to wall area. Thermal analysis

The daylight autonomy metric (Reinhart, Mardaljevic, and Rogers 2006), - that could be considered a ‘selective’ mode analysis in this context, - can be rendered using DAYSIM. It shows the yearly percentages of time where the light distribution levels are above a certain threshold deemed adequate for given tasks (fig. 13).

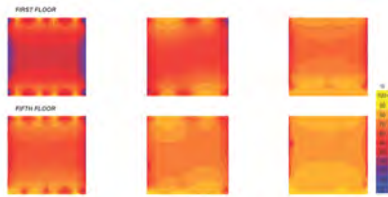


Figure13 SELECTIVE: DAYSIM Daylight autonomy renderings of the yearly percentage of time where daylight levels exceed 200lux during occupancy hours 6-18.

When compared to a likewise temporal analysis of the energy use for lighting, the connection between the two figures is hard to see. Though each method uses radiance to calculate the time that light levels are above 200lux in a sensor point, the spatial imagery of DAYSIM is more visually communicative of the spatial qualities of the light. Though the new climate-based daylight metrics are greatly superior to the daylight factor, the analytical control of light's temporal dimensions should be improved in simulation so as to be able to grasp and communicate more of this variation. (fig. 13).

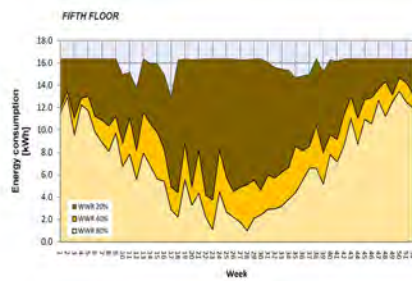


Figure13 REGENERATIVE: graph showing the temporal dimension of energy use for artificial lighting dependent on window to wall ratio.

7. CONCLUSION

A methodological framework was developed, derived from interviews with leading practitioners and key references from architectural theory. The framework establishes a holistic view and an integrated approach that emphasises that the layers, scales, components, materials uses etc. of a building or a built environment plays multiple roles, and must be understood in temporal dimensions that

include the day to day, seasonal, yearly and lifetime dimensions. The framework is structured according to a reinterpretation and expansion of Brand's and Banham's original concepts to differentiate and connect building performance analysis of the built environment, the influence of occupants behaviour and the optimization of service systems, showing ways to connect the areas of operational and embodied energy, environmental management and resource management.

The framework is used to map a series of studies ranging from the urban scale to the facades, integrating thermal and daylighting performance dynamically, while tracing the impact of the urban context on building performance. The mapping of this particular study, points out the need for future research in the ‘blank’ spaces of the framework: Specific studies across the boundaries of the operational/embodied energy fields, further investigations of the temporal potentials in climatebased daylighting metrics, and a continual evolution of conceptual clarifications that allows knowledge to be integrated and disseminated across professional borders.

References

- BANHAM, R. 1984. *Architecture of the Well-tempered Environment*. 2nd ed. Chicago University Press.
- BEHLING, STEFAN, AND EVENDEN. 2009. *Foster and Partners Interview by P. A. Sattrup. Green Architecture for the Future*.
- BERGE, BJØRN. 2009. *The ecology of building materials*. Architectural Press.
- BRAND, STEWART. 1997. *How Buildings Learn: What Happens After They're Built*. New edition. Weidenfeld & Nicolson.
- BRAUNGART, MICHAEL, AND WILLIAM McDONOUGH. 2009. *Cradle to Cradle*. Vintage.
- EBERLE, DIETMAR. 2008. *Arkitekt, Arkitektur og Ressourcer Interview by P. A. Sattrup. Arkitekten 11*.
- KNOWLES, R. L. 1985. *Sun, Rhythm, Form*. New edition. MIT Press,
- REINHART, CHRISTOPH F., JOHN MARDALJEVIC, AND ZACK ROGERS. *Dynamic daylight performance metrics for sustainable building design*. Leukos 2006, no. 3. Leukos (July): 1-25.
- SCHULER, MATTHIAS. 2008. *Paven af Integreret Design Interview by P. A. Sattrup. Arkitekten 12*.
- SIMMENDINGER, PIA, AND ULRIKE SCHRÖER. 2006. 'City and House – a Strategy Of Teaching Design'. In *The Complexity of the Ordinary*. Copenhagen: Royal Danish Academy of Fine Arts

Appendix

		CONSERVATIVE MODE	SELECTIVE MODE	REGENERATIVE MODE
CONSTRUCTION				
Opaque Construction				
External Walls	U-value	0.2 W/m ² K	0.2 W/m ² K	0.2 W/m ² K
Roofs		0.15 W/m ² K	0.15 W/m ² K	0.15 W/m ² K
Exposed floors		0.15 W/m ² K	0.15 W/m ² K	0.15 W/m ² K
Thermal capacity		140 kJ/(m ² K)	140 kJ/(m ² K)	140 kJ/(m ² K)
Thermal bridging coefficient		0.035 W/m ² K	0.035 W/m ² K	0.035 W/m ² K
Glazed Construction				
External Windows	U-value	1.5 W/m ² K	1.5 W/m ² K	1.5 W/m ² K
	g-value	0.6	0.6	0.6
Visible light normal transmittance		0.7	0.7	0.7
USE OF THE BUILDING				
People				
Internal heat gain		–	1.5 W/m ²	1.5 W/m ²
Lighting				
Lighting level			200 lux	200 lux
Maximum Power			6 W/m ²	6 W/m ²
Variation Profile			6am–9am & 3pm–10pm (weekday) 9am–10pm (weekend)	6am–9am & 3pm–10pm (weekday) 9am–10pm (weekend)
Switched-on-percentage			30 %	30 %
Dimming profile			Manual/on-off, (200 lux)	Manual/on-off, (200 lux)
Miscellaneous				
Maximum Sensible gain			3.5 W/m ²	3.5 W/m ²
Variation Profile			on continuously, 50 %	on continuously, 50 %
AIR EXCHANGES				
Infiltration				
Min Flow		0.13 l/s m ²	0.13 l/s m ²	0.13 l/s m ²
Variation Profile		on continuously	on continuously	on continuously
Natural ventilation				
Max Flow			0.9 l/s m ² , t > 25 °C	0.9 l/s m ² , t > 25 °C
Variation Profile			(week 19 – week 37)	(week 19 – week 37)
Mechanical ventilation				
Min Flow				0.3 l/s m ²
Variation Profile				on continuously
System specific fan power (SFP)				1.0 W/(l/s)
Vent. Heat recovery effectiveness				65%
Cooling efficiently				COP=2.5
HEATING AND COOLING				
Heating set point	winter	–	–	20 °C (on continuously)
Cooling set point				25 °C (on continuously)
Heating set point	summer	–	–	23 °C (on continuously)
Cooling set point				26 °C (on continuously)

Table 1 Simulation systems settings

Session 6: Augmented Reality

135 **Lifecycle Building Card: Toward Paperless and Visual Lifecycle Management Tools**

HOLGER GRAF, SOUHEIL SOUBRA, GUILLAUME PICINBONO, IAN KEOUGH, ALEX TESSIER and AZAM KHAN
Fraunhofer IGD

143 **3D Scans of As-Built Street Scenes for Virtual Environments**

NAAI-JUNG SHIH, CHIA-YU LEE and TZU-YING CHAN
National Taiwan University of Science and Technology

149 **Visualizing Urban Systems: Revealing City Infrastructures**

CHRIS KRONER, PHU DUONG, LIZ BARRY and MIKE SZIVOS
Columbia University

Lifecycle Building Card: Toward Paperless and Visual Lifecycle Management Tools

Holger Graf¹, Souheil Soubra², Guillaume Picinbono², Ian Keough³, Alex Tessier⁴, Azam Khan⁴

¹Fraunhofer IGD

Fraunhoferstr. 5
64283 Darmstadt, Germany
holger.graf@igd.fraunhofer.de

²CSTB

École des Ponts ParisTech 6
77455 Marne-la-Vallée, France
{first.last}@cstb.fr

³Buro Happold

9601 Jefferson Blvd, Suite B
Culver City, CA 90232, USA
ian.keough@burohappold.com

⁴Autodesk Research

210 King Street East
Toronto, ON, Canada
{first.last}@autodesk.com

Keywords: Building Information Model, Augmented Reality, Building Lifecycle, Visual Simulation

Abstract

This paper presents a novel vision of paperless and visual lifecycle building management tools based on the coupling between Building Information Models (BIM) and Augmented Reality (AR) called Lifecycle Building Card. As the use of BIM increases within the architecture, engineering, and construction industries, new opportunities emerge to help stakeholders and maintenance operators to leverage the BIM dataset for lifecycle issues using realtime environments and simulation. In particular, a tighter coupling of BIM with computer vision techniques could enable innovative lifecycle management tools based on AR concepts. In this context, this work explores the possibilities and derives theoretical and practical concepts for the use of BIM enhanced by AR for supporting maintenance activities in buildings. An implementation of a wireless spatially-aware display is presented as a first step toward the stated vision.

1. INTRODUCTION

Society is facing an overwhelming number of urgent issues related to global warming, carbon footprint and energy consumption reductions. The building sector is particularly under pressure as it is one of the biggest consumers of energy, either directly for lighting and thermal comfort (heating and air conditioning) or indirectly from the production of building materials. It also largely contributes to the massive use of critical resources (such as energy, water, materials and space) and is responsible for a large portion of greenhouse gas emissions. For example, in Europe, the construction sector generates up to 25% of the greenhouse gases and uses up to 45% of energy overall. Therefore, any serious strategy aiming to reduce greenhouse

gas emissions and deliver energy savings will have to include the construction sector. Thus, the architecture, engineering, and construction (AEC) sector is under significant pressure to reduce its ecological footprint but at the same time, to provide better living and working conditions.

Current business models and working methods are clearly not resulting in sustainable systems and so, new solutions are needed to address these complex and multidisciplinary issues. Improvements needed in the AEC processes relate to:

- Improved architectural and technical design.
- Building components coherent with the design.
- High-performance construction processes.
- Management of the overall lifecycle process i.e. design, construction, operation, maintenance, refurbishment, and destruction with a feedback loop of learned best practice to future projects.
- An operation process that ensures maintenance activities adapted to use and transparent to occupants.
- Systematic inspection of the building envelope.
- Inspection of "weak points" (openings, cabling, electrical plugs) to ensure proper insulation installation.
- Verification of HVAC components (regulation, ventilation, heating) in operation.

A central problem in buildings is the successful implementation of the original design intent. To support continuous improvement after construction, to reach the original design goals, the creation and movement of data, throughout the lifecycle of an AEC project, must be accessible and usable.

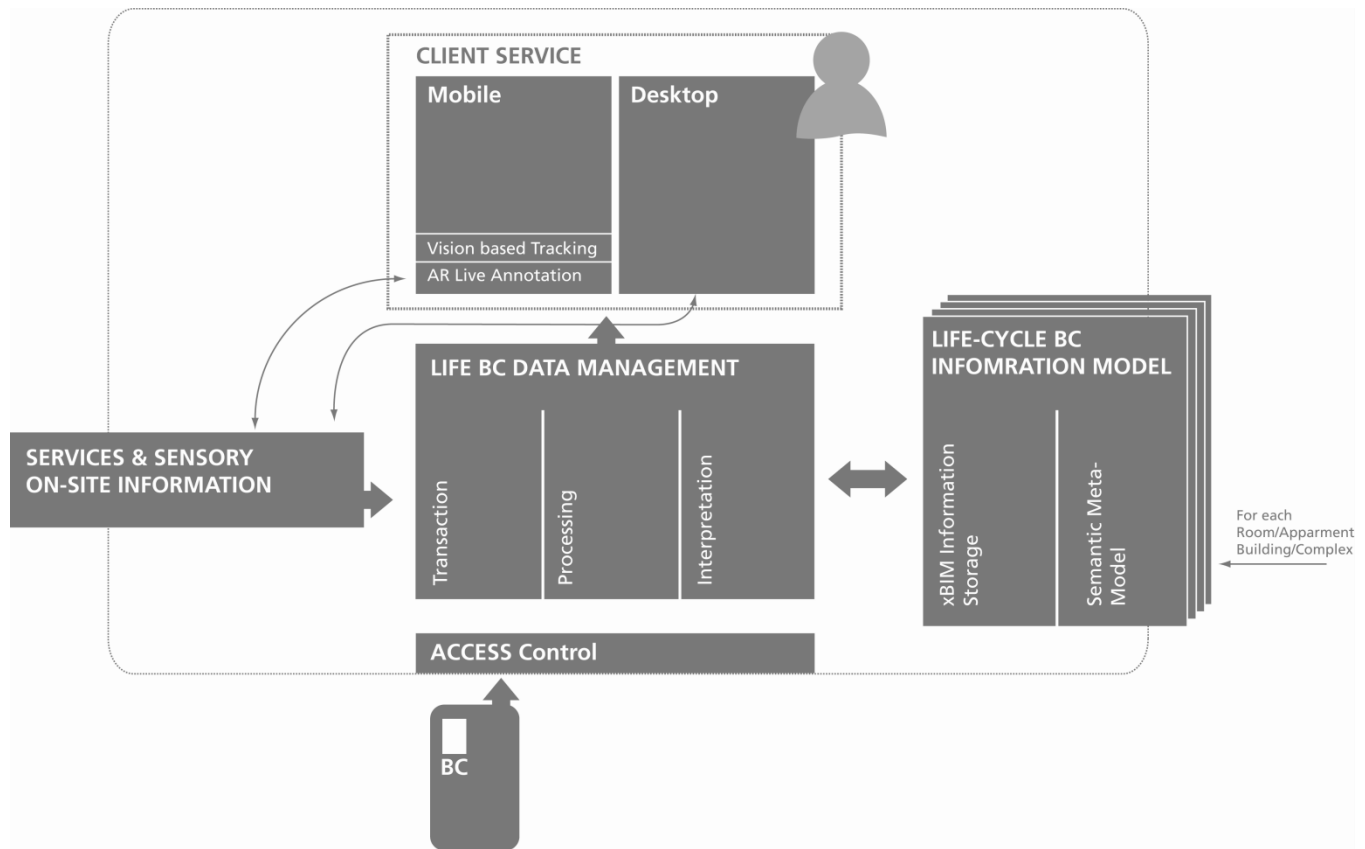


Figure 1. Architectural layout for the visual management of lifecycle building information.

2. LIFECYCLE BUILDING DATA

The majority of the building stock that will be available in 2050 already exists today. The original design intent of these buildings is generally available through the construction drawings but the difference between intent and as-built (the on site condition) is challenging to discover. Surveys (TNO 2010) show that most common complaints, in the area of comfort and high energy consumption, are caused:

- 15% by design errors
- 85% by equipment handover & maintenance problems

This indicates that maintenance and facility management (FM) operations on this existing stock are essential areas of focus for improvements. In particular, the combination of Building Information Modeling (BIM) together with computer vision and tracking technologies would allow future applications to capture and visualize the “as-built” information for “just in time” operation and maintenance tasks. Furthermore, innovative information and

communication technologies (ICT) offer potential for the deployment of ubiquitous, anytime anywhere, applications that can be adapted to these needs.

In order to allow a fusion of BIM entities and Augmented Reality a dedicated workflow backbone has to be envisioned for the purpose of an overall lifecycle management of building information. Ideally, it will be a service oriented information system consisting of components which provide an integrated management of BIM data. The planned architecture (see Figure 1) provides a scalable client application interface for the access, retrieval and distribution of enhanced building information model data, its annotation branches and change history. A data management system suits as a service backbone serving different purposes of information processing. The data management system follows a modular service oriented approach in which different services are provided such as transcoding of information models, processing of image data, feature comparison for online tracking, and interpretation modules for the interlinking of semantic

information. Users or other involved persons would be extending the database throughout the operational life of a building by gathering and maintaining the data and versions.

Therefore data in the BIM would be extended reflecting the users need for additional meta-information attached to each object or element. Simple textural information or multiple media elements have to be processed and interlinked to physical and virtual 3D/3D elements. The information model will then be wrapped through a semantic module which would take care of the management of the metadata. A physical token, the Lifecycle Building Card serves as an eCard and will allow access to building information throughout the overall lifecycle of the building.

3. BUILDING INFORMATION MODEL (BIM)

Virtually all of the surveys carried out in the construction industry place interoperability as a key issue for the use of ICT (Wix 2009). The evidence for available cost benefit comes from a study conducted by the U.S. National Institute for Standards Technology (NIST) in which the lack of interoperability was estimated in 2004 to cost U.S. industry more than US\$15 billion per year (NIST 2004). However more recent estimates suggest that this figure might be too low.

The key to interoperability is that there must be a common understanding of the building processes and of the information that is needed for, and results from, their execution. This is where BIM plays a key role since it is the term applied to the creation and use of coordinated, consistent, computable information about a building project in design, in construction and in building operation and management. The term has been adopted within the building and construction industries to replace Computer Aided Design (CAD). With CAD, the emphasis was on the creation of a representation of the building geometry, such as walls, windows, and doors, in 2D using basic geometric entities such as lines, arcs, and circles. BIM places the emphasis on the objects from which the building is constructed or that describe the process of construction such as walls and windows, but also abstract concepts such as tasks or approvals. The emphasis is placed on 'information' and 'modelling'. BIM is sometimes referred to as virtual design and construction (VDC) (Kunz 2009) because it can be used for simulation of the real building.

Building Information Modelling is the process of using modern software tools to deliver a 'Building Information

Model' which is the collection of objects that describe a constructed item. Both the process and the result use the term BIM. It should be noted that whilst the word 'building' is used, BIM can be applied to any constructed artifact including bridges, roads, process plants and others.

An object represents an instance of 'things' used in construction. These can include:

- physical components (e.g. doors, windows, pipes, valves, beams, light fittings etc.),
- spaces (including rooms, building storeys, buildings, sites and other external spaces),
- processes undertaken during design, construction and operation/maintenance,
- people and organizations involved,
- relationships that exist between objects.

An object is specified by its identity, state and behaviour:

- each object has a unique identifier that makes it distinct from any other object even where they are otherwise exactly the same,
- state is determined by the values given to the data attributes of that object,
- behaviour specifies how the object reacts to operations carried out on them.

Within BIM, the geometric representation of an object is an attribute. This differs from CAD in which geometric representations of objects is critical. A BIM handles objects as though they were real things and not just as a representing shape. Shape is an attribute (or property) of an object in exactly the same way as cost or construction time or the material from which the object is constructed. Quite often, a BIM is discussed as a 3D object model. Use of '3D' makes the attempt to characterize BIM as being geometry driven, similar to usage of the term in CAD. A BIM will often be represented as 3D. However, it could also be represented as an abstract list of objects and relationships, or 2D CAD-like drawings may be extracted from the BIM.

BIM software is typically seen as being the large mainstream applications such as Revit, Bentley, ArchiCAD and similar. Increasingly however, downstream applications such as those used in structural, energy and HVAC applications are becoming BIM applications. The definition of 'what is BIM software' needs to be wider than has been

considered in the past. It is possible even that BIM is not one single software application but is the result of multiple software applications working collaboratively.

IFC (Industry Foundation Classes) emerged as the major standard for BIM implementation in the scope of construction industry information exchange [IFC2x3]. Its development is the result of an industry consensus building process over several years and across many countries. IFC contains common agreements on the content, structure and constraints of information to be used and exchanged by several participants in construction and FM projects using different software applications. The result is a single, integrated information model representing the common exchange requirements among software applications used in construction and FM specific processes. It is currently registered with ISO as a Publicly Accessible Specification [ISO16739] with work now proceeding to make it into a full ISO standard.

4. AUGMENTED REALITY (AR)

A comprehensive overview of Augmented Reality within the AEC sector can be found in Wang (2009) and Graf (2010). Most research in AR focuses on the challenges of high quality rendering, using advanced scene graph technology in combination with fast graphics accelerators (e.g. for occlusion calculations of real/virtual objects) and tracking technology for mobile applications. Nevertheless, within the architectural domain one of the most obvious applications is visualisation of buildings on site. In this context, Dunston et al. (2002) use AR technology for visualising AEC designs. An obvious extension of onsite visualisation is being able to design onsite as well. Hence, another application area has been studied in aiding the design and construction process. Shin and Dunston (2008) showed that AR is a potential technology that can aid several work tasks on a construction site for building and inspection, coordination, interpretation and communication. By presenting construction information in a way that is easier to perceive, AR is expected to provide more cost and labor effective methods to perform the work tasks. The potential benefits deploying user studies for AR in construction have been suggested or demonstrated e.g. excavation information (Roberts et al. 2002) or steel column inspection (Shin and Dunston 2009).

Spatially-aware displays have been studied for some time (Fitzmaurice 1993). Since that time the processors in

mobile phones have become fast enough to also support AR applications (Möhring et al. 2004, Henryssen and Ollila 2004, Stricker et al. 2009). The mobile phone is an ideal AR platform because the current phones have full colour displays, integrated cameras, fast processors, and even dedicated 3D graphics chips. Wagner (2007) identifies several advantages for PDA and mobile phone AR such as low per-unit costs, compact form factor, low weight allowing comfortable single-handed use, and touch screens for intuitive user interfaces. However, in view of the management and presentation of building lifecycle data, currently no platform is available that allows the efficient use of AR within the AEC sector, nor addresses potential application areas in which this data might be useful. As mentioned above, there are some fragmented solutions but clearly lack of integration. In order to make use of this technology and create sustainable impact on the AEC sector, one of the key issues is to establish a robust and markerless tracking system that is adapted to the use of BIM. Furthermore, it should allow capturing and tracking of changes to the building envelop due to maintenance tasks in the context of lifecycle management of building information. Thus there is a need to establish feedback channels from the real environment into extended user-defined BIM structures.

5. TRACKING

The localization of the user on the site represents a key issue of augmented reality. While several mobile devices provide a global positioning service (GPS) receiver and an internal digital compass, only coarse positioning information is available and even then, primarily in an outdoor setting. Furthermore, such sensors are too slow and not accurate enough to allow proper superposition of virtual views onto images of the surroundings in real-time. Vision-based sensors can provide additional data but when looking at a white surface such as a wall, for example, to see what may be inside the wall, further data is needed to differentiate which floor the user may be on and which wall may be in view. Many approaches have been developed, the main issue being reliability and accuracy (no jitter). Difficult lighting conditions of indoor and outdoor environments further complicate matters. Within the envisioned work, we intend to deploy a markerless robust tracking workflow that comprises two major steps: first, establishing and initialisation of a feature map without a priori information that allows for an optimal transformation and alignment

using a small subset of known feature points. Secondly, the fusion of the reconstructed map and the obtained on-site image sequence, in order to train a feature classifier for subsequent tracking. This establishes frame-to-frame 3D tracking and a retrieval of 3D/3D-correspondences between a small subset of reconstructed points and their true anchors within the CAD part of the BIM model.

5.1. Initialisation Step

The initialisation is a major challenge in order to obtain a first estimate of the scene's 3D geometry. Typically Kanade-Lucas-Tomasi features (KLT) (Bleser et al. 2006; Baker et al., 2004) are initialized with a corner detector (Rosten et al. 2006) in the first frame and tracked in 2D throughout the following frames. Several tracks of the features are stored back into a database for later recall. The idea is to establish a structure from motion analysis with a synthesis mechanism that allows retrieval of 2D point correspondences of two subsequent images. This mechanism delivers a matrix that can be decomposed into a rotational matrix and a translation vector. This vector can be used to triangulate 3D positions of the points. However, there are situations in which this procedure fails, i.e. when the matrix is not well defined. This can happen when the translation vector is not available, which means there is no translational move within the camera's motion. In this case, a homography-based reconstruction method must be used (Faugeras et al. 1988).

The following steps have to extend the initial found feature map incrementally while moving the camera. Several features will traverse different states in which their contribution to the overall reconstruction process changes gradually: points which are only passively triangulated at the beginning may serve later actively for pose estimation of the camera (Bleser et al. 2009). When the pose of the camera is valid, new features are initialized with a corner detector at locations in the image, where the feature density is low. Therefore, we rasterise the image into grid cells and apply the corner detector with non-maxima suppression. We then classify a feature as initialized if the corresponding image cell does not contain a maximum number of successfully tracked features. After initialization and the successful tracking of a feature in a second frame, an initial 3D location estimate can be obtained through triangulation (see above). If the feature's 3D-position is considered to be sufficiently accurate it may serve for pose estimation. Otherwise smoothing using Kalman filtering can be applied.

5.2. 3D/3D Correspondences and Frame-to-Frame Tracking

The establishment of the 3D/3D correspondences can be obtained in different ways, either interactively using a dedicated workflow tool, that allows to interactively manipulate a CAD reference model from our BIM or using a text field in order to fill in the coordinates. This has to be done only for few relevant features as the following tracking exploits the relative position to the remaining initialised features of the feature map (see Figure 2).

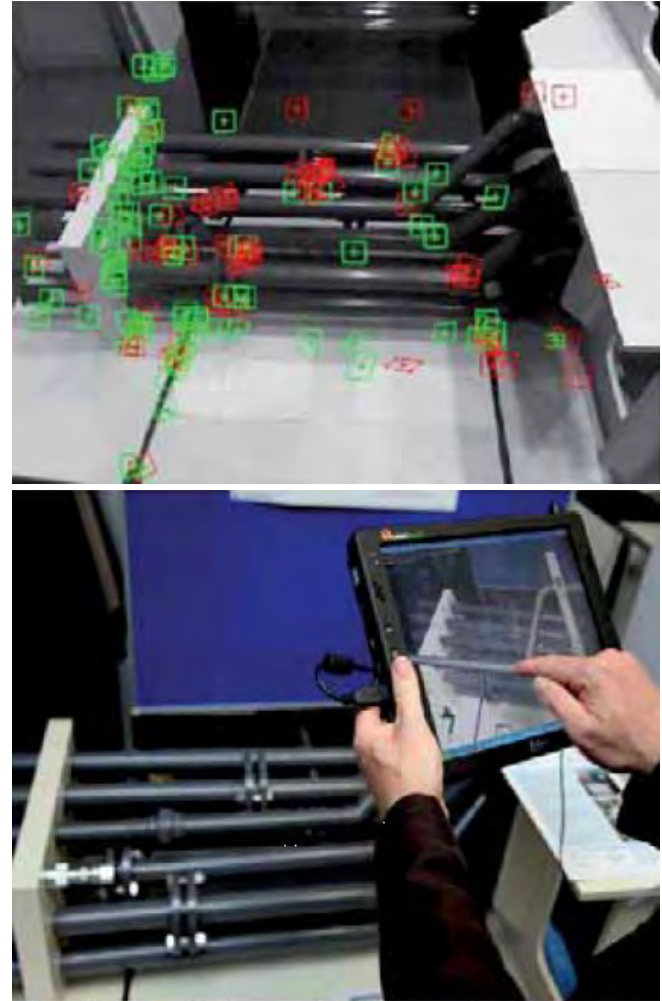


Figure 2. Example of initialised features (top) and its mapping to 3D virtual objects modeled using a CAD model (bottom).

Alternatively, it is also possible to take into account a list of known points given at the beginning (e.g. coordinates of markers or salient points on a known object within the real environment). It is now possible to calculate an optimal transformation based on rotational \mathbf{R}^* , translational \mathbf{t}^* and scale \mathbf{s}^* entities for the reconstructed feature map based on a

least square optimization. A minimum number of three 3D/3D-correspondences is needed (which should not be collinear). Up to 10 points are usually sufficient depending on how far the selected anchor points are spread along the scene.

5.3. Line Model Tracking

A complementary approach which could support the robustness of the above mentioned way is the use of an online line model tracking algorithm as presented in (Wuest et al. 2007). The algorithm creates a view-dependent 3D line model based on a prediction of the camera pose and exploiting an edge map by analyzing discontinuities in the z-buffer and the normal-buffer. Here, two types of edges are used which either correspond to a partially occluded surface or crease edges which are the locations of two adjacent surfaces with different orientation. Finally, the tracking is established using an extraction of 3D-lines out of the edge map based on a Canny like edge extraction algorithm. This results in a 3D contour within the world coordinate system by un-projecting every pixel in the edge map to the information stored in the z-buffer. During the contour following of the edges in the edge map 3D control points and their 3D direction are generated directly and used as the input for the tracking step. The real time edge detector allows 2D edge information being registered with edges obtained from the CAD model within the BIM (see Figure 3).

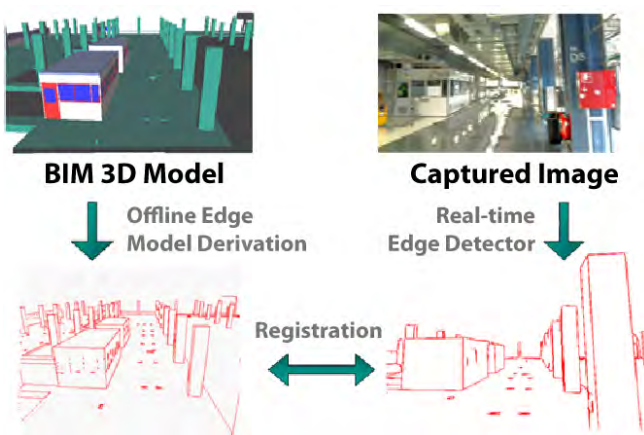


Figure 3. Model based tracking using BIM data.

6. PROTOTYPE APPLICATION

Mobile devices will be used in order to present “as built” information to the owner, landlords or persons in charge of maintenance. Users will be able to browse through

historically grown information spaces in which changes done throughout the lifecycle of the building can be monitored, visualised at a certain level of detail, explode geometries retrieving higher resolution of the information space and to capture changes made to certain built in elements such as pipes, electrical wires, cabling, etc. (see Figure 4). Moreover, the envisioned system will investigate several possibilities to feedback real-time multidimensional data and multimedia annotations in order to enhance the BIM. A semantification module will make them accessible to experts, timestamp those and record the historic evolvement of the quality of the building status. A physical access token will grant access at different levels of information ensuring privacy and security enabling adequate filtering mechanism within the Life BC management system.

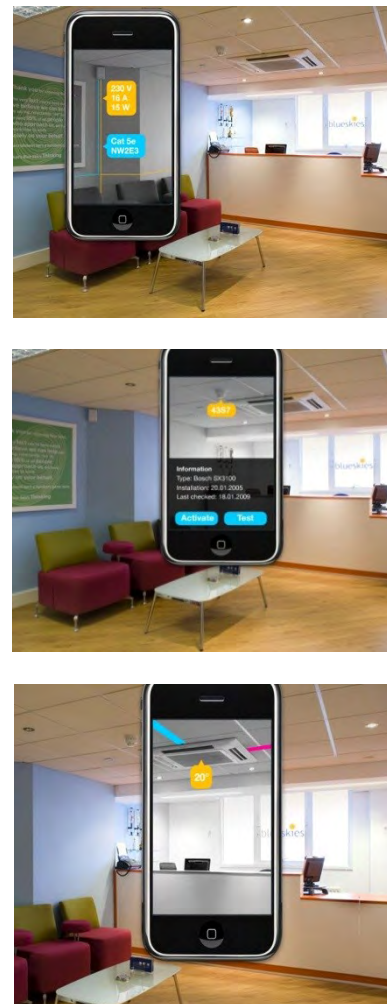


Figure 4. Application scenario of maintenance activities; “visualize the invisible”.

In order to prototype a mobile application, an existing BIM viewer called goBIM (Keough 2009) was enhanced as a proof of concept. Using special markers for in-situ calibration, the goBIM application is able to overlay live camera data and display overlayed, locally cached BIM data. Future prototypes will add tracking and remote data base capabilities.



Figure 5. Registering at calibration tag (top). Inspecting HVAC above ceiling (middle). Image shown on mobile device screen (bottom).

6.1. Using Markers towards a markerless system

Regardless of the quality and accuracy of feature based tracking, the need for calibration and verification of tracker accuracy is a fundamental requirement of any such system. In order to test and prototype a markerless system, the use of markers and fiduciary symbols is a proven technique employed in previous approaches (Zhou et al. 2008).

Fiduciary symbols by themselves provide easily identifiable features for tracking but generally, do not encode complex or structured information. QR tags (Denso Wave) provide unique, machine identifiable markers and also have the capability to store slightly more than 4 kBytes of information. The information stored can take the form of custom structured information or remote tags and identifiers like URLs or GUIDs. BIM models lend themselves well to the generation of QR tags for the labeling of real world objects as they map to their digital representations in a BIM.

Accelerometer and feature based tracking often requires calibration and tracking. Using GUIDs or BIM camera positions embedded into QR tags authored by the BIM application, we can easily identify and locate objects in the real world for verifying and calibrating markerless systems. Towards that end, our prototype system uses QR tags to embed both the unique BIM identifier as well as the camera position. Our prototype uses accelerometers within the mobile device for local navigation of the immediately surrounding area while overlaying BIM data with the current camera feed for the given view as seen in Figure 5.

6.2. BIM Annotations

Creating and maintaining calibration markers in BIM can be problematic. So, we propose the annotation of a BIM by authoring calibration markers as part of the BIM authoring process. To create the markers used in our prototype, a custom written addin generates a QR tag and stores it as a BIM annotation in the form of an image or texture. In this fashion, tags can be authored and centrally maintained using existing software. Open BIM datamodels, such as IFC, provide various mechanisms for annotation so that tags are portable and maintainable. IFC entities such as IfcAnnotation derived classes as well as IfcSurfaceTextures and IfcImageTextures provide the mechanism for storing the QR image. IfcAnnotation classes contain representations for lines, curves, fill areas, symbols, text and even general 3D geometry which can be used to create physical geometry annotations like blocks for the QR markers to be pasted on.

Using surface textures we can attach the tags to the custom surfaces via IfcSurfaceAnnotations to provide physical proxies of the tags in 3D.

7. CONCLUSION

The presented work explored the possibilities and derived theoretical and practical concepts for the use of BIM in combination with real-time visual presentation systems such as Augmented Reality. It further on provides advanced concepts for the BIM and AR technology being integrated based on a scalable, service driven architecture in which tasks for the transcoding of BIM models and their use for registration with real world environments are supported. A first prototypical implementation was shown to help discover potential difficulties. Finally, we hope to efficiently be able to support maintenance activities in the future.

Acknowledgements

The endeavours for a Lifecycle Building Card are supported by the Inter-Carnot Programme of ANR (France) and BMBF (Germany).

References

- BAKER, S., MATTHEWS, I. (2004) LUCAS-KANADE 20 YEARS ON: A UNIFYING FRAMEWORK. *INT. J. OF COMPUTER VISION*, 56(3):221–255.
- BLESER, G., STRICKER, D. (2009) ADVANCED TRACKING THROUGH EFFICIENT IMAGE PROCESSING AND VISUAL-INERTIAL SENSOR FUSION. *COMPUTER & GRAPHICS (C&G)*, 33(1):59–72.
- BLESER, G., WUEST, H., STRICKER, D. (2006) ONLINE CAMERA POSE ESTIMATION IN PARTIALLY KNOWN & DYNAMIC SCENES. *ISMAR*, 56–65.
- DENSO WAVE. WWW.DENSO-WAVE.COM/QRCODE/ABOUTQR-E.HTML
- DUNSTON, P.S., WANG, X., BILLINGHURST, M., HAMPSON, M., (2002). MIXED REALITY BENEFITS FOR DESIGN PERCEPTION, *PROC. THE 19TH INTERNATIONAL SYMPOSIUM ON AUTOMATION AND ROBOTICS IN CONSTRUCTION*, pp.191–196.
- FAUGERAS O. & LUSTMAN, F. (1988) MOTION AND STRUCTURE FROM MOTION IN A PIECEWISE PLANAR ENVIRONMENT. *INT. J. OF PATTERN RECOGNITION AND ARTIFICIAL INTELLIGENCE*, 2:485–508.
- FITZMAURICE, G.W. (1993). SITUATED INFORMATION SPACES AND SPATIALLY AWARE PALMTOP COMPUTERS. *COMMUNICATIONS OF THE ACM*, 36(7), 38–49.
- GRAF, H. & SANTOS, P. (2010). AUGMENTED REALITY FRAMEWORK SUPPORTING CONCEPTUAL URBAN PLANNING AND ENHANCING THE AWARENESS FOR ENVIRONMENTAL IMPACT. *SIMAUD'10*, pp. 142–149.
- HENRYSSON, A., OLLILA, M. (2004) UMAR: UBIQUITOUS MOBILE AUGMENTED REALITY. *PROCEEDINGS OF THE 3RD INTERNATIONAL CONFERENCE ON MOBILE AND UBIQUITOUS MULTIMEDIA*, pp. 41–45.
- KEOUGH, I. (2009). GOBIM: BIM REVIEW FOR THE IPHONE. *ACADIA '09*. pp. 273–277.
- KUNZ, J., FISCHER, M. (2009). VIRTUAL DESIGN AND CONSTRUCTION: THEMES, CASE STUDIES AND IMPLEMENTATION SUGGESTIONS, CIFE WORKING PAPER #097, VERSION 9: JANUARY 2009.
- MÖHRING, M., LESSIG, C., BIMBER, O. (2004). VIDEO SEE-THROUGH AR ON CONSUMER CELL PHONES, *ISMAR'04*. pp. 252–253.
- NIST (2004). WWW.BFRL.NIST.GOV/OAE/PUBLICATIONS/GCRS/04867.PDF. [LAST VISITED JULY 2010]
- ROSTEN, E., DRUMMOND, T. (2006) MACHINE LEARNING FOR HIGH-SPEED CORNER DETECTION. *LNCS PROC. EUROPEAN CONF. ON COMPUTER VISION (ECCV)*, VOLUME 1, pp. 430–443.
- ROBERTS, G., EVANS, A., DODSON, A., DENBY, B., COOPER, S., HOLLANDS, R. (2002) THE USE OF AUGMENTED REALITY, GPS AND INS FOR SUBSURFACE DATA VISUALISATION. *PROC. FIG XXII INTERNATIONAL CONGRESS, TS5.13 INTEGRATION OF TECHNIQUES*.
- SHIN, D., DUNSTON, P.S. (2008) IDENTIFICATION OF APPLICATION AREAS FOR AUGMENTED REALITY IN INDUSTRIAL CONSTRUCTION BASED ON THE SUITABILITY. *AUTOMATION IN CONSTRUCTION* 17(7):882–894.
- SHIN, D., DUNSTON, P.S. (2009) EVALUATION OF AUGMENTED REALITY IN STEEL COLUMN INSPECTION. *AUTOMATION IN CONSTRUCTION* 18(2):118–129.
- STRICKER, D., PAGANI, A., ZÖLLNER, M. (2009). IN-SITU VISUALIZATION FOR CULTURAL HERITAGE SITES USING NOVEL AUGMENTED REALITY TECHNOLOGIES, *INTERNATIONAL MEETING ON GRAPHIC ARCHEOLOGY AND INFORMATICS, CULTURAL HERITAGE AND INNOVATION (ARQUEOLÓGICA 2.0)*
- BORSBOOM, W. (2010). TNO - BU BUILDINGS AND SYSTEMS. SEMINAR AT ENVIRONMENTAL ENERGY TECHNOLOGIES DIVISION, LAWRENCE BERKELEY NATIONAL LABORATORY
- WAGNER, D. (2007). HANDHELD AUGMENTED REALITY, PH.D., INSTITUTE FOR COMPUTER GRAPHICS AND VISION, GRAZ UNIVERSITY OF TECHNOLOGY.
- WANG, X., SCHNABEL, M.A. (2009). MIXED REALITY IN ARCHITECTURE, DESIGN AND CONSTRUCTION. *SPRINGER*. pp. 157–170
- WIX, J. (2009). IMPROVING INFORMATION DELIVERY IN COLLABORATIVE CONSTRUCTION INFORMATION MANAGEMENT. *TAYLOR AND FRANCIS*. pp. 156–165.
- WUEST, H., WIENTAPPER, F., STRICKER, D. (2007) ADAPTABLE MODEL-BASED TRACKING USING ANALYSIS-BY-SYNTHESIS TECHNIQUES. *CAIP'07*. pp. 20–27.
- ZHOU F., DUH H., BILLINGHURST M. (2008). TRENDS IN AUGMENTED REALITY TRACKING, INTERACTION AND DISPLAY: A REVIEW OF TEN YEARS OF ISMAR. *ISMAR 2008*. pp. 193–202.

3D Scans of As-Built Street Scenes for Virtual Environments

N. J. Shih¹, C.Y. Lee², T.Y. Chan³

¹ National Taiwan University of
Science and Technology
43 Section 4, Keelung road
Taipei, Taiwan, 106
shihnj@mail.ntust.edu.tw

² National Taiwan University of
Science and Technology
43 Section 4, Keelung road
Taipei, Taiwan, 106
legoleehk@gmail.com

³ National Taiwan University of
Science and Technology
43 Section 4, Keelung road
Taipei, Taiwan, 106
newniku@gmail.com

Keywords: 3D scans, Urban landscape, Virtual environments

Abstract

The purpose of this research is to build digital urban models by as-built environmental 3D scan data. Scans were made of a street and surrounding buildings for 5.5 km through the center of Taipei, Taiwan. A 3D long-range laser scanner was used to record buildings, landscapes, and open spaces. The scan tolerance was controlled in 4 mm/50 m. The final urban information creates a precise description of objects in a virtual environment with colors and textures feasible for internet browsing, infrastructure dimensioning, and construction monitoring. A 3D representation of scan noise, in terms of pedestrian pattern in front of Taipei Metro, was conducted to illustrate the response of human flow to obstacles.

1. INTRODUCTION

The purpose of 3D modeling is to provide sets of geometric media to communicate among different expertise. 3D urban models not only provide an intuitive media for visual communication, but also facilitate and support the comprehension in urban development, spatial analysis, and decision-making. Due to the large amount of complicated static and dynamic information, the need for urban as-built information is obvious. The information system can be used by government departments, urban and rural area planners, environmental protection agencies, telecommunication companies, and by other expertise in consulting, designing, and engineering where the demand types and application patterns are of concern.

Traditional 3D urban models mainly come from the input and computation of 2D information, which needs additional effort to create details, to map images, and to analyze activities. Aerial photos provide 2.5D information

but lack detailed description of urban as-built forms. Recent scan applications in architectural survey have been enlarged to urban scale and are integrated with computer vision, digital photogrammetry, and computer graphics (Früh and Zakhor 2004; Ikeuchi et al. 2004; Slabaugh et al. 2004; Teller et al. 2003). In Finat's study (Finat 2005) of small towns and midsize urban spaces, the space information and the cases were categorized based on the characteristics, purpose, and content of subject. Computer vision software now facilitates 3D modeling with registration function (Hartley and Zisserman 2000). High resolution digital photogrammetry also provides 3D environment reconstruction function with correct method to confirm results (Triggs et al. 1999). In architecture, 3D scan has been used toward more complicated building configuration (Dick et al. 2004; Fernández-Martin et al. 2005), and the information management and verification systems for different resolution are needed. Close-range laser scan and high resolution data retrieval is also under development (Martinez et al. 2005).

Car-loaded scanners can be used in large urban area with higher efficiency (Früh and Zakhor 2004), however, the cloud data have lower accuracy compared to a fixed scan station and limited scope and range compared to roof-top scanning. Current software development tools interchange and rebuild data between 2D range imaging and 3D data (Hartley and Zisserman 2000), but the integration between general 3D scan information and details still needs future development (SanJosé et al. 2005).

The purpose of this research is to build 3D digital urban landscape (also called urbanscape), based on as-built environmental information. The 3D information will be used to create an information system used by academics and practitioners. Design studios will have more precise urban

data around sites, and professional practices can use the data to evaluate the environmental impacts of new projects. A 3D long-range laser scanner is used to record buildings, plants, and open spaces in static configuration, plus the record of pedestrians, vehicles, objects in dynamic form. As the scan tolerance can be controlled in less than 4 mm, the final urban information management system creates the precise description of objects with colors and textures. The system should also be feasible for urban infrastructure monitoring.

After three years of scanning the streets oriented north-south, the fourth year of scanning effort was made to a street of 5.5 km (Fig. 1) oriented west to east. The whole project aims at the urban space re-discovery, inter-relationship identification, behavior observation, historical subject retrieval, etc. In total, 37 scanworlds and about 66,323,123 points have been retrieved so far in 19 days. Fig. 1 shows the whole street, which intersects the scans made in previous years. Scans were made on ground level and on roofs along both street sides. Final cloud models were registered using reference points inside overlapped areas of adjacent scans. Individual scans and the final model are 3D files which can be translated into other 3D formats to work between different CAD platforms or to browse in different VR systems.

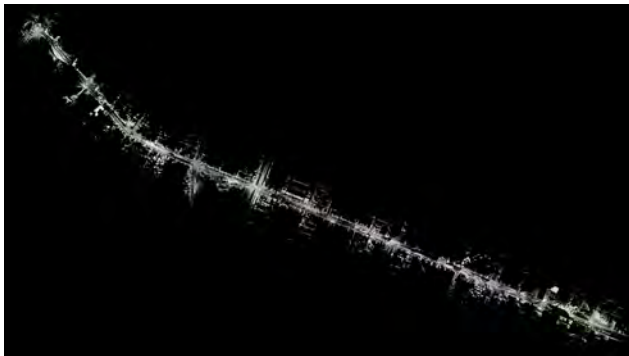


Figure 1. The point clouds of the entire Shin-I road

2. DIGITAL URBAN MODELS

3D urban models are usually created by extruding aerial photos or government land records into volumetric mass. The former has accuracy up to 10 cm, and the latter uses field measures to define property boundary. Most of the sources are 2D data and subject to additional field measurements or the retrieval of original design data to create 3D models. For as-built description of an environment, even more on-site measurements have to be

made to ensure the existing data are accurately updated. While the urban spaces are involved by different government departments, the integration of urbanscape subjects has to prevent the format fragmentation or to assist editing agents. A mechanism which defines the inter-relationship between source providers is in great demand and the final registered 3D cloud model can now provide a relatively complete data description to ensure the consistency between different departments.

The “digital urbanscape” is contributed by static and dynamic scenes of environments. While an environment is scanned, in the mean time the behaviors of occupants are also recorded. Traditional scan usually categorizes the interference of plants, pedestrians, vehicles as “noise” which has to be deleted afterward. In this study, instead, the “noise” becomes part of dynamic scenes to be used as an evidence of human-environment interaction for the evaluation of circulation system and opening allocation. The “noise” parts came from unexpected intrusion of pedestrians or vehicles in scans, however, turned out to be a perfect indication of human behavior of the people’s response to layout. The interaction between people and as-built environment occurred at shop fronts or street corners, in terms of urban furniture, landscapes, or other installations. By excluding surroundings of a focused region we can pay attention to the subject in full scale and in a much more focused manner in addition to an image-based description.

As the scan data are capable of recording exact locations of objects, the scanned routes of human beings, vehicles, cargos are used to illustrate the result of interaction with the gateways and circulation systems inside a region, an open space, or in front of a building. The records include movements, casual stops, intermediate actions, or special occasions (festivals, celebrations, parades). The static background environments are scanned before the activities occur. Comparing scans with/out activities is a way to observe a space and evaluate its post-occupancy.

The pedestrian pattern (Fig. 2) in front of Taipei Metro station presents the response of human flow to obstacles, such as lamp pole, traffic light, other pedestrians, or even our scan team. Since the scan was made in 360 degrees clockwise, the presence of shapes actually covers the activities over a scan sequence for about 15 minutes. The scan data, in point cloud, is then uniformly reduced by a distance of about 20 cm and wrapped by polygons. The

shapes were made by closeness in order to form continuous surfaces. As a result, the activity pattern viewed from the center of the 3D scanner is presented through a collection of free-formed geometries: a rather abstract form for adjacency or density in a defined period of time. One of the shapes is shown at the top-left corner of Fig. 2. The shapes do not necessarily represent the configuration of a single person. The spaces with void geometries or the shapes in smaller size indicate a low frequency of walking through. The lamp pole and traffic light next by are used as a reference of scale. The overlaid cloud indicated the relative location of the scanner and the occurrences of body movements.

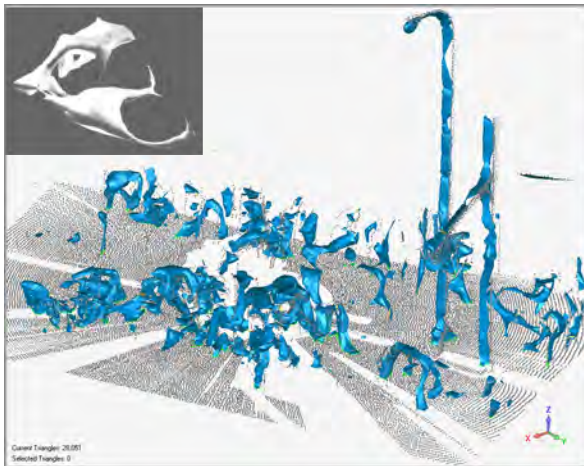


Figure 2. The pedestrian walking pattern in front of a metro station

The scan enables a 3D-based description of context patterns which are categorized by spaces and users. The main category is classified by plants, noise, façade, and street sections. An environment can be changed due to new construction, renovation, demolition, or growth (plants), the urbanscape remains static or close to static during each scan. The scans retrieve as-built data as cloud models (made of point clouds) and polygon models.

3. THE INTEGRATION OF URBAN ELEMENTS

Different urban regions have been planning and remodeling over years. As a construction schedule cannot be recorded in a single scan, a chronological series of scans have to be made to collect sufficient amount of data for analysis. The collection, retrieval, and maintenance of urbanscape data demands long term effort and should follow a city's development. The data should be life-cycle-awared of even before the first scan begins. Since a great number of buildings or regional renovations are undergoing every day,

the geometric data documented before can be referred to afterwards, for example, by overlapping scans to verify the changes. Overlapping periodical scans at the same places is an efficient way to quantify the progress or changes.

The scans of urban circulation system are very important in creating a framework to integrate previous and future works. The scan schedule is planned based on two principles: 1) to expand former scan scope from individual building to regions; and 2) to record the urban development process. New scans are carefully planned to indicate the undergoing construction schedule as part of chronological records for future comparison.

The scan data are also used to combine geographic information with daily scenes. The presentation of urbanscape and the most updated construction process is made by overlapping aerial photos with 3D scans (Fig. 3). The as-built scans present rather updated data for urban planning, as the professionals and students can import the data around design models for evaluation and simulation. For example, the point cloud in Fig. 3 is replaced by new building designs with reference to other scans and urban aerial photos.



Figure 3. The overlapping of point clouds, new building design, and aerial photos

A reference is presented with facades and details available in 3D up to the resolution of 4 mm / 50 meters citywide. The scan data are mapped with camera images. Cases can be also seen by comparing the appearance of photos (Fig. 4) and point clouds (Fig. 5). One way to differentiate the camera images and the picture-like images of scans is the presentation of sky: blue for the camera image and black for the scan data as “black” is the background color on computer display when no data are retrieved.



Figure 4. The street scenes



Figure 5. 3D point clouds



Figure 6. Street scenes near the construction site of a MRT station

Aerial images and urban drawings are not sufficient to describe simulation-related details for most of the site environment. Missing details mainly came from new

constructions or renovations. Urban details are now substantiated and presented by scans of which were made at lower viewing angles to reflect normal visual perception experiences.

Interaction between designers and environments which used to be studied based on design data now can be made in as-built 3D form. The reality of urban scenes can reach higher levels, typically in terms of construction progress (Fig. 6). Before construction starts, existing environments can be scanned to provide detailed as-built information for new design in programming stage. The new building can be imported into a virtual world with scan data to illustrate both visual details and structural details, including urban furniture (sidewalk seats, traffic control facilities) and landscapes (by types, shapes, height). The mass, proportion, and details appear to be very realistic. In contrast to 2D land registration drawings, a user can browse block by block of 3D scenes similar to the translation, scaling, and rotation of computer models in virtual reality. Earlier tests imported point clouds to GIS platform to integrate with satellite images and aerial photos (10/20 cm resolution); however, this manipulation turned out to be slow or even crashed at the data level of about one million points. In order to display the urbanscape more efficiently, a new platform is designed to present the whole scene by city blocks and to visualize results through appropriate internet-friendly format (i.e. vrmf or oct).

Most of the final architectural drawings may or may not include the as-built environment description in 2D drawings or in 3D models. One way to enlarge the scope of the as-built scan data is by public involvement of integrating individual building data, like the interiors and the spaces below ground level, to create a comprehensive definition of private properties and public spaces. The integration lets the owner be aware of the infrastructure interrelationship between the building and public spaces, and have to meet the development of environments. The “gray area” or the “interstitial spaces” between buildings and sites can be recorded in a chronological manner for the management of urban life cycle data. As new buildings are designed, designers can have the data retrieved from the system at the earliest design stage and have data feedback to the system along or after the construction stage.

4. THE FAÇADES

Facades and skylines are the major identifications of a city. Skylines are not only made by roofs, but also the installations on it (Fig. 7). Most of the facades are divided into three parts: the ground level with recessed enclosure for entrances and weather protection, the body with evenly divided grid pattern, and the roof with service towers, antennas, and advertising panels. The street width leads to a code-defined building height. As the average height is about ten stories, building front becomes a perfect location to draw a driver or a passenger's attention even in a construction stage.

Most of the buildings are offices and apartments. The entire façade has gaps, which are made by low rise temple, park, or other streets. The infill open spaces and the buildings create a specific solid-void rhythm which is usually considered to be an identity of this area.



Figure 7. Façades and skylines on both sides of the street

5. SCAN SYSTEM

Instead of using traditional survey data which were retrieved discretely and manually, this study applied a long-range (350 meters) 3D laser scanner, the Cyrax 3000 (Fig. 8), for continuous data retrieval. The system comes with Cyclone 5.5 software for scanner control and data manipulation. A Class B laser is used and the distance is measured by the differentiation of time the laser beam travels between the scanner and the target. The laser scanner projects laser dots in 360x270 degrees. Actual density can be different in width and height depending on the distance specified between two adjacent dots at a certain distance

away from the scanner. The laser dots in a scan or registered scans form a “point cloud” in which each dot indicates the x-, y-, and z-coordinate of the point. A point cloud represents a collection of geometric data, which belongs to an object's surface and therefore can be used to show the appearance of the object. In the database, point clouds are represented in terms of scanworld or scans with exported data contain x-, y-, and z-coordinates and attributes, such as intensity, or color. 3D scanning is considered to be non-intrusive technology. Therefore, areas blocked by other objects can only be scanned from other orientations. Scanned point clouds were wrapped into 3D surface models for visualization and the creation of section profiles.



Figure 8. The 3D scanner

Scans can be made individually or registered into a large project by referring to tie-points or reference points. Each scan has a tolerance of 4mm/50m (2mm/50m in face model). The system comes with a notebook computer to handle the data received on site. Additional data operation was made on a desktop.

6. CONCLUSION

Scan data, in terms of point clouds, were retrieved to represent as-built geometric information. The digital models were used as the references for chronological comparisons to discover possible configuration changes or construction process. While the study scope has extended from a building, street blocks, to whole street, the mutual relationship among individual building is included to provide a more precise description of environments for design and planning control. The scan data are combined with aerial photos and urban drawings. The integration is life-cycle-aware of and is feasible for future construction reference. The 3D representation of scan noise, which is usually eliminated afterwards, was studied to illustrate the response of pedestrian to the environment in front of Taipei Metro.

Future works will be extended to other major streets to have the whole city scanned. Eventually all the city area will be scanned to establish fundamental references for academic studies and professional practices.

Acknowledgements

This research is sponsored by National Science Council of ROC. The project number is NSC 98-2221-E-011-123 - MY3 (the second year).

References

- Dick, A.R., Torr, P.H.S. and Cipolla, R.: 2004, Modelling and Interpretation of Architecture from Several Images, *Intl J. of Computer Vision*, Vol.60, n° 2, 111-134.
- Fernández-Martin, J.J., SanJosé J.I., Martínez J., and Finat J.:2005, Multi-resolution Surveying of complex façades: a Comparative analysis between digital photogrammetry and 3d laser scanning, *CIPA Symposium*, Torino.
- Finat, J.:2005, Ordering criteria and information fusion in 3D laser surveying of small urban spaces, University of Valladolid, 47011 Valladolid, Spain.
- Früh, C. and Zakhor, A.:2004, An automated method for large-scale, ground-based city model acquisition, *Intl J. of Computer Vision*, 60(1), 5-20.
- Hartley, R. and Zisserman, A.:2000, Multiple view geometry in Computer Vision, University of Cambridge, UK.
- Ikeuchi, K., Sakauchi, M. Kawasaki, and H. Sato, I.:2004, Constructing Virtual Cities by using Panoramic Images, *Intl J. of Computer Vision*, Vol.658 n° 3, 237-247.
- Martínez, J., Finat, J., Fuentes, L.M., Gonzalo, M. and Vilorio, A.:2005, A coarse-to-fine curved approach to 3d surveying of ornamental aspects and sculptures in façades, *CIPA Symposium*, Torino.
- SanJosé J.I., Finat J., Fernández-Martin J.J., Martínez J., M.Fuentes L. and Gonzalo, M.: 2005, Urban lasermetry. Problems and results for surveying urban historical centers: Some pilot cases of Spanish Plaza Mayor, *CIPA Symposium*, Torino.
- Slabaugh, G.G., W.B-Culbertson, Malzbender, T., Stevens, M.R. and Schafer, R.W.: 2004, Methods for volumetric reconstructions of Visual Scenes, *Intl.J. of Computer Vision*, Vol.57, n° 3, 179-2004.
- Teller, S., Antone, M., Bodnar, Z., Bosse, M., Coorg, S., Jetwa, M. and Master, N.: 2003, Calibrated, Registered Images of an Extended Urban Area, *Intl. J. of Computer Vision*, Vol.53, n° 1, 93-107.
- Triggs, B., McLauchlan, P.F., Hartley, R.I. and Fitzgibbon, A.W.:1999, Bundle Adjustment: A modern synthesis, in Triggs, B., Zisserman, A. and Szeliski, R. eds: *Vision Algorithms: Theory and Practice* (Corfu, 1999), LNCS 1883, Springer-Verlag, 298-372.

Visualizing Urban Systems: Revealing City Infrastructures

C. Kroner¹, P. Duong², E. Barry³, and M. Szivos⁴

Columbia University
Graduate School of Architecture, Preservation and Planning
1172 Amsterdam Ave,
New York, NY, USA, 10027

¹dck2103@columbia.edu,

²ptd6@columbia.edu,

³eeb2108@columbia.edu,

⁴mszivos@softlabnyc.com

Keywords: Urban Design, Geographic Information Systems (GIS), Digital Modeling, Simulation, Animation, Videography, Video Composition.

Abstract

Modeling urbanism affects urban understanding. This paper introduces a layered pedagogy focusing on urban design representation. It emphasizes the ephemeral and experiential dimensions of the contemporary metropolis. While this research-oriented coursework has targeted a visual study of infrastructural systems found in the New York-New Jersey metropolitan region, the topical nature of the urban discourse remains universal: urban systems and social production they elucidate transpire across the city at large. Beginning with geographic tools, initial mappings become subjective endeavors with specific intentions or empowered by a design agency (political). This information then is filtered through an algorithm oriented process to efficiently reconstitute 3d attributed data into modeling software for further interrogation and speculation (computation and visual). By running simulations driven by this data, urban systems reveal visible and invisible relationships upon urban space. Cinematic techniques are developed through film analysis studies which are then employed to integrate these animated models with live-action video footage. The resulting work of the lessons aims to produce short videos that animate, analyze and link disparate spatial qualities together. The videos make visible the various latent relationships of how infrastructural systems work and what they mean and do for people in cities.

INTRODUCTION

This research structures fundamental conceptual modes of thinking to visualize and understand urban systems and their visible and invisible relationships found in urban space. In urban design study, the subject of „community“ requires a designer to observe, record, retrieve, and ultimately question how to re-present a nuanced perspective of a study site and its related components. Therefore understanding the physical matter of the city is the vehicle for exploring and developing inventive digital techniques for urban design study and speculation.

1. GEOGRAPHIC INFORMATION SYSTEMS (GIS)

To research specific urban environments, ArcGIS Suite software instruction provides a foundation for managing, discussing and extracting urban data. While GIS provides vast amounts of records, “subjective” questioning, which is also required simultaneously, helps guide a means for data distillation. For example, using the practice of mapping to unearth geographic precinct delineations found in census data sets can be transformed from a passive acceptance of data to an engaged application of data deployed under strategic means. In other words, a line of questioning that a data set can contextualize extracts a revelation for an underserved population within a community.

The boundaries of these selected urban fragments require precision, and tools such as attribute filters and geographical sorting algorithms are layered together and classified to begin interpreting the data. Additionally, customized GIS symbology offers an analytical understanding of the city, while custom data fields can also be added and visualized to individual precincts to inform a

specific line of questioning with current empirical observations.

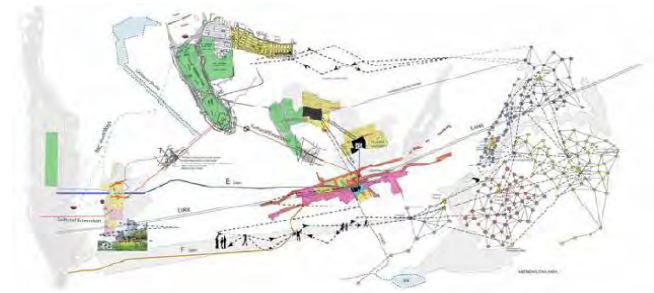


Figure 1. Action Drawing: a visual summary of systems in place exploded to produce an assembly of unknown and often overlooked states and purposes. Columbia University GSAPP, Urban Design 2004.

2. ALGORITHMIC IMPORTATION (MEL)

A custom script (written in Maya Embedded Language, or MEL) has been developed and tested for the importation of urban information into a three-dimensional modeling environment. Improving the typical import-export workflow, this script offers the ability to import a custom dataset table precisely, and efficiently traces the spatial data in three-dimensions using real-world coordinates (using a reference point given by the database) with low polygon counts. By optimizing the processing time it affords quicker feedback loops within the design process. Additionally, the information from the custom precinct database becomes translated as query-able attributes, attached to the discretely modeled urban design elements (fabric, blocks, etc.), for further postulation.



Figure 2. Urban Massing Model after scripting process
Columbia University GSAPP, Urban Design 2009.

3. DIGITAL MODELING (MAYA)

From the script, an initial urban massing model (i.e. buildings) is procedurally reconstituted into an interactive three-dimensional environment, ready for additional manipulation, questioning and detailing. Since the coordinate values of the urban model derive from precise units, additional layers of three-dimensional vector information (i.e. hydrology, roadways) can also be imported to scale for additional description of the urban precinct or neighborhood. Desired detail inquiries, such as surface relief, vegetation textures, materiality, lighting, street infrastructure, among others embed further experiential understanding of the city.

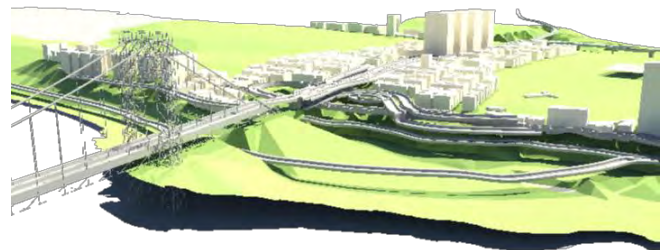


Figure 3. Layered Model with Lighting, Terrain, and Infrastructure
Columbia University GSAPP, Urban Design 2010.

4. LIVE FOOTAGE

It is necessary to identify a system's logic, its working pieces, its changing uses, and the livelihood or conflict it presents within the context of the post-industrial city. To that goal, all studied precincts are also investigated with live audio and video equipment. Characteristics of the physical environment, not qualitatively described by statistical attributes can therefore be observed and recorded, to add another layer of experiential information to this analytical investigation.

5. ANIMATED ANALYSIS (MAYA)

While conventional modeling procedures can help to detail a standard "base" model, the software also posits the opportunity to simulate the intangible – experiential and environmental effects that affect the city and its communities. Perspectival walk-throughs, fluid dynamics (i.e. wind analysis, rain, weather agents), night/daylight

studies, tidal flows, congestion and collision models, all can be tested in this three-dimensional environment as time-based simulations.

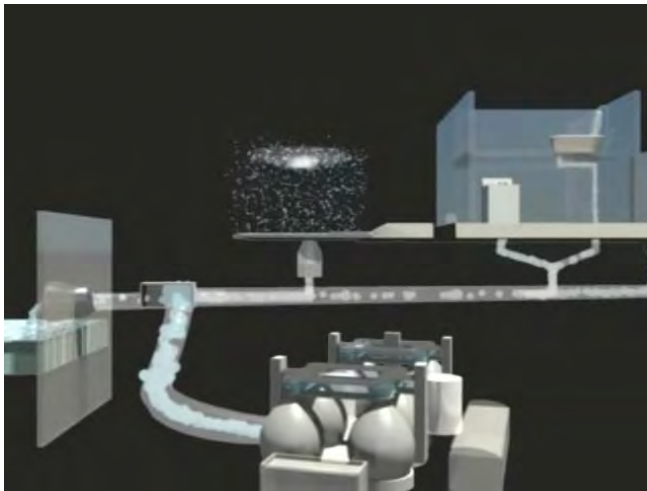


Figure 4: Combined sewer overflow (CSO) infrastructural model and simulation. Columbia University GSAPP, Urban Design 2010

6. CAMERA SYNCHING (MAYA)

The recorded site inquiry offers distinctive perspectives into the community, while the model offers infinite perspective of the myriad issues facing a precinct. Using a camera matching toolset, the two perspectives can be resolved into a combined viewscape allowing the digital and the physical world to coexist- overlaid upon and interacting with each other to address the urban fabric as a temporal phenomenon.

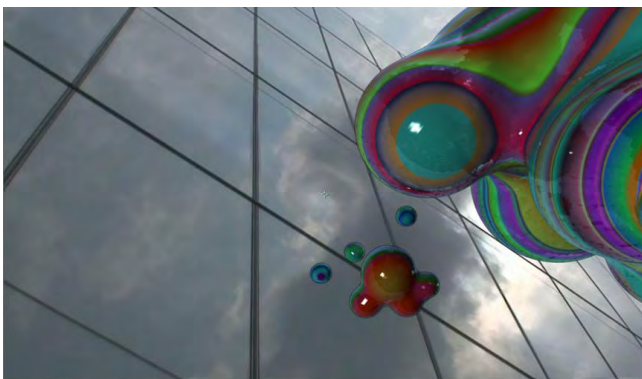


Figure 3. Camera matched footage with computer animated simulation
Columbia University GSAPP, Urban Design 2010

7. VIDEO COMPOSITING (AFTER EFFECTS)

A cinematic approach to the site allows examination of each site not only as a spatial or formal condition, but as a network of forces, pressures, intensities, etc. Beyond mere documentation are added layers of animated and interactive notation. Video editing tools choreograph the physical, temporal, and experiential dimensions with the use of live footage to play with speculative questioning in the context of an urban condition found in the city.

8. CONCLUSION

The result of such study does not end with the discovery of dynamic systems on a particular study area, but rather, each video will begin to postulate the effects of changes and adjustments to an urban system. The collective work of this series of short videos aspires to bring nuanced knowledge to understand the nature of neighborhoods and the impact of visible urban systems embedded in various locales of the contemporary metropolis. Through narrative based film sequences, specific effects can be uncovered within urban systems that expose uncommon understandings of the places that people inhabit. The broader purpose of this use of software and techniques better equips urban designers with a dynamic, yet perhaps a more convincing medium to convey potentially politically charged messages. Short videos reveal urban issues for software experimentation and contemplation while extending urban design pedagogy. Beyond the static drawing, this tooling between applications with the moving image can empower designers, their clients and the public at large with a voice that provides greater access to challenging urban thinking that may be more effective than print and press.

References

- ASCHER, KATE. *THE WORKS: THE ANATOMY OF A CITY*. PENGUIN PRESS, 2005.
- LEFEBVRE, HENRI, *THE PRODUCTION OF SPACE*. BLACKWELL PUBLISHERS INC., 1992.
- MCGRATH, B., AND GARDNER, J. 2007. *CINEMATICS: ARCHITECTURAL DRAWING TODAY*, WILEY-ACADEMY, 2007.

Session 7: Parametric Urbanism

155 **City of Love and Hate**

ADNAN IHSAN, AMIRALI MERATI, EVA POULOPOULOU and FOTEINOS SOULOS
Columbia University

163 **Components for Parametric Urban Design in Grasshopper. From Street Network to Building Geometry**

CHRISTIAN SCHNEIDER and ANASTASIA KOLTSOVA
ETH Zurich

171 **Multi-Objective Optimization in Urban Design**

MICHELE BRUNO, KERRI HENDERSON and HONG MIN KIM
Columbia University

The City of Love and Hate

Adnan Ihsan, Amirali Merati, Evangelia Pouloupoulou, and Foteinos Soulos

MScAAD Candidate
Columbia University
New York, NY, USA,
10027
ai2247@columbia.edu

MScAAD Candidate
Columbia University
New York, NY, USA,
10027
am3555@columbia.edu

MScAAD Candidate
Columbia University
New York, NY, USA,
10027
ep2557@columbia.edu

MScAAD Candidate
Columbia University
New York, NY, USA,
10027
fts2105@columbia.edu

Abstract

The rising complexity of 21st century cities demands a more rigorous and intense understanding of their inherently complex programs, which cannot be resolved by a conventional design methodology. This paper proposes a new design process using an automated workflow that incorporates the computational iterations and the design values of an architect in a unified process aiming to produce high performing optimized results. The procedure, that is described, uses CATIA for parametric simulation and modeFrontier for multi-criteria optimization. The “value meter”, a qualitative assessment, is used to grade the results according to subjective design criteria. The setup operates in two stages, Phase 1 [city scale] elaborates on broad urban land-use goals, whereas Phase 2 [neighborhood scale] explores detailed objectives such as density and infrastructure. The complete workflow operates under the realms of conventional urban design ideas, but produces exponentially large variety of design alternatives.

1. INTRODUCTION

The use of computation and simulation in urban design has a long line of precedents. Conducted research varies from Michael Batty's studies at UCL on agent-based models (ABM) and cellular automata (CA) to the gaming approaches of MVRDV such as the “Space Fighter”, “Function Mixer”, “Access Optimizer” (Batty 1976; Batty 2005; Maas 2007; Dekkers et al. 2002). These approaches treat issues of entropy, evolutionary urbanism, complexity, game theory or the use of multiple scenarios for the introduction of decision making processes in urban planning. This long list of new concepts and processes used in urbanism has a complementary analytical aspect related to Geographical Information System (GIS) and software such as Urban-Sim and Arch view that merge cartography, statistical analysis, and database technology.

The evolution of urban planning tools is of major importance since it denotes a general shift in the way we regard the urban environment. In many ways it marks a decisive turn of urban theory towards concepts of self-organization and emergence for the production of successful urban designs. Regarding the city as a complex system, conventional planning has become an imposed top-down approach of form to urban space. In reality, cities, or at least what we consider as successful cities, are never the results of simply top-down processes. On the contrary, they include a great part of bottom-up generation processes (Alexander 1964). Additionally, due to globalization, cities are now evolving much faster than in the past. This practically means that urban design should somehow incorporate the idea of adaptability to an early planning stage, so that it can take into account mutations that may happen to the urban environment even before actually applying a scheduled urban design. In other words, we need to create a dynamic design method for an increasingly dynamic urban environment.

In this paper, we use a method that synthesizes traditional urban planning concepts with a composite workflow that includes both parametric simulation and automated multi-objective optimization (Benjamin 2010).

Looking back into the precedents one can see examples of urban planning that already use multi-objective optimization. However, in most cases the optimization is part of an intuitive process, therefore it is not automated. In other cases, extensive simulation is used but the interface doesn't allow for a clear overview of the changing inputs. Or even if the simulation is legible, the design tools cannot generate thousands of designs. Learning from the shortcomings of previous experience we introduce a new design tool with the following features:

- Multi-objective optimization
- Automated process
- Legible interface of inputs-outputs
- Thousands of design permutations
- Post-processing tools for the evaluation of the results

For the first phase of our experiment we took a simple urban grid as our case study and chose proximity and adjacency as our primary criteria based on the “state change” technique (Preparata and Shamos 1988). State change is a widely used scripting technique through which an integer is assigned to each specific type of design feature (e.g. the different types of program) in order to give the possibility to the user or the optimization software to automatically replace the feature by changing the integer. This technique is very useful for the optimization process since it allows for easy control of the design permutations. The reason for choosing state-change and proximity rules as our primary simulation technique is three -folded:

- 1) This technique has been widely used and is still being used by a variety of advanced urban analysis and simulation institutes (e.g. CASA: Centre for Advanced Spatial Analysis, University College London- Cities as Complex Systems) and it has proven to perform well both for high resolution and low resolution simulations regardless of its simplicity.
- 2) It is applicable to a spectrum of scales from regional to local.
- 3) It has proven to be a good match once it is integrated with Genetic Algorithm (GA) which is the primary optimization engine.

Further, the same logic was re-applied to the urban grid but this time instead of using a neutral empty grid we incorporated some natural (lake) and manmade (historical building) elements to our grid as a means of simulating preexisting urban conditions. In the second part of our experiment we expanded the model by adding circulation-infrastructure (roads) and density (height regulations).

Our workflow includes the parametric software CATIA (Dassault Systemes) used for modeling and simulation of urban conditions such as attraction or repulsion of program

through the scripted process of proximity and scoring. The complete design setup is fed in modeFRONTIER (Esteco) for automation and multi-criteria optimization.

It is important to point out that the urban simulation is incorporated to the same workflow through the proximity method which is evaluated as positive or negative according to a value system that can take different forms depending on designers' preference, urban legislation or community boards and commissions. We name this set of fitness criteria “value meter” since it is meant to summarize a set of design preferences and, thus, becomes the medium through which the optimization can be achieved. The “value meter” is combined with another set of quantitative goals (percentages) that refer to various land-use programs, leading to a multi-objective problem.

2. EXPERIMENT 1

The first experiment operated on more generic premises of urban context. An 8x8 point grid was generated as the basic urban structure. Four land uses are assigned as inputs, Residential, Commercial, Industrial and Green Space. Each point could host any of these four programmatic land uses. Even such a small setup can create 64 to the 4th power =16,777,216 permutations of possible relationships among urban programs. And part of the intention for automating design is to discover a wide-design space and novel proposals that are beyond known rules of thumb. The aim of the experiment was to explore the relationships/adjacencies formed between these programs and evaluate them based on their arrangement. Basically, our ambition was to achieve heterogeneous arrangements of program in addition to specific qualities of proximity (e.g. housing attracted to green space but repelled by industry). These relationships are evaluated using a scoring system or as we call it a value meter, whereby a relationship is rated, either positive or negative, and stored as a score. The criteria for the score of individual relationship is not scientific but is rather based on architectural / urban case-studies. The formulation of these criteria is later discussed in the research paper.

The automation of this parametric model was done using a multi-objective optimization software called modeFRONTIER, using the “state change” technique that assigns an integer from 0 to 3 to each one of the land uses. ModeFRONTIER starts the workflow by choosing an initial urban- arrangement population from a random search algorithm. One major advantage of using a random initial

population is to explore a wide possible design space before narrowing it down to a specific goal. After the initial population was created, a genetic algorithm of type MOGA-II was used to produce newer generations of higher-performing results based on the objectives set in the modeFRONTIER. In this case the high performing results are produced by maximizing the total score of relationships. The program scores defined in the value meter are used as a type of geometric urban design simulation to drive the multi-objective optimization process.

The automated test ran with an initial population of 50 urban land-use arrangements generated using random algorithm, and another 20 generations was created using MOGA-II search producing 1000 design outcomes. Each experiment took approximately two minutes. In the end the optimization software's charts were used to identify high-performing designs (Figure 1). It is important to note that the optimization software does not give a definite solution but varying design proposals that are optimized. It is further the designers job to look through these design outcomes for the most interesting and unexpected results that overcome the standard rules of thumb.

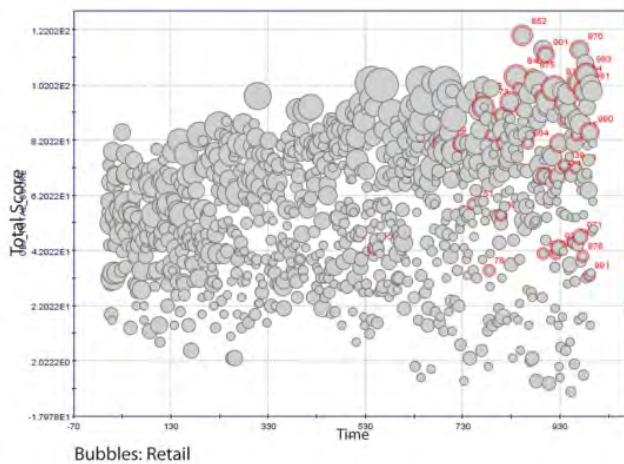


Figure 1. Graphs of the results of the first experiment for retail space.

As far as the content of the experiment is concerned, we expected to observe some kind of clustering of land uses depending on our value meter's setup. For example, clusters of housing surrounded by green spaces or clusters of industry surrounded by commercial. We also expected to see results of totally even grids (e.g. only residential), so we introduced percentages to the experiment in order to avoid this possibility (percentages are further discussed

separately). The results of the experiment confirmed our expectations, but also produced high-performing results that did not correspond to the above clustering logic (Figure 2). Unpredictability was increased in the experiment after the introduction of the pre-existing elements and the extension of the value meter in order to include local relationships between the new land uses, the manmade and natural topography.

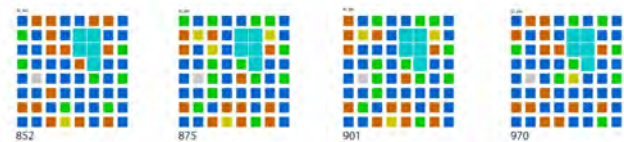


Figure 2. Selected designs from the automated test of the first experiment.

3. EXPERIMENT 2

After completing experiment 1, we used some of the high-performing designs to initiate our second experiment. The second experiment addresses more detailed issues of urban planning and design. In this experiment we explore how different arrangements of road types affect the density on the chosen outcomes of previous experiments. There are three types of road inputs; green for highway roads, red for double-lane road and blue for a single lane road. The tracing of the road network was pre-defined manually in our experiment but the type of road was allowed to change through “state change” operation in Engineering Knowledge Language (EKL) scripting of CATIA. This decision allows the designer to define the neighborhood scales. After defining the tracing, the rest of the process is once more automated.

Among the land use programs, green-spaces and industrial zones are not affected or automated in the process of this experiment but the residential and commercial zones are affected by their proximity to a particular road type. The probable outcomes of road configurations are translated into vertical lengths representing density and height limitations in our experiments. The green line (highways) induces a height of 15m, the red line (double-lane roads) generates a height of 10m and lastly the blue-line (single-lane) creates a height of 5m on its neighboring zones. These discrete height values are not an outcome of an equation but are defined by us as qualitative parameters through another value meter. Once again the adjacencies of land uses and roads generate a value based on their relationship. These values of relationships are defined by our research on the best known

practices. The parametric CATIA model is fed to modeFRONTIER to optimize for highest score and narrow its count to a desired percentage of the total. An initial population of 50 random designs was used through a generation of 10 to produce optimized design proposals (Figure 3). The intent behind this automated process is to create a tool that can be used not only for exploitation, but rather for exploration of new novel designs. Although we were expecting the road types to scatter randomly we realized that the high-performing results (Figure 4) that the workflow came up with, ended up presenting aspects of hierarchy and symmetry (e.g. same type of roads are parallel and next to each other).

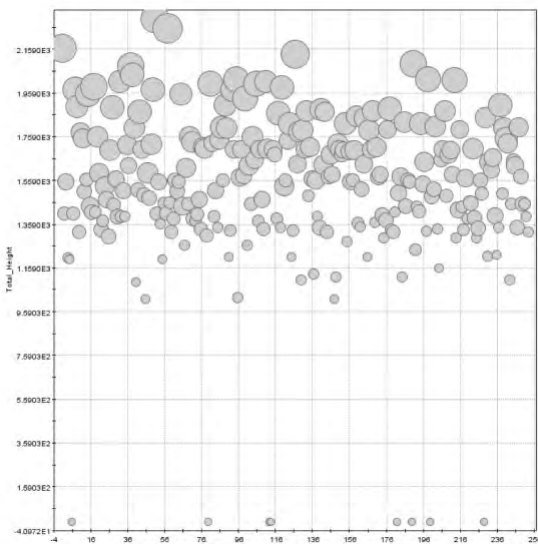


Figure 3. Graph of all the results of the second experiment.

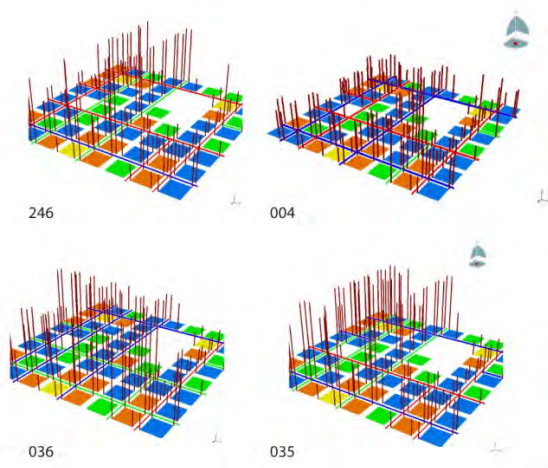


Figure 4. Selected designs from the automated test of the second experiment.

4. SIMULATION TOOLS

The way we address urban simulation is through incorporating it to the design and optimization workflow, without using any additional software. A scripted method of calculating and scoring proximity relations as positive or negative is summarized by the value meter that directly affects the final design, and manages to turn architectural values and preferences into digits. This technique introduces the use of computation for solving qualitative aspects of social, cultural and urban problems in addition to quantitative technical issues. This can be part on a general debate of what defines the design of a successful city in the 21st century.

4.1. Value Meter

The value-meter refers to the idea of preference and how this can be translated into computational knowledge. There are numerous examples of similar table-filling evaluation methods used in several sciences and urban planning over the past decades (Gawande 2006). Especially in the case of urban design, where no such thing as a global setup can be applied, it is important to find a way of letting the computer know what would be preferable each time, so that the proposals derived from the workflow can make sense on different contexts. Since urban design is primarily about relations of proximity of different land uses, our main goal became to create a tool through which the urban planner/architect could easily set his intentions in a simple workflow that can incrementally become more and more complex. In order to do so, we used a scoring system through which the user can transfer his preferences into the program. Through this system that we call value meter, the program (modeFRONTIER) will be capable to evaluate the result according to the designer's personal set of values.

The value meter was created by setting positive and negative scores for programmatic adjacencies, following a debate on what makes a successful city. The results of the debate are inserted to a table of values including the relationships between all the possible combinations (Figure 5). So, in the first experiment we made a list that contained all of the possible neighboring elements, such as residence next to residence, industry next to commercial etc. Then, the second thing that we defined was the range of possible scores. Generally, the range depends on the quantity of combinations that exist, and that is why in our case of a balanced system of four different elements, the range is a set of integers from '-2' to '+2'. The

simulation setup is designed with the intelligence to avoid negative relationships and encourage positive ones in order to produce high-performing heterogeneous arrangements. Similarly, in the second experiment a table with combinations between land uses and road sizes is also designed (Figure 6) in order to define what kind of road will be preferably placed next to a particular land use.











	Pre-existed elements					
						
	+1	+1	-2	+2	+1	+1
	+1	+1	+1	0	0	+1
	-2	+1	+1	0	-2	-2
	+2	0	0	+1	+2	+1

Figure 5. The value meter table of experiment 1.








			
	+2	+1	-2
	+2	+1	0
	-2	0	+2
	+2	0	-2

Figure 6. The value meter table of experiment 2.

Through the proximity setup that the experiment uses, the software (CATIA) can calculate all of the neighboring conditions and evaluate them according to our scoring system through a custom script. ModeFRONTIER is also set up to keep all these evaluations to high numbers. This way, the software constantly weighs all of the contradicting goals and comes up with several designs, while the generative algorithm (GA) tries to filter all the desired characteristics in order to satisfy the objectives and produce even better results. We realized during the process of the experiments that the generative algorithm (GA) needed a certain amount of time in order for to start learning how to produce a more successful design. Most likely the large numbers of 64 inputs with 16,777,216 possible design

permutations were making it hard for the software to narrow down the optimized results early in the experiment. However, already after running the experiment for a random initial population of 50 for 20 generations with a MOGA-II algorithm, which corresponds to approximately 1.30 minutes per iteration, modeFRONTIER came up with results that can be used in order to derive conclusions (Esteco Technical Report 2003-06). Obviously, running the experiment for longer would produce even better designs.

4.2. Percentages

In order to prevent the optimization software from converging towards one type of high-scoring relationship and additionally address the question of quantifying the land uses in an urban design, we created a group of percentage parameters in CATIA that represent the four land uses of the experiment. The reason behind this decision is that the ratios of land uses and program are critical to urban planning and most of the times define whether a proposal meets the necessary requirements. Since one of the major tendencies we observed in the designs that the workflow produced was the scattering of land uses across the master plan, it was hard to quickly quantify the land uses, which made the percentage output useful.

Taking this to another level, these percentages are also set as objectives. For instance, we have set the program to optimize for 60% residential, 15% retail, 5% green spaces and 10% industry. To do that we use the formula:

$$\text{Percentage residential} = \text{abs}(60\% * \text{total number}-x)$$

In this formula ‘x’ is the number of the produced residential blocks, and modeFRONTIER is being set to minimize this absolute difference so as to achieve this percentage. Another experiment would be to define only some of the land uses through percentages and leave others be processed by the optimization software in order to explore more possibilities and potentially come up with unexpected high-performing designs. Percentages can therefore obtain a double role, as they can contribute in the evaluation of the results or they can be set as goals, depending on the designer's needs.

We believe that the value meter combined with the percentages are powerful tools that are applicable to processes beyond the urban simulation approach. They are easily adjustable to several types of experiments, as

demonstrated in our case where they were used for different things in two subsequent experiments. Although in each case the elements and the values were different, the process still worked successfully.

5. CONCLUSION AND FURTHER DEVELOPMENT

The proposed urban experiment can be incorporated into an urban design workflow including multi-objective optimization. Obviously, the grid is a generic model that can be customized in order to simulate particular situations. Some selected points of the grid can represent topographical or other pre-existing elements that need to stay intact from urban intervention. The extent of the grid can be adapted to the design problem as well. Although this particular phase of the project is made possible through the use of CATIA and its link to modeFRONTIER, it is easy to imagine an intermediate step to the experiment, including a link to Analytical Software such as GIS that will add to a more realistic simulation of spatial data and improve the final outcome.

In addition, the system can potentially work in different urban scales with increasing precision. The first version of the experiment presented here is a simulation of general land uses, while the second is a study of densities. One can easily imagine a series of consequent experiments developing in detail the content of the building volume of the city through 3D optimization at the scale of building block. Applying the setup in specific urban contexts and taking into account the existing urban legislation can provide additional restrictions and opportunity for the setup of the urban experiment. For example, in the case of a fictional scenario of expanding New York's city plan, the general grid would optimize for R-Residence, C-Commercial, M-Manufacturing, IH-Inclusionary Housing and L-special purpose districts. The extended experiment (building block and density version) would take into account a set of existing regulations for the specific land uses, such as setback regulations, allowed types of program per land use, allowed building heights, mandatory void spaces and other related quantitative restrictions.

The ambition behind the automation of already established urban design strategies, apart from the rapid production of massive design permutations, is the production of possible unexpected high-performing results. This can lead to the development of different strategies, as it reduces the designers' predisposition towards received ideas

about neighboring programs, densities, circulation networks etc at the early design stage. Of course, this doesn't mean that the workflow completely neutralizes their contribution to the produced urban form. After all, as already explained, the system is highly dependent on the architect's values (value meter and scoring systems) and will produce different results for different value setups. An initial setup valuing the relationship between industry and housing as positive will favor the neighboring of these two uses, although this is largely considered as unacceptable in terms of quality of life, but could be considered as desired in another type of industrial performance-based scenario.

Though most of the process is automated, yet the final selection is based on data interpretation and the designer's judgment. These non-automated filtering factors are mainly based on concepts such as viability, practicality, economy and even aesthetics. The automated process gives numerous design outcomes sorted by the evaluation criteria as generally high-performing in relation to the objectives, but cannot prioritize these objectives. This is where the designer's critical ability and perspective must be employed again.

As a conclusion, this experimental workflow intends to create a balanced system of urban design in terms of computational objectivity and architectural predispositions or urban design policies. This exploration tool does not claim to present instant ideal solutions to urban design problems. On the contrary it aims to automate the process of urban design in a way that will allow for a more thorough exploration of the potential that lies in the highly normative urban space. Realizing the flexibility behind the topographical and regulative constraints would be succeeding this experiment. Interpreting and overlaying the results of the computation process to the usual manual methods would probably enrich and facilitate the process of urban design.

References

- BATTY, M., 2005, *CITIES AND COMPLEXITY: UNDERSTANDING CITIES WITH CELLULAR AUTOMATA, AGENT-BASED MODELS AND FRACTALS*, CAMBRIDGE, MASSACHUSETTS, MIT PRESS, 1-17, 515-521.
- BATTY, M., 1976, *URBAN MODELLING: ALGORITHMS, CALIBRATIONS, PREDICTIONS*, CAMBRIDGE, CAMBRIDGE UNIVERSITY PRESS, XIX-XXV, 1-20, 353-358.
- ALEXANDER, C., 1964, *NOTES ON THE SYNTHESIS OF FORM*, HARVARD UNIVERSITY PRESS.

BRENT, B. ET AL., 2007, SPACEFIGHTER: THE EVOLUTIONARY CITY (GAME :) : MVRDV/DSD IN COLLABORATION WITH THE BERLAGE INSTITUTE, MIT AND C'THROUGH, BARCELONA, ACTAR.

DEKKERS D. ET AL. 2002, THE REGIONMAKER, RHEINRUHCITY CITY, OSTFILDERN-RUIT, HATJE CANTZ.

ESTECO, 2003-06, TECHNICAL REPORT, ON MOGA-II ALGORITHM, AN ENHANCED METHOD OF EXPLORING THE PARETO FRONTIER.

BENJAMIN D., FALL 2010, PROOF 6, ADVANCED ARCHITECTURAL DESIGN STUDIO, NEW YORK, COLUMBIA UNIVERSITY, GRADUATE SCHOOL OF ARCHITECTURE, PLANNING AND PRESERVATION.

GAWANDE A., 2006, THE SCORE: HOW CHILDBIRTH WENT INDUSTRIAL, THE NEW YORKER.

MODEFRONTIER IS COPYRIGHT ESTECO.

CATIA IS COPYRIGHT DASSAULT SYSTEMES.

PREPARATA F. P. AND SHAMOS, M. I., 1988, COMPUTATIONAL GEOMETRY – AN INTRODUCTION, SPRINGER-VERLAG, 185-265.

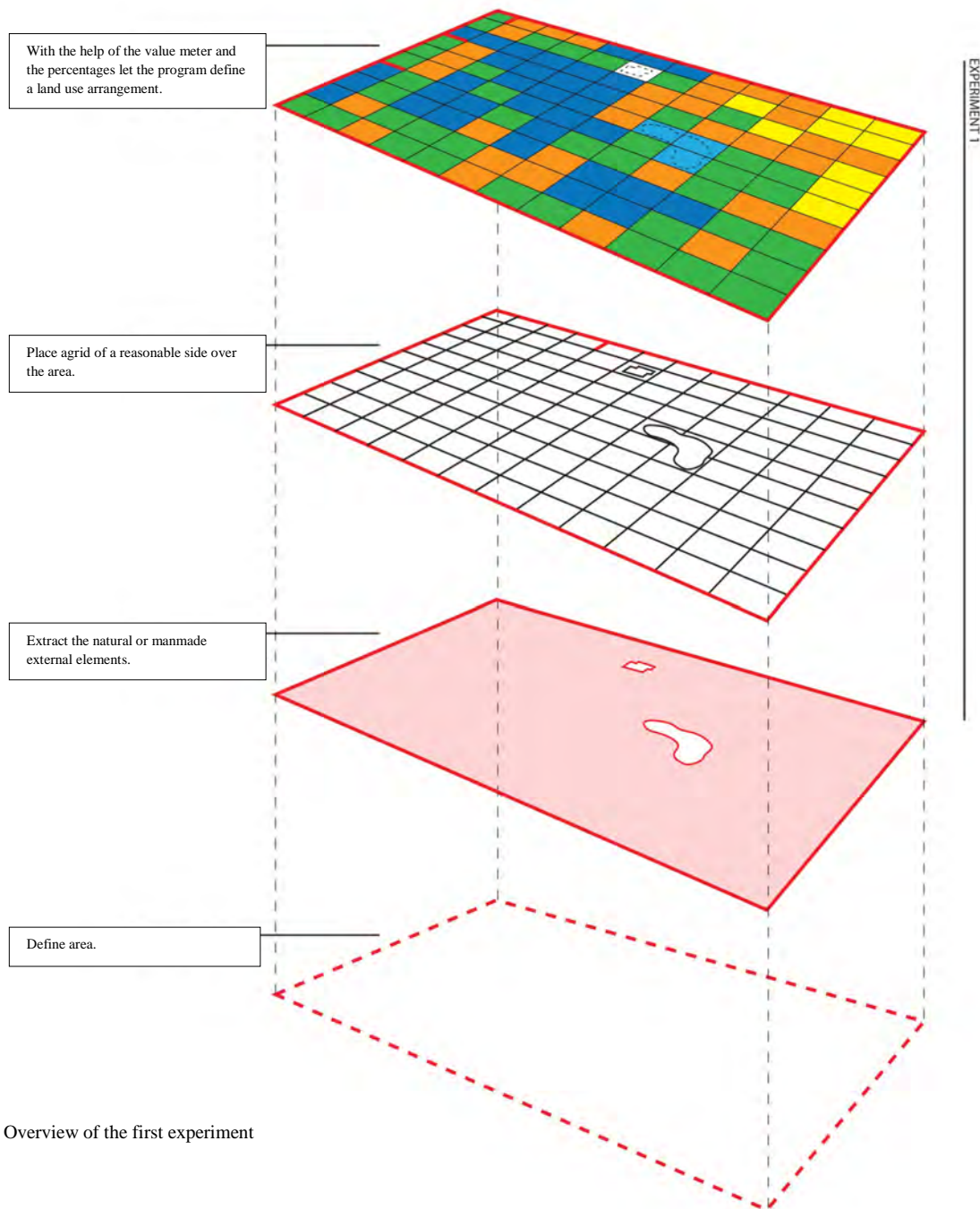


Figure 7. Overview of the first experiment

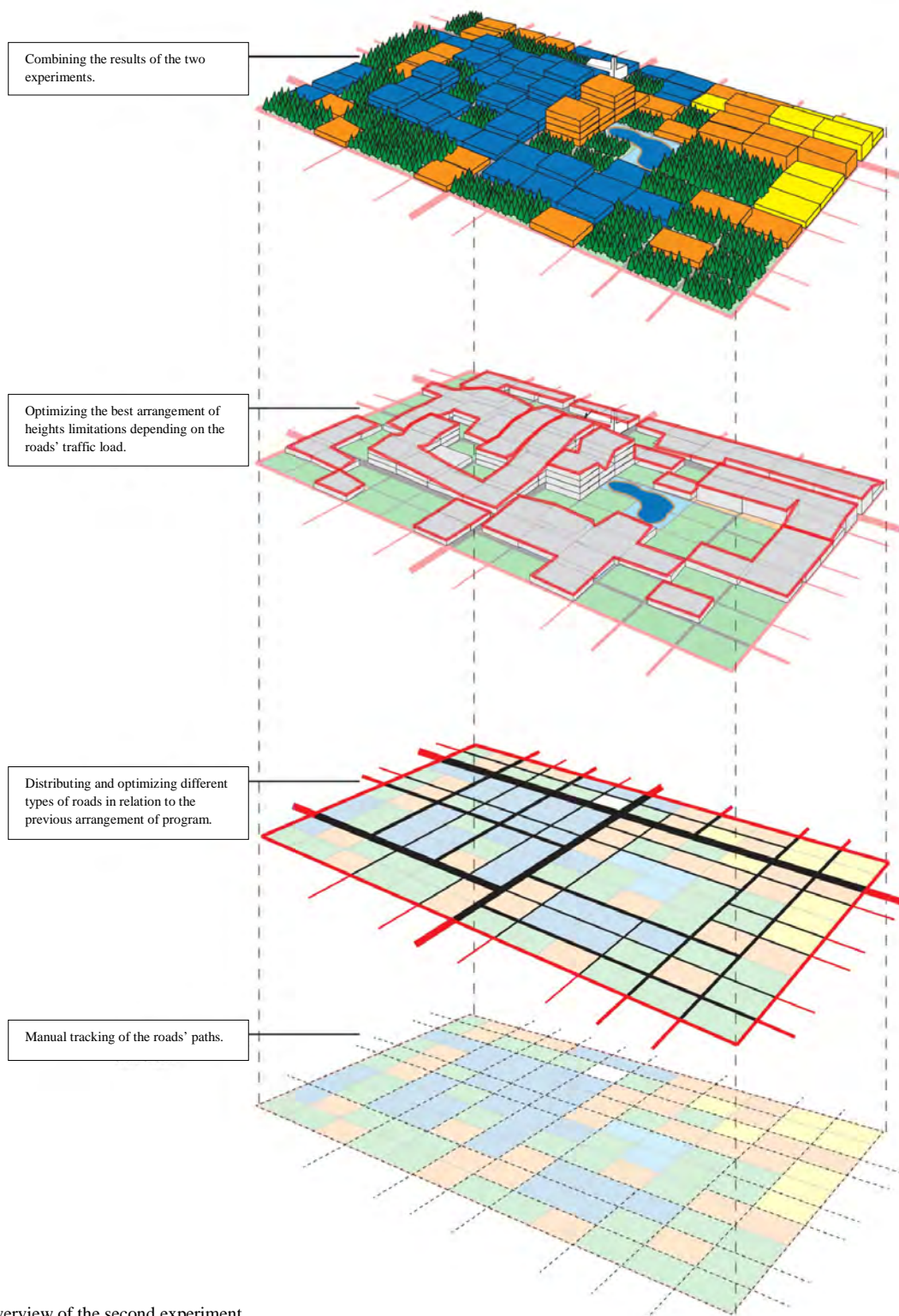


Figure 8. Overview of the second experiment

Components for Parametric Urban Design in Grasshopper. From Street Network to Building Geometry.

Christian Schneider¹, Anastasia Koltsova², Gerhard Schmitt

ETH Zurich

Wolfgang Pauli Strasse,

Zurich, Switzerland, 8093

¹christian.schneider@arch.ethz.ch, ²koltsova@arch.ethz.ch

Keywords: Urban Planning, Parametric Design, Street Network

Abstract

The main contribution of our work is in combining the methods for parametric urban design of highly specialized software such as CityEngine and general-purpose parametric modeling platform such as Grasshopper. Our work facilitates and prompts the use of parametric tools by architects and planners for urban design. In this paper we present a custom grasshopper component for street network generation and block subdivision. The component was developed in C# using the RhinoCommon SDK. We used Grasshopper for the development of an urban design proposal at a teaching exercise. To meet the requirements of the urban design project, additional functionalities had to be added to the range of existing Grasshopper components. In particular, we needed components for street network generation and block subdivision. To develop the component we implemented the street expansion strategies described in (Weber et al., 2009) and the methods for block subdivision described in (Vanegas et al., 2009). Additionally, we adapted and enhanced the strategies to meet the NURBS modeling capabilities of Rhinoceros.

1. INTRODUCTION

The process of urban design involves working with quantitative parameters that define elements constituting the design (e.g. buildings, open spaces etc.) and relations between these elements. In order to find an optimal solution for the configuration and arrangement of these elements, multiple scenarios must be generated and evaluated. Using parametric design methods it is possible to declare parameters of the elements. By assigning different values to the parameters one can generate different configurations.

Mathematical equations are used to describe the relationships between elements in a parametric model. By associating information from the local context (building heights, proximity to the points of interest etc.) to the elements of the design model we can place the design within the local context. Once defined, a parametric model can be used to generate multiple design variations in a relatively short period of time.

Most of the 3D modeling software used by architects today such as Rhinoceros, Maya or 3D Studio Max by default do not provide methods to handle contextual and site information required to support an urban design process. At the same time, software such as CityEngine - originally targeted at a different domain i.e. gaming and movie industries - presents various possibilities for creation of parametric urban environments. CityEngine is a procedural modeling tool based on shape grammars and targeted at the creation of large-scale urban environments. Buildings and streets are created parametrically via the software's interface or from scratch through programming (Steino N., 2010). The implementation of methods from CityEngine within design software such as Rhinoceros and Grasshopper enables designers to take advantage of useful urban design strategies and algorithms. Coupled with NURBS modeling possibilities provided by Rhinoceros this approach allows designers to freely perform formal explorations.

In our approach, parametric tools act as an aid for the design process. We develop additional Grasshopper components that solve problems at certain stages and scales of the design project. Therefore it is more useful to use a general-purpose parametric modeling environment such as Grasshopper. This is a disintegrative approach as described in (Derix, 2009) as opposed to the currently more dominant

paradigm of 'total integration', which can be observed in areas such as Building Information Modeling (BIM). The developed components integrate well with the existing components in Grasshopper. This allows using all the advantages offered by Grasshopper while also being able to integrate features otherwise only provided by expert software, in our case CityEngine. However, we do not aim to replace highly specialized software. This would be beyond our scope. Implementing many features of such software would lead to unacceptable performance bottlenecks within Grasshopper. In order to solve more complex urban design problems and to take full advantage of the features and performance of CityEngine, it would make more sense to communicate the latter through a Grasshopper component. The Geco component library that seamlessly interfaces Grasshopper with Ecotect shows that the connection between expert software and Grasshopper is not only possible but also opens a new range of possibilities such as feedback loops at an early design stage.

2. RELATED WORK

Our work is most closely related to parametric urban design. In our current research work within a teaching exercise we enhance methods for the parametric urban design that have been previously developed during a Master's research tenure. One of the main challenges that have been faced during the Master's design research was the lack of reasonable methods for street network generation and block subdivision. The recent works by (Weber et al., 2009) and (Vanegas et al., 2009) describe methods for street network generation and block subdivision respectively. Both methods are well documented. Moreover, they are both integrated in CityEngine and as they confront a large audience were extensively tested for usability. (Weber et al., 2009) describe their expansion strategy as embedded in an urban simulation environment. The position, length and orientation of a street segment are therefore not a product of coincidence. They follow the most promising parameters of an evaluation procedure. However, the presented work does not contain any evaluation procedure. Nevertheless it was proven in (Weber et al., 2009) that evaluation of the network is feasible and therefore an integration of the procedure would be possible in future enhancements. The block subdivision procedure of (Vanegas et al., 2009) was tailored for user interaction. Its behavior is consistent when facing user interaction. It allows local control and flexibility in the subdivision process. The exposure to a large

audience, perspectives for improvement, thorough documentation as well as flexibility lead to the decision of selecting both of the presented strategies.

Software which we use during the design process, i.e. the Grasshopper plug-in for Rhinoceros, does not provide methods to solve specific urban design problems. However, Grasshopper is a general-purpose parametric design platform which allows the extension of the existing component range with C# or VB .NET using the RhinoCommon SDK. Thus, it was possible for us to develop new components for street network generation and block-subdivision based on the work of (Weber et al., 2009) and (Vanegas et al., 2009). Both methods from (Weber et al., 2009) and (Vanegas et al., 2009) are implemented within CityEngine. In our opinion, the implementation of methods from CityEngine within a modeling environment that is commonly used in architectural design practice such as Rhinoceros would facilitate the use of powerful urban design tools by architects and urban planners.

In the remainder of this paper we describe results of the Master thesis design research - which will serve as a benchmark for the actual research work. In the next section we explain the implementation of (Vanegas et al., 2009) and (Weber et al., 2009) within Grasshopper. Then we illustrate the results (and possible optimization strategies) with an example of our students' work and a case study. We conclude with the discussion of results and our intentions for the future work.

3. PREVIOUS RESULTS

The goal of the research work performed during the Master tenure was to develop a parametric urban design system that would use information from the local context to drive the proliferation on the project site in Moscow, Russia. A series of computational tools using Rhinoscript was developed during the research. These tools use distance to the influencing elements as a driver to achieve differentiation and to articulate building geometry on a project site. The base setup includes a grid of points and an influencing element. The influencing elements become the immediate context of the site such as transportation nodes, park, roads and river. The base setup includes a grid of points and an influencing element (Figure 1, a). The variety offered by the computational tools includes shifting the position of points on the grid, creating connections with its neighboring points, scaling of elements and rotation all in

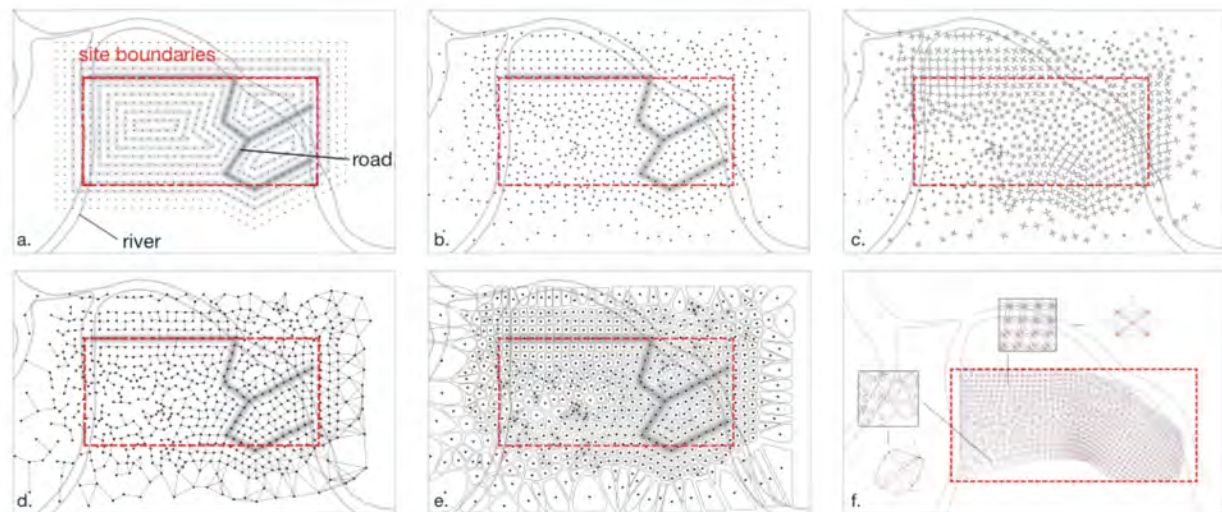


Figure 1. a. Initial point grid setup b. Point shift c. Orientation d. Connectivity e. Voronoi subdivision f. Geometry in Voronoi cell

proportion with the distance value of each point from the influencing element (Figure 1, a,b,c,d). Voronoi is used as a method for space subdivision (Figure 1: e). Each Voronoi cell obtains additional geometric information, which defines a form of a future building (Figure 1, f). These tools are applied to the project site in Moscow, Russia. The application of the computational tools resulted in the urban design proposal illustrated in Figure 2.

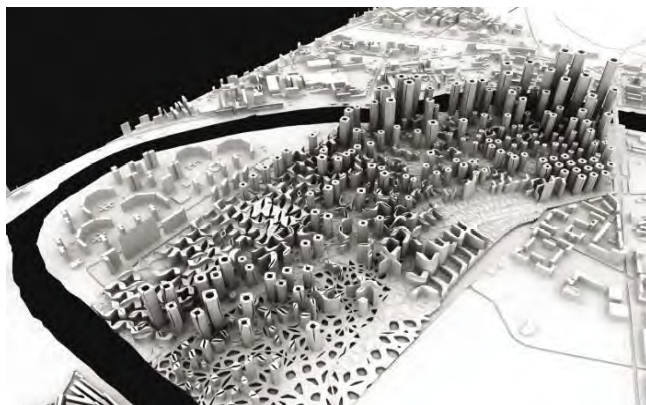


Figure 2. Final urban design proposal

The results of the Master research project proves the ability of computational tools to generate a great variety of building forms differentiated in response to the local context. However, the computational design method developed during the Master tenure has to be improved in order to be applied in real-world urban design practice. One of the main weaknesses of the proposed method is the

absence of a convincing method for street network generation. The computational tool, which generates connections between points does not reflect any real-world information except for the proximity to certain influencing elements. It is neither possible to control the line length that connects the points nor the angle between lines. At some spots overlaps occur and the pattern has to be altered manually. Moreover, in our project a point defines the location of a building, which means that the connectivity lines bump into the building. Thereby we used the connectivity method as an analysis tool to illustrate the accessibility on the site. However, this method cannot be used to produce an actual road network. The same applies to Voronoi cell boundaries, which in fact represent the boundaries of the building geometry, hence can serve as a street network. Nevertheless, the resulting pattern is far from being realistic and can hardly be used for real world design tasks.

Taking further the idea of using local context as a driver for proliferation on the project site we organized a teaching exercise. During this exercise the students were asked to develop an urban design proposal using parametric tools for a university campus in India. They had to investigate local topography, climate conditions, existing infrastructure and so on and to extract parameters that would control the urban design. Due to time limitations it was not possible for the students to develop the final urban design proposal. Therefore, they were combined in small groups, each group concentrating on a specific topic. The generation of a street network was one of the major tasks as it was a starting point for the design project. It was also a point of interest to

investigate further in the Master's research work. We developed our own Grasshopper component library for street network generation and block subdivision based on (Vanegas et al. 2009) and (Weber et al. 2009) which was used by the students during their work on the design project. In the next sections we will discuss the details of each of the custom components and will share some results of the student research.

4. COMPONENT LIBRARY

C# in Visual Studio 2008 in conjunction with the RhinoCommon SDK and Grasshopper base libraries were used to develop the presented component library. It consists of four components: three street network generation components and one block subdivision component. Each component provides a set of additional input parameters, which are described in section 4.1. and 4.2.

4.1. Street Network

The strategy used for the street network generation is largely inspired by (Weber et al., 2009) and adapted to the capabilities of the RhinoCommon SDK. The RhinoCommon SDK provides support for NURBS. Therefore it was possible to extend the functionalities of the algorithms to support the latter. Following input parameters are shared among the three street network components (Figure 3):

- *minLength*: The minimal length of a street segment
- *maxLength*: The maximal length of a street segment
- *deviation*: The mean deviation angle applied to the desired direction of the street
- *minDist*: If two junctions are within this distance, they snap together. This avoids very close, undesired junctions
- *growthSteps*: A number of growth steps
- *seed*: Allows for setting of different seed values in order to create different solutions based on the same input
- *startPoints*: A set of points from which the street network starts to grow

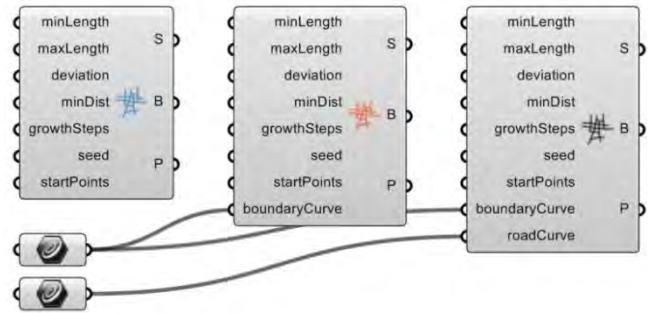


Figure 3. Three street network components

By default, all input parameters are set to meaningful initial values. This results in a simple street network. Since Grasshopper allows users to get real-time feedback on parameter changes, one quickly gets an understanding of how variations influence the outcome. While the above parameters are available in all three street network components, two of them include additional inputs:

- A curve representing the boundary of the project site.
- A set of curves representing existing roads to be integrated in the street network

All three components provide the same output parameters:

- The generated street segments as a set of line-like curves.
- A set of closed curves representing urban blocks.
- The corner points of the street blocks grouped correspondingly to the block they belong to.

Street segments can be used to build street profiles. Block boundaries can be used for a further subdivision with the corresponding component to derive a building geometry. Corner points of the third output can be used to articulate the open space and build geometry at a more local scale.

On a logical level the street network is represented as a set of street segments (edges) and junctions (nodes). In terms of Graph Theory, a graph can be described as $G = (E;N)$, where E is a set of edges and N a set of nodes. While a publically available graph library (Quickgraph) was tested, the implementation uses a custom graph library to stay lightweight in terms of memory and dependencies.

4.2. Block Subdivision

The block subdivision component provides a way to subdivide closed curves from the street network into sub-regions.

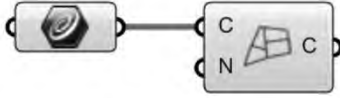


Figure 4. Block subdivision component

The implementation is based on the algorithm described in (Vanegas et al., 2009) for block subdivision. According to (Vanegas et al., 2009) the algorithm represents blocks as polylines. The algorithm is adapted to support the NURBS curves. It proceeds by taking a closed base curve, usually the one representing a block. The block is then subdivided into two curves, which together form the shape of the base curve. The same procedure is recursively applied to the resulting sub-curves in order to create a further subdivision. We use the proposition of (Aliaga et al., 2009) to use oriented bounding rectangles as guide shape for the process. The minimal bounding rectangle is calculated. Further, the bounding rectangle is divided at the mid section of the longer side. The section line is used for dividing the corresponding curve into two sub-curves. In order to find the minimal oriented bounding rectangles of a polyline, an adaption of the rotating calipers algorithm was applied (Toussaint, 1983). While the rotating calipers algorithm uses polyline edges to calculate the bounding rectangles for each iteration, different angle offsets from the base orientation were used to calculate a number of oriented bounding rectangles. The one with the smallest area forms the minimal bounding rectangle. The algorithm is illustrated in Figure 5.

Two inputs are required to define a subdivision process:

- A closed curve
- A natural number representing the number of division steps

The process results in 2^n curves, $n > 0$ being the number of division steps.

Algorithm 3 Subdivide curve

Require: curve representing street block c

Require: subdivision steps n

```

if  $c$  is a closed curve then
   $l_a \leftarrow \{c\}$ 
  for  $i \leftarrow 0$  to  $n$  do
     $l_b \leftarrow \{\}$ 
    for all  $c \in l_a$  do
       $l_c = \{\}$ 
       $b \leftarrow \text{minimalBoundingRectangle}(c)$ 
       $s \leftarrow \text{divisionSegment}(b)$ 
       $s \leftarrow \text{extend}(s)$ 
       $l_c \leftarrow \text{divideCurve}(c, s)$ 
       $l_b \leftarrow l_b \cup l_c$ 
    end for
     $l_a \leftarrow l_b$ 
  end for
  return  $l_a$ 
end if

```

Figure 5. Subdivide curve

In the next section we present the work of our students and a case study, which illustrate possible implications of the component library in the design process.

5. RESULTS

5.1. Student work

Two students, Matthias Knuser and Tobias Wullschlegler, used the component library in conjunction with other component libraries in Grasshopper to develop an urban design proposal for the university campus. Based on a terrain height and slope analysis they, yet manually, define primary roads. The location of major functions, such as the convention center, faculty buildings, housing as well as sport centers, is defined following the analysis part. Accompanying functions distribute independently around the major functions, e.g. faculty lounges near faculty offices; cafeterias near student rooms, lecture halls, offices, conference areas, etc. In order to distribute these functions, the whole organization diagram was translated into a spring-mass-system (Kangaroo) where the spring length represents desired distances between the core functions. A secondary street network is generated with the street network component taking into consideration slope analysis and functional distribution. The A* path-finding algorithm is used for finding the shortest paths interconnecting the major functions. Path lengths are used to evaluate the fitness of a road network. Evolutionary algorithms (*Galapagos*) are used to optimize the solution (Figure 6).

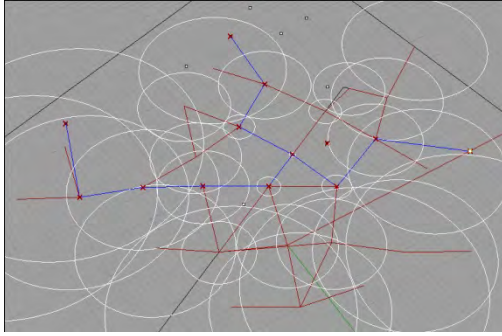


Figure 6. Shortest paths between major functions

5.2. A case study

The following case study illustrates the application of the developed component library for urban design. We demonstrate how to address local factors such as site boundaries, natural constraints (parks, lakes, forests) and existing infrastructure. Dimensions and parameters used in this case study do not reflect any real world parameters. They merely serve as a proof of concept to show how a parametric urban layout can be set up.

Figure 7 illustrates a possible initial setup with existing roads in black, forest / park area in green and a body of water in blue. The red closed curve serves as site boundary. The generated streets remain within this boundary. Two points serve as distinct starting points for the network generation. Different street lengths and deviation values are assigned to each of the points.



Figure 7. Initial setup

This results in two distinct network topologies. These topologies automatically merge during the generation process. Moreover, existing roads and site boundaries are taken into account when the street network is generated. The site boundaries are altered by calculating the region difference between site boundaries and the forest on the right. This prevents the generation of streets in the forest. Figure 8 shows the generated network.

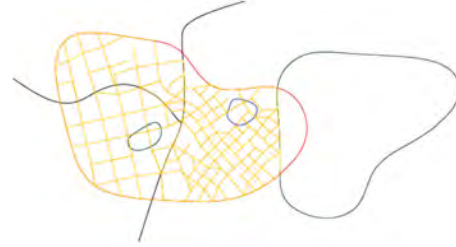


Figure 8. Generated street network

Existing roads are seamlessly weaved into the network and the latter stays within the site boundaries whereas park and water body are excluded. The *CurvesSplitIntersect* component of the *StructDrawRhino* plug-in helps to split the roads at the park and water body boundary respectively. Road segments within these areas are removed with appropriate Grasshopper components. The boundaries are integrated into the network and serve as block borders (Figure 9).



Figure 9. Altered street network

In the next step we define three different areas (Figure 10), which can represent areas with different land values. On an abstract level they represent three different regions, which in this case are used for informing the lot area sizes.



Figure 10. Areas with different land values (in pink)

Surface areas of the existing street blocks are therefore measured. Then, considering area type, the number of subdivision steps is estimated to achieve a lot area close to already defined goal area. The block subdivision component

then splits the block area into lots. A mixed usage is attained in the overlapping areas (Figure 11).

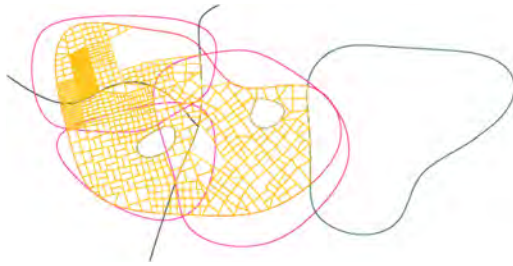


Figure 11. Block subdivision according to the area

A parameter representing the gross floor area is then assigned to each of the areas. From that point, the floor area for each lot is calculated. This informs the building height (Figure 12). Building colors reflect the building intensity described in (Berghauser and Haupt, 2010). The presented parametric urban system reacts to changes of initial conditions such as site boundaries, position, size and number of water bodies or parks, shape and number of existing roads. While the system responds to these changes, it seeks to maintain desired gross floor areas for the different functional regions. The parametric nature of the model allows testing different scenarios within a short period of time. While Grasshopper is intrinsically parametric, the component library for the street network generation and block subdivision add the missing link to create urban layouts in Grasshopper. Figure 13 illustrates the combination of Rhino modeling features together with a generated street layout.



Figure 12. Building volumes



Figure 13. Rhino geometry

6. CONCLUSION AND FUTURE WORK

We presented a method for parametric street network generation and block subdivision in Grasshopper. The possibilities of street network generation and block subdivision allows the user to combine procedural street generation and parametric design abilities of Grasshopper.

Performance of the street network component must be further optimized. One reason for unsatisfying performance on a technical level is the representation of the street network as a directed graph as well as an undirected graph. The latter is used for determining the cycles representing street blocks. Double implementation leads to performance bottlenecks and large memory use when applied at bigger urban scale. We managed to utilize the component for urban designs at the scale of a university campus (approx 80 hectares), but its application at the larger scale is not yet possible in Grasshopper. Another point for improvement is the cycle finding process. For certain street network configurations we fail to find all the cycles, which form street blocks. As a result, it is difficult to include an entirely parametric model on the block level, which would react to changes of the street network. A method to find all the required cycles in a consistent manner has to be introduced.

By combining both the components for street network and block subdivision as well as other Grasshopper components, we were able to create hierarchical patterns. The incorporation of a standard graph library would allow the application of graph algorithms such as Dijkstra, Prim and Kruskal (Dijkstra, 1959; Prim, 1957; Kruskal, 1956). This would open up a new range of possibilities such as calculation of shortest paths and distances between points along the street network. In section 5 we presented an attempt to calculate the shortest path using the A* path-finding algorithm. Important urban indicators such as

network density and other indicators described in (Berghauser and Haupt, 2010) could be directly calculated.. Direct evaluation and comparison of different solutions would be possible.

The subdivision component generally works well. Up to ten subdivision steps leading to 1024 subcurves still maintain the interactivity within the performance boundaries imposed by Rhino and Grasshopper. The recursive nature of the algorithm generates subcurves of order 2^n . Alternative methods must be developed to achieve a variety in both subcurve geometry and their number.

In our future work we will concentrate on the development of evaluation and optimization methods for the street network generated by the components. All components currently work on a plane. It would be useful to incorporate terrain information, e.g. to create more meaningful networks by taking into consideration the slope/landscape variations. The introduction of local context information as illustrated in the case study or the shortest path algorithms applied by students are examples of optimization strategies.

References

- ALIAGA D. G., VANEGAS C. A., BENES B., 2008. Interactive example-based urban layout synthesis. *ACM Transactions on Graphics* 27, 5, 1–10
- BERGHAUSER PONT M., HAUPT P., 2010. *Spacematrix: Space, Density and Urban Form*, NAI Publishers, Rotterdam
- CITYENGINE (PROCEDURAL INC. 2010). Official Website: <http://www.procedural.com/>, Internet Accessed: 09.11.10
- DERIX C., 2009. In-Between Architecture Computing, *International Journal of Architectural Computing*, Issue 04, Volume 07, pp 565-585
- DIJKSTRA E. W., 1959. A note on two problems in connection with graphs, *Numerische Mathematik* 1: 269–271
- GECO ([UTO]), Official Website: <http://utos.blogspot.com/>, Internet Accessed: 19.11.10
- GEOMETRY GYM (GEOMETRY GYM LTD.), Official Website: <http://ssi.wikidot.com/> Internet Accessed: 14.1.11
- HART, P. E.; NILSSON, N. J.; RAPHAEL, B., 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, SSC4, 4 (2): 100–107
- KANGAROO (PIKER D.), Official Website: <http://spacesymmetrystructure.wordpress.com/>, Internet Accessed: 14.1.11
- KRUSKAL J. B., 1956. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proceedings of the American Mathematical Society*, Vol 7, pp. 48–50
- MAYA AUTODESK, Official Website: <http://www.autodesk.de/adsk/servlet/pc/index?siteID=403786&id=14657512>; Internet Accessed: 01.11.2010
- MCNEEL, R. 2010. GRASSHOPPER - GLTERNATIVE MODELING WITH RHINO, McNeel North America, Seattle, USA. <http://www.grasshopper3d.com/>, Internet Accessed: 09.11.2010
- MCNEEL, R. 2010. RHINOCEROS – NURBSA MODELING FOR WINDOWS (VERSION 4.0), McNeel North America, Seattle, W A, USA. www.rhino3d.com/, Internet Accessed: 11.11.2010
- MICROSOFT .NET FRAMEWORK, Official Website: <http://www.microsoft.com/net/>, Internet Accessed: 19.11.10
- MICROSOFT VISUAL STUDIO, Official Website: <http://msdn.microsoft.com/vstudio/>, Internet Accessed: 19.11.10
- STANDARD ECMA-334 - C# LANGUAGE SPECIFICATION (ECMA International, 2006), <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-334.pdf>, Internet Accessed: 19.11.10
- TOUSSAINT G., 1983. Solving geometric problems with the rotating calipers, *Proc. MELECON '83*
- PRIM R. C., 1957, Shortest connection networks and some generalizations, *Bell System Technical Journal*, pp. 1389–1401
- QUICKGRAPH, GRAPH DATA STRUCTURES AND ALGORITHMS FOR .NET, Version 3.3, 2010, Official Website: <http://quickgraph.codeplex.com/>, Internet Accessed: 19.11.10
- RHINO .NET FRAMEWORK SDK, Official Website: <http://wiki.mcneel.com/developer/dotnetplugins>, Internet Accessed: 19.11.10
- RHINOCOMMON SDK, Official Website: <http://www.rhino3d.com/5/rhinocommon/>, Internet Accessed: 19.11.10
- STEINO N, 2010, *PARAMETRIC THINKING IN URBAN DESIGN*, Paper for the ASCAAD Conference
- VANEGAS C. A., ALIAGA D. G., BENEŠ B., WADDELL P., 2009. Visualization of Simulated Urban Spaces: Inferring Parametrized Generation of Streets, Parcels, and Aerial Imagery, *IEEE Transactions on Visualization and Computer Graphics*, 15(3), pp. 424-435
- VORONOI DIAGRAM, Wikipedia: <http://de.wikipedia.org/wiki/Voronoi-Diagramm>, Internet Accessed: 10.11.2010
- WEBER B., MUELLER P., WONKA P., GROSS M., 2009. Interactive Geometric Simulation of 4D Cities, *EUROGRAPHICS*, Volume 28, N 2

Multi-Objective Optimization in Urban Design

Michele Bruno¹, Kerri Henderson², Hong Min Kim³

¹Columbia University
1172 Amsterdam Ave.

New York, New York, USA, 10027
mb3408@columbia.edu

²Columbia University
1172 Amsterdam Ave.

New York, New York, USA, 10027
kh2388@columbia.edu

³Columbia University
1172 Amsterdam Ave.

New York, New York, USA, 10027
hk2601@columbia.edu

Keywords: Optimization, Catia, Mode Frontier

Abstract

Urban Design is a multi-objective task. Traditionally, urban spaces are designed hierarchically; organizational inputs are idealized uniquely, and negotiated through sequential overlay. In our investigation, parametric modeling (with the software application *Catia*) and evolutionary optimization employing genetic algorithms (with the software application *Mode Frontier*) enable the exploration of a non-linear design space whereby multiple objectives may be optimized concurrently. This paper describes an experiment that builds from prior research in multi-objective *optimization* of architectural design and applies that workflow to multi-objective *optimization* in urban design. The experiment employs given constraints, custom procedural algorithms and genetic algorithms to examine a wide design space and identify designs that perform well in multiple arenas. Design, data and latent influences are exposed and negotiated quantitatively to render topological variation through *optimization*. By using multi-objective *optimization* we define and apply quantitative metrics in order to examine the potential for a new workflow in urban design.

1. INTRODUCTION

The context of our research elaborates on the recent work of David Benjamin and Ian Keogh in Multi-Objective Optimization in Architectural Design (Keough and Benjamin 2010). Benjamin and Keough created an automated workflow that linked parametric modeling (*Catia*), structural analysis through a custom-designed software (Catbot) and a multi-objective *optimization* engine (*Mode Frontier*). Their workflow was used to compute multiple architectural design permutations and aid as a tool

to evaluate those designs. Building on their investigations and gained knowledge, we hope to broaden the potential influence of this workflow to include urban design. Our research does not integrate the structural analysis loop but uses parametric software and a multi-objective *optimization* at the scale of the city. Our experiment abstracts buildings to basic geometric primitives such as cylinders in order to study programmatic and spatial relationships. We outline the importance of the definition of metrics for the success of the experiment and discuss how explicit metrics could influence current practices in urban design.

Urban design concerns the arrangement, design and functionality of cities. The discipline traverses many fields and interests such as architecture, urban planning, construction, politics, economics, real estate development, environmental systems and social theory. In some cases, urban design is influenced disproportionately by one or more of these fields, or by a particular stakeholder. In other cases, early decisions may have a much stronger influence than later decisions. In yet other cases, decisions may be made to satisfy each objective in sequence, which rules out some possible design results. As an alternative to these examples, we propose a workflow of multi-objective optimization in which many design criteria are evaluated simultaneously, with relatively equal influence.

Optimization software computes a parametric model through its range of possible permutations to find a set of high-performance designs. The application of this technique is novel in the context of urban design. In engineering, architecture and product design, *optimization* is often tied to simulation software such as finite element analysis (Kicinger et al. 2005). Inputs are identifiable and quantifiable; permitted tolerances are determined specific to the project, or are taken from known rules of thumb. Using primarily known materials, practices and tolerances for

inputs, the workflow often produces designs that are both novel and high performance (Koza et al. 2003).

Often in urban design, projects are developed hierarchically; organizational inputs are idealized uniquely, and negotiated through sequential overlay. Complex problems in urban design may present multiple primary design factors to multiple invested parties (Galster et al. 2001). The strength of the computational process is the software's ability to evaluate multiple objectives concurrently and render a range of high-performance designs.

There are many quantitative factors to be considered in the urban design process. Zoning; program; density; solar gain; shadow projections; wind velocity, location to city service points for energy, water, and waste collection; traffic flow and projected economic revenue are just a few of the factors involved in the process. Furthermore, there are often qualitative factors that are addressed in urban design; they can include quality of life, cultural distinction and aesthetics. These qualitative factors require metrics for design and critical evaluation. Urban design lays the foundation for the new buildings, public spaces and services that shape our lives.

New technologies enable new workflows to address the complexity of urban design projects. Automated genetic algorithms have been used to exploit parametric permutations by generating, evaluating and improving the performance of possible design options (Keller 2006). This workflow is not geared toward a specific task; it is a tool to aid reflective, responsible design practice.

2. WORKFLOW

Our workflow begins with a set of constraints, generates design permutations through custom procedural algorithms and evolves high-performing designs through genetic algorithms. Beginning with design constraints is a familiar launching point for architects and urban designers. Given constraints can include the site, existing infrastructures, services, budgets and legal parameters. Identifying and drawing the given constraints create the initial environment in which to operate.

Inputs are identified through conversation with involved parties. An input is any quantifiable factor, specified by an acceptable range that would support a desired state or objective. For example: Input: building height range 50'–

75'. Ideal building height = 75'. Objective: maximize building height.

Once given constraints and inputs have been identified, they may be drawn or modeled digitally using parametric software (*Catia*). The custom procedural algorithm is the *Catia* script. The architect or urban designer writes this script. In doing so, he or she sets up the relationships between the inputs and the parameters that can affect their values. The designer may also set up rules to further articulate relationships between design parameters (example: when x is 2, y is 0.5x). The role of the designer is to identify and create the morphological identity of the inputs. Using the custom procedural script, the designer builds the domain of influence of the genetic algorithm, setting the breadth of the potential design space (Figure 1).

The design of a good experiment establishes clear design metrics, bases input parameters upon valid data, is procedural in its modeling techniques and enables the genetic algorithm to explore a wide design space through the custom procedural script.

Connected to scripting is the notion of State Change. State Change is a function of an If/Then condition. That is to say that if x is true, proceed with State A, if y is true, proceed with State B, etc. The State can effect morphological or topological changes in the design. State Change widens design space in that it enables the algorithm to explore possible relationships and design permutations that would not necessarily occur to the independent designer.

Once the model, parametric relationships and script are set, the *Catia* file is linked to the *optimization* software, *Mode Frontier*. *Mode Frontier* is an evolutionary computational software that employs a genetic algorithm as a search heuristic to generate multiple design permutations. It differs from stochastic search in that it learns from ratings of previous permutations. Specifically, our experiment employed the MOGA-II. MOGA-II is a scheduler based on a multi-objective genetic algorithm (MOGA) designed for fast Pareto convergence. MOGA II supports directional crossover, implements elitism, enforces user-defined constraints and allows Steady State evolution. *Mode Frontier* starts initially with a random population of input parameters. Through generational growth, it evaluates the results of each cycle to ultimately reach a set of designs that offer the best possible outputs for the objectives. For each permutation, *Mode Frontier* generates data sets that the

designer can then evaluate and compare with other permutations of the experiment.

Given Information - Existing conditions
 Custom Procedural Algorithm -Catia scripting
 Genetic Algorithm - Mode Frontier

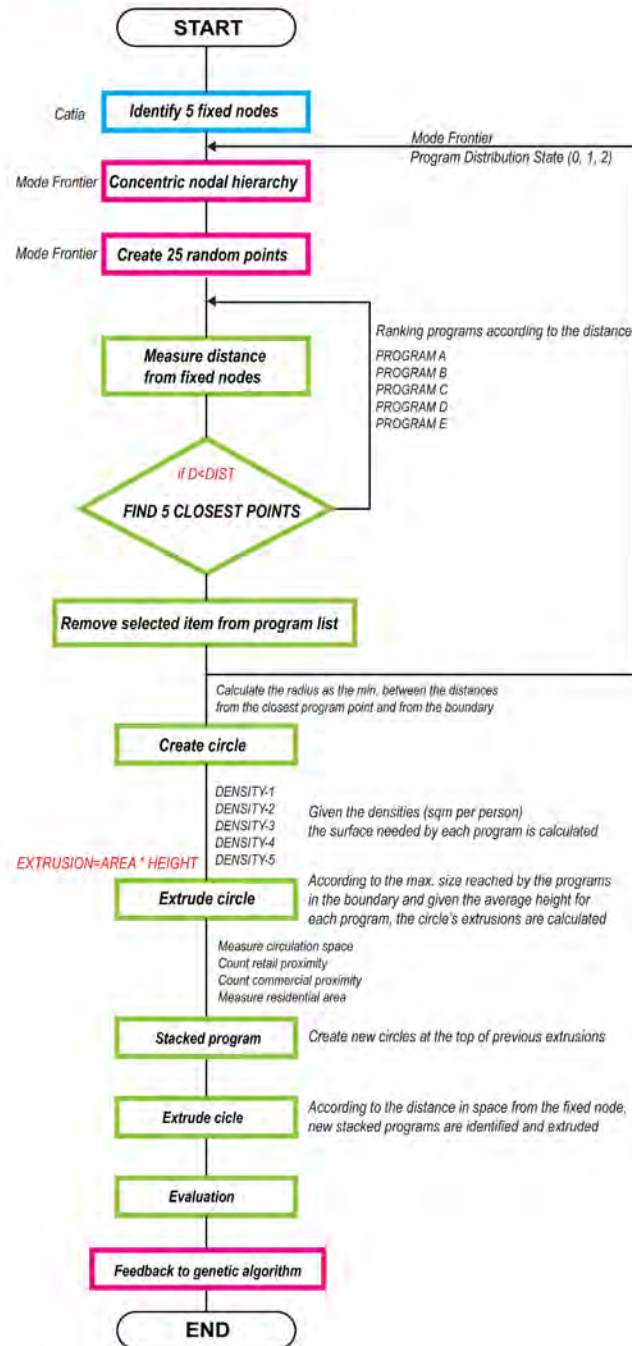


Figure 1. Workflow diagram specifying each agent's role in the process.

Within *Mode Frontier*, objectives are set for the experiment. An objective is a value to which the outputs should be optimized. It can be set to “minimize” or “maximize” global parameters, or at specific target. The objectives articulate the purpose of the experiment. When defining metrics for the objectives, it is possible to create conditions that produce competing objectives. Competing objectives are ones where the conditions champion the maximum performance of one objective and diminish the performance of another. Establishing the objectives involves design decisions that are as crucial as the design of the inputs.

In setting up the experiment, it is the role of the designer to clearly construct the metrics by which the inputs are evaluated. In the case of urban design, objectives can be based on known rules of thumb. Objectives may also be developed as a way of quantifying less mathematical inputs such as quality of life or aesthetics. Often both types of metrics play a role in the experiment (DeLanda 2002). Defining metrics to evaluate design creates a new workflow and design culture in many ways:

- (i) Each design must begin with the question: what are the necessary inputs for urban design? What does it take to plan a great city?
- (ii) It challenges the architect or designer to set a range of acceptable parameters for each possible case, identifying and expanding the definition of what makes a “good” design.
- (iii) The explicit definition of metrics lessens the importance of subjective preconceptions in the design process. Once rules are established, design evaluation can be more critical and thorough. This novel approach could identify high-performance designs that reach beyond established practices.
- (iv) By re-programming design methodology, this new workflow opens up conversation with representative stakeholders, designers, engineers, investors and community members early on.

To adequately address the numerous variables involved in urban design, the initial setup of experiment is paramount.

For our experiment, we decided to focus on five inputs that we believe to be influential for the urban design of

Masdar, UAE: program, density, proximity and mixed-use quality.

In terms of computational resources, we have found that with a PC computer (Intel® Core™ Quad CPU, 4GB RAM) running for 24 hours, with 5 inputs, we can evolve 1,500 design permutations. Increased inputs, model complexity and wider parameter ranges could warrant longer computation times, networked processing or the organization of multiple experiments.

3. MASDAR 2.0 (BETA)

Masdar is a new city being constructed 17 km east-southeast of Abu Dhabi. Masdar aims to be a highly efficient, sustainable, zero-carbon, zero-waste ecology development, relying entirely on solar energy and renewable resources (Adrian Smith and Gordon Gill Architecture 2010). Beginning Tabula Rasa, the design of the city invites not only questions of efficient operational practices but also an optimistic interrogation into the factors involved in creating an 21st century city. We hope to develop programmatic distribution for Masdar that not only can be evaluated by measurable criteria but also to create a new workflow and design culture.

This case study involves the use of two existing software applications: *Catia* and *Mode Frontier*. The morphology of the experiment is abstracted: points, circles, cylinders and color are used diagrammatically to represent relationships of program, density, proximity and mixed-use quality (Figure 2).

The procedure starts with a set of fixed nodes as given constraints. For this test the fixed nodes are based on the existing transportation system. These nodes are transit stations that have already been planned and constructed in Masdar. The previous master plan also used these hubs as a primary factor of influence.

In *Catia*, as part of the custom procedural algorithm, we generated a field of potential program locations. We used 100 possible locations. The spacing and number of nodes in this field are influential for the overall design output. It is the responsibility of the designer to set these constraints during the experiment.

The genetic algorithm in *Mode Frontier* creates 25 random points at the possible locations. The distance from all the points to all the nodes is measured through the custom procedural algorithm, and points are ranked in

accordance with their node proximity. A list is generated for each node of its sequentially adjacent five points.

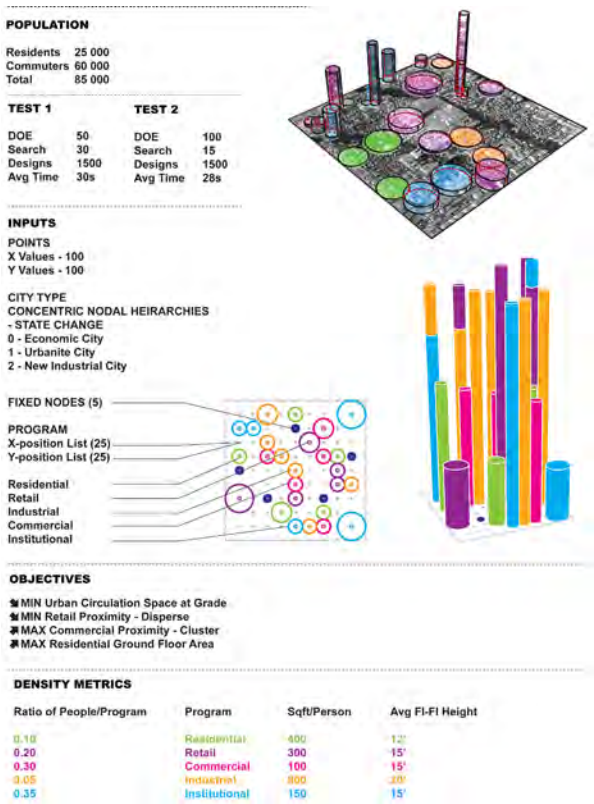


Figure 2. General setup of the experiment.

For the identified five programs or inputs, a nodal hierarchy is established by the designer (Figure 3). This determines which program should be placed closest to its node and thereon. This hierarchy can determine the conceptual base for the city. For our case study we defined three nodal hierarchies that were of interest to us. In *Mode Frontier*, State Change was employed to determine the nodal hierarchy of the given fixed point. This allowed the genetic algorithm to create a variety of urban spaces, widening the design space and enabling potentially unforeseen *optimizations* for our given inputs. Ratios of total program area of urban space to inhabitants were established. A set of rules was established for proximities between programs.

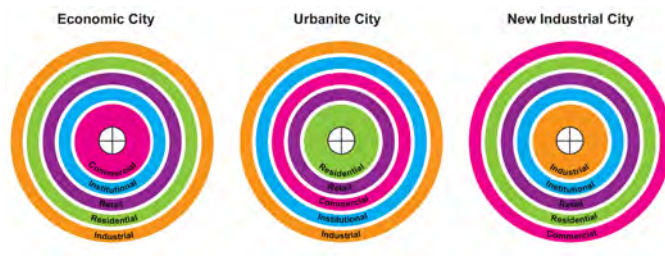


Figure 3. Nodal hierarchies determine the programmatic ordering sequence of geometry created during the experiment.

The proportion of programs within the same urban type is defined by stacking the new programs onto the previously extruded iteration. The hierarchy of programs is consistently valued according to the vertical distance from the transportation hubs (fixed nodes). This looping of programmatic distribution affects the degree of mixed-use program within a building and its neighborhood.

Open space is defined in the experiment as the space at grade that is not assigned to a building program. Open space includes public space, green space, right of way and all circulation space for vehicles and pedestrians.

Program and circulation space are simultaneously and jointly optimized. The custom procedural algorithm also aims to embrace a planning model that overcomes traditional 20th-century zoning. Taking the perspective of developers, neighborhoods are classified according to adaptability of program combinations to height and potential for economic development.

Following the algorithmic computation, *Mode Frontier* returns data on each permutation's inputs, outputs and objectives. This data can be used in *Mode Frontier* to generate 4D bubble graphs and data charts. Using both visual (4D bubble graphs) and numerical (data charts) data, the designer can look for trends and high-performance results within an iterative process. An Utopia Point is the point on the graph where all objectives would be idealized. A Pareto curve is the set of all best designs. The experiment plays competing objectives against one another. It may not be possible to idealize every objective, but rather to establish a range of designs that achieve high threshold of performance for multiple objectives. Here, the designer re-enters the design process to evaluate influence of the urban design factors and weigh the best designs. Unlike multi-objective *optimization* in engineering fields, we aim to produce a range of high-performance designs that may be further evaluated post-computation by designers. This

strives to champion the best possible design for the specific situation.

Our experiment returned a set of 62 best designs (Pareto designs). Each one achieved high performance for one or more objectives. Our initial objectives were to cluster commercial properties, disperse retail, maximize the ground floor area of residential properties and minimize the overall circulation space of Masdar at grade (Figure 4).

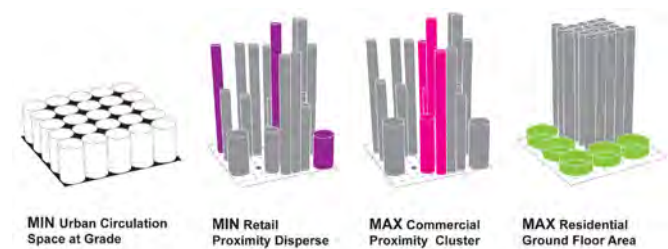


Figure 4. Predictions for each objective individually considered.

Upon examining the Pareto designs we found that three best-fit categories emerged: Best Clustering, Best Area Coverage and Best Overall. For Best Clustering, Design 1493 exhibited the closest proximity values of commercial properties and the best dispersement of retail properties specified in our objectives. The clumping of commercial and dispersement of retail created two identifiable business districts, though it performed less well in overall site coverage. For the Best Coverage category, Design 1244 covered more total area than any other design and showed the maximum ground coverage of residential program. Best Clustering and Best Coverage are competing objectives; it would be impossible to optimize 100 percent for both. Our ambition was to explore designs that performed as best as possible for these categories.

The Best Overall permutation, Design 1177, performed well in clustering. In this case, two dense commercial areas are apparent, retail is reasonably dispersed, and there is a high presence of residential program at grade, and a high total coverage of land area, as prescribed by our initial input of factors. Design 1177 was chosen as the best-fit design of the experiment (Figures 5, 6 and 7).

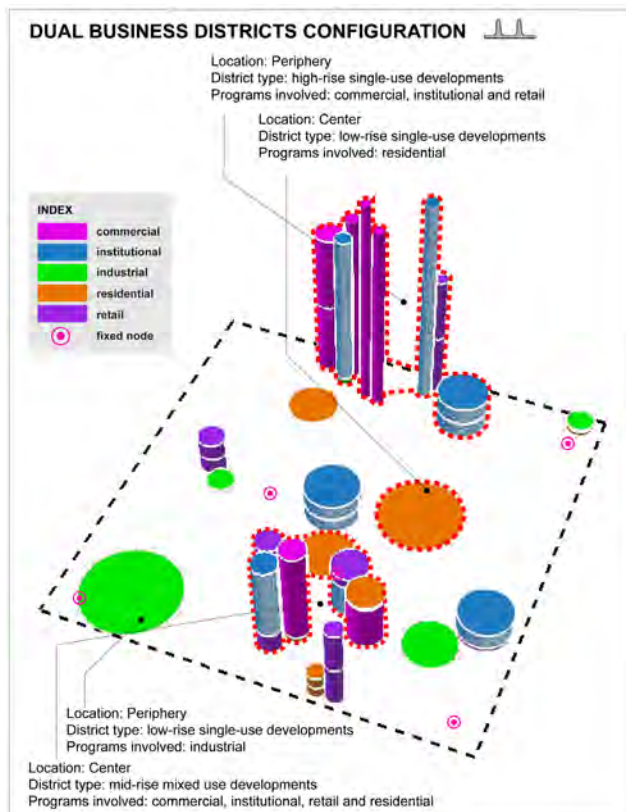


Figure 5. Design 1493: closer analysis.

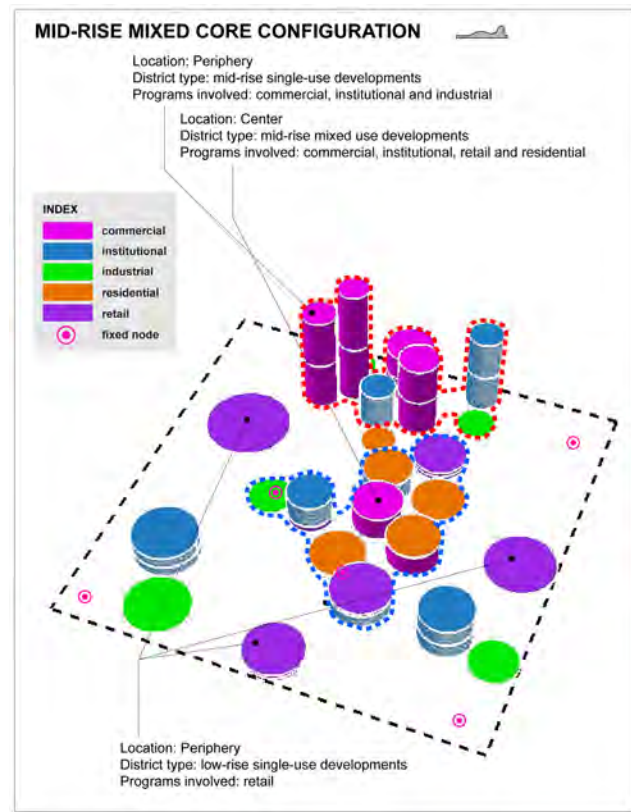


Figure 7. Design 1177: closer analysis.

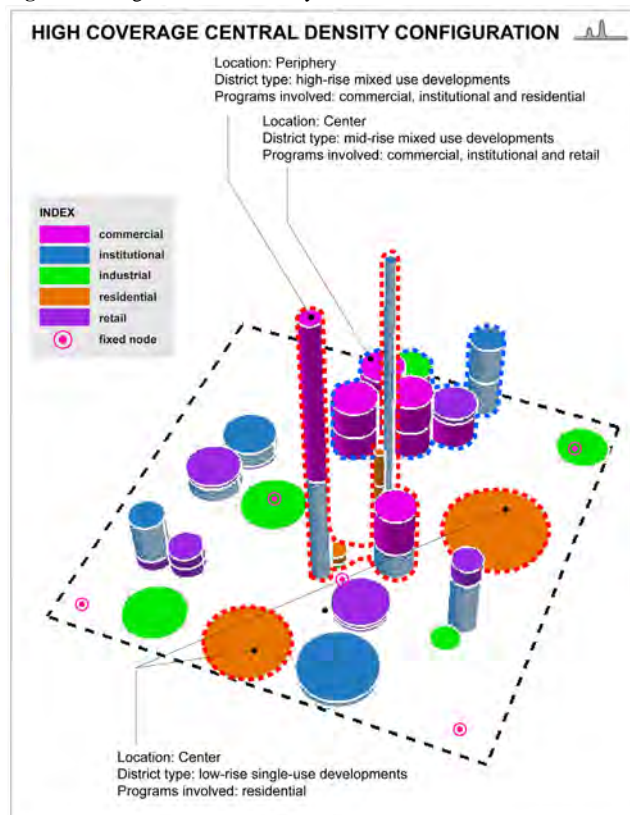


Figure 6. Design 1244: closer analysis.

The purpose of this experiment is to challenge traditional urban design hierarchies in order to obtain novel district types. The optimal design shows the presence of a traditional, high-rise district, with single-use buildings at the periphery with a mid-rise mixed-use development close to the center. Here, residential areas stack on top of institutional or commercial areas. Institutional areas stack on retail areas. A low-rise mixed-use city center that shows residential developments stacked on top of commercial or institutional development could be seen as a novel district arrangement. This solution also shows a gentle gradient in building aspect ratios between high, mid, and low-rise developments.

4. NEXT STEPS

Masdar 2.0 (beta) illustrates our initial set of experiments with programmatic *optimization* in urban design. We would like to extend this experiment to include different inputs within an *optimization* workflow. As part of our future work, we can consider additional types of inputs such as building, block and city morphology, circulation systems, city services, land value and potential economic *optimizations* as they relate to urban form and social inputs.

Each type of input will require the definition of a metric by which they will be evaluated. Each type of input can be tested to compare how the input will perform in an environment of multi-objective *optimization*. This process is ideal if automated-testing returns results that are otherwise unattainable by traditional processes. It is possible that certain types of inputs are more suited to this process than others. We would like to test not only how different types of inputs can be optimized, but also how each type could perform in a multi-type, multi-objective experiment.

In our research, we discovered two potential limitations to *optimization*. The first is that the design of a good experiment is crucial. The design inputs must be valid. The parametric model and custom procedural script must be designed to enable the genetic algorithm to explore a wide design space. The experiment must also be designed to methodically and realistically return convincing results. The second concerns computational power. Succinctness in modeling and in defining parametric relationships can help control the computational needs of the experiment. Exponentially more computational power is needed as accuracy increases, approaching reality.

We recognize that urban design is viewed through many different lenses and must perform according to various criteria. The type of input or comparison of types of inputs can frame the scope of the experiment. It is important to be able to create the ability to test for as many different factors as are involved in urban design so that the conversation surrounding the potential design can be inclusive.

5. CONCLUSION

Building on previous research, we have adapted, applied and automated an existing workflow for *optimization* in architecture to urban design. Using the experiment's unique given constraints, we designed a specific parametric model, custom procedural algorithm and *optimization* objectives to create a new type of design experiment and broaden the potential influence a multi-objective workflow to other fields. Our workflow was executed on typical hardware (PC computers) using existing software without any previous advance knowledge of scripting, Engineering Knowledge Language or the function of genetic algorithms. The workflow requires the experiment itself to be well designed in order to be an advantageous tool. Depending on the number of inputs, input ranges and influences, complexity of the model, possible States, and overall scope, the experiment will require a specific design. The authorship of

the design workflow is of great importance; it frames not only the parameters of the experiment but also the design project itself. The goal is to design a design space large enough to compute and return design possibilities that would otherwise be impossible to come to independently and to control the scope of the experiment so that it is possible to compute with typical hardware. The design of multiple experiments that build on one another is possible and interesting. The author must understand how factors relate to one another within a given experiment and throughout possibly multiple experiments. The experiment is only as good as the data and the design of the experiment itself.

The use of a functionalist algorithm may contribute to the development of a 21st century aesthetic. This new methodology does not rest uniquely on morphological output of data. The designer can define the metrics by which the data is evaluated, how that data is expressed and the possible ranges and potential interactions across the design space. The choice is to control more rigorously the factors that contribute to design practice and imagine new possibilities in design and workflow.

Automated *optimization* processes do not produce a single best design but a range of high-performance designs. The output data of the experiment must be evaluated and judged by the parties involved. High-performance results that are surprising tend to expose latent assumptions embedded in design culture. In this exploration we seek to open a conversation about what makes a great urban design within the context of measurable factors. A more collaborative, accountable and quantifiable methodology in urban design will change the way developments, neighborhoods and cities are built. We believe that by bringing explicit factors of urban design to the table we can rigorously discuss why a particular design or approach may be favored over another. By openly discussing design priorities we can learn how to create more ideal places to live, work and play.

Acknowledgments

This work was conducted under the guidance of David Benjamin with the help of Teaching Assistants Jesse Blanketship and Danil Nagy at Columbia University, School of Architecture, Planning and Preservation.

References

- KEOUGH, I., AND BENJAMIN, D. 2010. Multi-Objective Optimization in Architectural Design. Symposium on Simulation for Architecture and Urban Design, 1–9.
- KICINGER, R., ARCISZEWSKI, T. AND DEJONG, K. 2005. "Evolutionary Design of Steel Structures in Tall Buildings." *Journal of Computing in Civil Engineering*, vol. 19, no. 3 (July): 223–238.
- KOZA, J.R., KEANE, M.A. AND STREETER, M.J. 2003. "Evolving Inventions." *Scientific American Magazine*, February: 52–58.
- GALSTER, G., ET AL. 2001. "Wrestling Sprawl to the Ground: Defining and Measuring and Elusive Concept." *Housing Policy Debate*, vol. 12, issue 4: 1–37.
- KELLER, S. 2006. "Fenland Tech: Architectural Science in Postwar Cambridge." *Grey Room*, vol. 23, Spring: 40–65.
- DELANDA, M. 2002. "Deleuze and The Use of Genetic Algorithms in Architecture." In Rahim, Ali (ed.). *Contemporary Techniques in Architecture*. London: Wiley-Academy, 9–13.
- ADRIAN SMITH AND GORDON GILL ARCHITECTURE. 2010. Unpublished presentation on Masdar. Columbia University, Graduate School of Architecture, Planning and Preservation, October 27.

Presenting Author Biographies



Anders Holden Deleuran

Anders Holden Deleuran is a Danish designer with an interest in interactive and digitally generated architectures and processes. His projects weave between the digital and physical, with physical constraints informing digital models, and digital models driving physical realisations. Anders is currently a Research Assistant at the Centre for Information Technology and Architecture (CITA), located in the Royal Academy of Fine Arts, Copenhagen.



Angelos Chronis

Angelos Chronis is currently working as a design systems analyst at Foster and Partners, UK. His academic background was initiated by a diploma in Architecture, which was awarded by the University of Patras, Greece. His dissertation, entitled "A Parametric Approach to the Bioclimatic Design of a Student Housing Building in Patras, Greece" was presented at the eCAADe 2010 conference and was selected among the 10 best papers for publication in a special issue of the Automation in Construction journal. He gained an MSc in Adaptive Architecture & Computation from the Bartlett School of Graduate Studies, UCL, with distinction, in the context of which he studied the integration of computational fluid dynamics and genetic algorithms as part of his dissertation.



Christian Schneider

Christian Schneider studied Advanced Computer Science at the University of Neuchatel in Switzerland and at the University of Aarhus in Denmark. His research and teaching activity at the ETH Zurich Department of Architecture involves projection mapping, responsive systems, data visualization, and computational design, where he currently teaches a course in parametric urban design.



Christopher Kroner

Christopher Kroner has taught digital design technology courses at leading design institutions across the country, most recently at Pratt Institute GAUD and Columbia University GSAPP. He is also a Senior Designer at the award winning firm Dean-Wolf Architects. His research includes modeling dynamically scripted systems, and using cross-program simulation for spatial and environmental performance analysis. Christopher holds a Bachelor of Science in Architecture from the University of Virginia, and a Master of Architecture from Columbia University, where he was the recipient of the Lucille Smyser Lowenfish Memorial Prize for design achievement.



Dimitris Papanikolaou

Dimitris Papanikolaou is a researcher and graduate student at the Smart Cities group of the MIT Media Lab where he guides the fleet management and sustainability directions of Mobility on Demand, one of the world's most revolutionary projects on the future of urban transportation. His research focuses on understanding complex behavior of self-organizing resource allocation systems using System Dynamics, Economics, and Game Theory. Dimitris holds a Master of Science degree in Design and Computation from MIT where he studied as a Fulbright Scholar, and a Professional Diploma in Architectural Engineering from the National Technical University of Athens in Greece.



Ebenezer Hailemariam

Ebenezer (Eben) Hailemariam is a research software developer with Autodesk Research. Eben received his Hon. B.Sc. in computer science from the University of Toronto in 2009.

Before graduating he spent over a year with AMD's graphics product group (formerly ATI) developing software to support digital video applications. Following graduation, he interned with Autodesk Research where he explored visualization and building information modeling (BIM) as a tool for sustainable building lifecycle management. In 2010, Eben joined the research team as a full time member where he continues to explore visualization and BIM as a tool to make buildings greener.



Eva Pouloupoulou

Eva Pouloupoulou is a licensed Greek architect, currently pursuing a Master of Science in Advanced Architectural Design at Columbia University with a fellowship from the State Scholarship Foundation.

She obtained her professional degree in Architecture from National Technical University of Athens in 2005, spending one year as an exchange student in the Ecole d' Architecture de Paris - la Villette. She also obtained a Certificate of Specialized Post-Graduate Studies in 3D Simulation and Visualization from the Ecole Nationale Supérieure des Arts Decoratifs in 2006. From 2006 to 2007 she worked as a visual artist in Paris for "La Cinematheque Francaise", while since 2007 she has established her own practice and worked on several individual and collaborative projects. Her participation in the project "City Of Love and Hate", co-authored by Adnan Ihsan, Amirali Merati and Foteinos Soulos, was realized during the Advanced Architectural Design Studio taught by David Benjamin.



Francesco Iorio

Francesco Iorio is a Senior Research Scientist on High Performance Computing at Autodesk Research. His research focuses on software scalability on multi-core and many-core systems, hybrid computing, accelerator systems and general design and exploitation of parallel algorithms on a variety of high performance platforms.

Prior to Autodesk Francesco was a Solution Architect for Next Generation Computing Systems at the IBM High Performance Computing Group in Dublin, with specific focus on accelerating financial, engineering and digital media workloads using hybrid high performance computing platforms. Francesco received his MSc in Information Technology - Software Engineering from The University of Liverpool, UK.



Jeffrey Niemasz

Jeffrey Niemasz is an Architecture student at Harvard University's Graduate School of Design. His research interests include solar zoning, light and health, and computational design. He is a developer of the DIVA-for-Rhino software, focusing on its practical use in design with the generative design software Grasshopper. Before starting his studies in Architecture, Jeffrey was a physician and instructor at the VA medical center in Minneapolis and the University of Minnesota. He completed a chief residency and internal medicine residency at the University of Minnesota and holds a medical degree from the University of Chicago.



John Michaloski

John Michaloski is a computer scientist and a member of the Intelligent Systems Division at the National Institute of Standards and Technology. He has over 25 years experience in robotics and automation technology for manufacturing systems. His current work interests include manufacturing communication standards, manufacturing data mining applications, and integration of manufacturing and Discrete Event Simulation technologies. John is currently a member of ISA, MTConnect Institute, OMAC Machine Tool Working Group and NIST liaison to the OPC Foundation. John Michaloski earned his MS in computer science from the Georgia Institute of Technology, and his BS in mathematics from the University of Maryland.



Kerri Henderson

Kerri Henderson is currently completing her Masters of Architecture at the Graduate School of Architecture, Planning and Preservation, Columbia University. Kerri is a Teaching Assistant at Columbia where she teaches Architectural Drawing and Representation II and Introduction to Architecture (design studio) She has been a guest critic at Parsons and Columbia. Her recent projects include: Proof Tower (a mixed-use tower in Masdar using multi-objective optimization), a lunar rover (collaboration with NASA) and Lean Urbanism (a master plan integrating urban development and sugar cane-based Ethanol production in Santo Domingo, DR) Kerri graduated from the University of Waterloo in Canada, with a Honours Bachelor of Architectural Studies. She has worked in New York, Toronto and Paris.



Laëtitia Arantes

Laëtitia Arantes is a French PhD candidate at the CRAterre laboratory at the School of Architecture in Grenoble and the CSTB (French Scientific Centre for Building Science). She holds a graduate degree in civil Engineering from the ENTPE (Ecole Nationale des Travaux Publics de l'Etat in Lyon, France) and a graduate degree in Architecture from the ENSAG (Ecole Nationale Supérieure d'Architecture de Grenoble, France). Her current research lies in the interface between Architecture and Engineering: it deals with the design of sustainable high-building and cities.



Lilli Smith

Lilli Smith, AIA, LEED® AP is a Senior Product Designer at Autodesk. After practicing architecture for several years, Lilli joined Revit Technology in its infancy. She was involved in creating much of Revit's core functionality including the conceptual design tools before turning her focus to sustainable design. She was the lead product designer on the Conceptual Energy Analysis feature in Revit Architecture and is currently working on Project Vasari, a spin-off of Revit which is focused on conceptual design and analysis.



Naai-Jung Shih

Naai-Jung received his Doctor in Architecture in University of Michigan with a focus on Computer Aided Design in Architecture. He has been teaching in the Department of Architecture, National Taiwan University of Science and Technology, for years.

Recent projects have been related to urban scans in Taipei and the historical preservation of temples and township in northern Taiwan, sponsored by National Science Council and local government departments.



Peter Andreas Sattrup

Peter Andreas Sattrup is a Danish architect and educator. With a background in architectural practice, Peter Andreas Sattrup is currently (2011) pursuing a PhD in Architecture on the subject of environmental simulation modelling. Having designed and worked on cultural, commercial and housing projects in Denmark and the UK for the past decade and a half, he applies his practical knowledge of the profession in his research. As an architect he takes the position that a holistic view on building performance is of greater value to the built environment than the optimization of technical sub-systems. Combining qualitative and quantitative analysis methods are essential to achieve that.



Robert Aish

Robert Aish studied Industrial Design at the Royal College of Art in London and has a Ph.D. in Human Computer Interaction from the University of Essex. He has developed engineering software with Arup, architectural software with Rucaps, naval architecture software with Intergraph and the GenerativeComponents parametric design software with Bentley. In 2005 the UK, 'Building Design' Magazine named Robert Aish as one of the top ten innovators in British Architecture. In 2006 he received the 'Association for Computer-Aided Design in Architecture' (ACADIA) Society Award. He is a co-founder of the SmartGeometry Group and visiting professor of Design Computation at the School of Architecture at the University of Bath, in the UK. His research interests include: the design of end-user programming languages and visual programming interfaces and the role of these systems in advanced architectural design. His role at Autodesk is to converge this research with the main stream of design and engineering software.



Shajay Bhooshan

Shajay Bhooshan currently, heads the research activities of the Computation and Design (colde) group at Zaha Hadid Architects, London and works as a course tutor at the AA DRL Masters program. He completed his Masters Degree AA School of Architecture, London in 2006.

He has taught and presented work at various events and institutions including Beyond Media Florence 2009, Siggraph2008, Frozen festival Amsterdam 2008, Universities of Applied Arts in Vienna, and Experimental Architecture in Innsbruck, and Institute for Advanced Architecture of Catalonia. Previously whilst at Populous, he played an active role in the completion of the O2 Arena within the Millennium Dome, Greenwich London, as also in the development of various design proposals for the Oval cricket stadium, London.

His team's Master's thesis project, swarm city, was awarded the Royal Society of Arts (London) Head Trust award in 2006. He has been part of various collaborative work that has been published in many publications including RSA Design Directions 2005/06 competition winners publication, International Conference on Computer Graphics and Interactive Techniques (ACM SIGGRAPH 2008 art gallery), and Journal of Indian Institute of Architects.



Souheil Soubra

Dr. Souheil Soubra is Head of the MODEVE Division at CSTB (France) which is the national research center for the built environment. After a Civil Engineering diploma and postgraduate studies at the "Ecole Nationale des Ponts et Chaussées" (ENPC - Paris - F), he undertook postgraduate research and received his Ph.D. from ENPC in 1992.

His main fields of interest include Simulation Environments, Building Information Modeling, Geographical Information management, Virtual and Augmented Reality. He is currently managing CSTB's strategic project entitled "Numerical Simulation and tools" in the frame of which an innovative immersive space dedicated to Virtual Environments for the construction sector has been put in place. He participated or managed several EC funded projects (Divercity, V-MAN, ViSICADE, Create, Intelcity, Intelcities ...). Souheil Soubra is a member of various expert committees in "Construction IT" and has co-authored several publications and books in this field.



Thomas Grasl

Thomas Grasl is partner at SWAP Architects, Vienna, Austria, where he also heads the R&D department. His research focuses on computational design, design support tools, and the application of grammars in Architecture. Recent collaborative projects include a funded research project on the generative description of the architectural typology of Federal Courthouses; the project sieve on the automated computation, enumeration, and visualization of nested symmetry substructures; and GRAPPA an implementation of the Palladian Grammar in Autodesk Revit. Thomas Grasl holds a M.Sc. in Architecture, TU Vienna, 2002, a Fulbright scholarship for advanced studies in architecture, Georgia Institute of Technology, USA, 2006, and currently he works towards his doctorate degree at TU Vienna.



Tyler Garaas

Tyler, now at Mitsubishi Electric Research Laboratories (MERL), received his Ph.D. from the Department of Computer Science at the University of Massachusetts Boston where he was an active member of the visual attention laboratory. Tyler earned his bachelor's degree in computer science at Montana State University. His work at MERL includes computation and visualization research applied to urban security systems, font rendering, computer vision, and shape representation among others.



Yoshihiro Kobayashi

Yoshihiro Kobayashi is a Faculty Research Associate and teaches Game Development courses at the School of Computing, Informatics, and Decision Systems Engineering at Arizona State University, USA. He received a PhD degree from UCLA in 2001 majoring in Design Computation in Architecture and a minor in Computer Science. His research focuses on developing computational design tools to generate buildings and cityscapes using the procedural modeling and image-driven approaches. He is also working for a civil engineering software developing company, Forum8 Co. Ltd, as a supervisor on developing and selling VR and BIM software packages.

Organizers

Symposium Chair

Ramtin Attar

Autodesk Research

Program Committee

Robert Aish

Autodesk

Marilyne Andersen

Interdisciplinary Laboratory of Performance-Integrated Design, École Polytechnique Fédérale de Lausanne

Sean Hanna

VR Center for the Built Environment, UCL Bartlett School of Graduate Studies

Michael Jemtrud

School of Architecture, McGill University

Ian Keough

Senior Technical Designer, Buro Happold Consulting Engineers

Azam Khan

Autodesk Research

Branko Kolarevic

Faculty of Environmental Design, University of Calgary

Lira Nikolovska

Autodesk AEC

Rivka Oxman

Vice Dean, Faculty of Architecture and Town Planning, Technion ITT

Christoph Reinhart

Graduate School of Design, Harvard University

Gabriel Wainer

ARS Lab, Carleton University

Bernard P. Zeigler

Arizona Center for Integrative Modeling & Simulation

Sponsors

Sponsored by

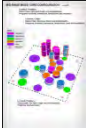
Autodesk®

In co-operation with



Cover Image Credits

171 **Multi-Objective Optimization in Urban Design**



MICHELE BRUNO, KERRI HENDERSON and HONG MIN KIM
Columbia University

115 **A simple method to consider energy balance in the architectural design of residential buildings**



LAËTITIA ARANTES, OLIVIER BAVEREL, PASCAL ROLLET and DANIEL QUENARD
Centre Scientifique et Technique du Bâtiment (CSTB)

87 **Use of Sub-division Surfaces in Architectural Form-Finding and Procedural Modelling**



SHAJAY BHOOSHAN and MOSTAFA EL SAYED
Zaha Hadid Architects

163 **Components for Parametric Urban Design in Grasshopper. From Street Network to Building Geometry**



CHRISTIAN SCHNEIDER, ANASTASIA KOLTSOVA and GERHARD SCHMITT
ETH Zurich

79 **Generative Fluid Dynamics: Integration of Fast Fluid Dynamics and Genetic Algorithms for wind loading optimization of a free form surface**



ANGELOS CHRONIS, ALASDAIR TURNER and MARTHA TSIGKARI
University College London (UCL)

05 **Designing with Deformation - Sketching material and aggregate behaviour of actively deforming structures**



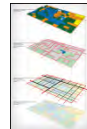
ANDERS HOLDEN DELEURAN, MARTIN TAMKE and METTE RAMSGARD THOMSEN
Royal Danish Academy of Fine Arts, School of Architecture

143 **3D Scans of As-Built Street Scenes for Virtual Environments**



NAAI-JUNG SHIH, CHIA-YU LEE and TZU-YING CHAN
National Taiwan University of Science and Technology

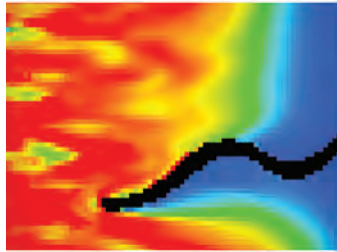
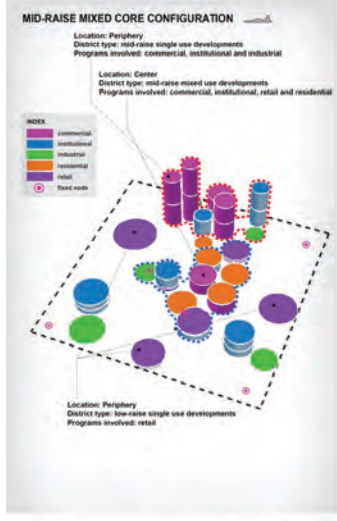
155 **The City of Love and Hate**



ADNAN IHSAN, AMIRALI MERATI, EVANGELIA POULOPOULOU and FOTEINOS SOULOS
Columbia University

Index of Authors

Aish, Robert 61	Keough, Ian 135	Shao, Guodong 13
Arantes, Laëtitia 115	Khan, Azam 23, 135	Shih, Naai-Jung 143
Arinez, Jorge 13	Kim, Hong Min 171	Smith, Lillian 53
Attar, Ramtin 23	Kobayashi, Yoshihiro 95	Snowdon, Jane L. 69
Barry, Liz 149	Koltsova, Anastasia 163	Soubra, Souheil 135
Baverel, Olivier 115	Kroner, Chris 149	Soulos, Foteinos 155
Bernhardt, Kyle 53	Lee, Chia-Yu 143	Strømmand-Andersen, Jakob 123
Bhooshan, Shajay 87	Leong, Swee 13	Szivos, Mike 149
Bruno, Michele 171	Lyons, Kevin 13	Tamke, Martin 5
Chan, Tzu-Ying 143	Marsh, Andrew 61	Tessier, Alex 135
Chronis, Angelos 79	Merati, Amirali 155	Thomsen, Mette Ramsgard 5
Deleuran, Anders Holden 5	Michaloski, John 13	Tsigkari, Martha 79
Duong, Phu 149	Niemasz, Jeffrey 105	Turner, Alasdair 79
Economou, Athanassios 45	Papanikolaou, Dimitris 35	Wonka, Peter 95
El Sayed, Mostafa 87	Picinbono, Guillaume 135	
Garaas, Tyler 31	Poulopoulou, Evangelia 155	
Goldstein, Rhys 23	Quenard, Daniel 115	
Graf, Holger 135	Reinhart, Christoph 105	
Grasl, Thomas 45	Riddick, Frank 13	
Hailemariam, Ebenezer 23	Rollet, Pascal 115	
Henderson, Kerri 171	Sargent, Jon 105	
Ihsan, Adnan 155	Sattrup, Peter Andreas 123	
Iorio, Francesco 69	Schmitt, Gerhard 163	
Jezyk, Matthew 53	Schneider, Christian 163	



Symposium on Simulation for Architecture and Urban Design 2011

