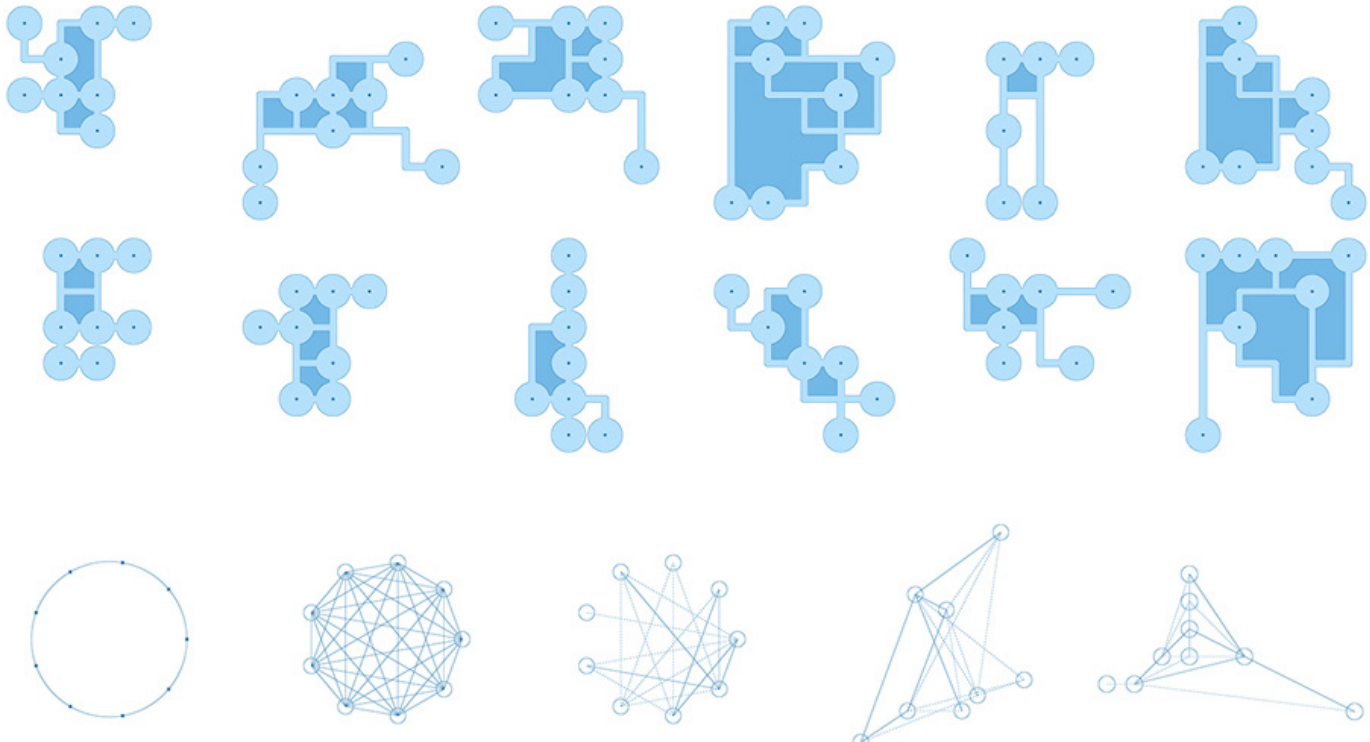# SELECTIVE INTERFERENCE
EMERGENT COMPLEXITY INFORMED BY PROGRAMMATIC, SOCIAL AND PERFORMATIVE CRITERIA

Christopher Welch
Jules Moloney
Tane Moleta
Victoria University of Wellington

1 Genetic Space Planning - diagrammatic space planning tests undertaken in Grasshopper

ABSTRACT

This research aims to demonstrate if a holistic approach to generative architectural design is feasible using algorithms and techniques now common in architecture studios, within the context of a twelve months design thesis undertaken in a professional architecture programme. The final stage of research executes on the strategy of bringing tributaries of information together at logical junctions to create a start-to-finish generative tool in a single Grasshopper definition. By formalizing and simplifying the intersections between discrete processes, a complex "open box" design structure is developed that produces responsive, novel conceptual designs in a marriage of designer input and computer processing.

INTRODUCTION

Parametric design tools and visual programming languages are fast becoming an important part of the architects' design process. Researchers note that the barrier to entry into the medium is lowering while the power of the tools available is increasing, allowing for greater agency in developing computer generated architectural form (Burry 2011; Davis & Peters 2013). However, independent of user input, complex interacting systems, guided by simple rules, can be used to produce novel, emergent solutions (Frazer 1995; Oosterhuis 2002; Coates 2010; Perony 2014). While there is precedent for singular applications such as the work of Frazer and Coates cited above, this is seldom developed in the context of realistic design contexts. Moreover, there is minimal precedent of a holistic approach that brings together and simultaneously considers concepts such as site planning, internal space planning and daylighting while allowing for the selective interference of the designer. This research aims to demonstrate if selective interference and a holistic approach to generative architectural design are feasible using algorithms and techniques now common in architecture studios, within the context of a twelve months design thesis undertaken in a professional architecture programme.

The project began with a series of technical studies exploring space planning, site delineation and circulation. These individual processes were then amalgamated to develop a single complex generative algorithm. In order to provide a direction and a degree of creative friction within the research process, the projects were framed around the development of a mid-size, urban sited secondary school. The viability of the algorithm was then tested through the generation of test buildings across a variety of sites.

The final algorithm is structured in such a way that the architecture emerges in a controllable, predictable way that can still surprise and inform, as well as consistently produces viable, interesting building layouts. As such, the research is situated between the intent of the designer *(top-down)* and the capacity of the computer to evolve solutions *(bottom-up)*. This has produced a wide variety of working concepts within a flexible and extendable system which allows for a high level of agency as a designer.

REVIEW OF RELATED WORKS

There is a temptation to use parametric design techniques as drivers for complicated form, using the processing of information to create geometry that would not be feasible by non-parametric means. However, if the information that enters into the parametric model is linearly processed, is the resultant structure informed or just transformed? Michael Meredith challenges the complexity that

emerges out of rote transformation as a "subversion of semiotic legibility" eroding meaning in the quest for a specific aesthetic condition (Meredith 2008). Transformation, multiplication and proliferation are straightforward tasks using these tools, but does that mean that the digital needs to be obsessed with processed, overly elaborate form? By shifting the focus away from transformation and towards informed interaction, the designer can retain agency while still exploring non-standard spatial arrangements.

In contrast to much contemporary parametric design examples where design intent is encoded in the parameters, *"bottom-up"* generative processes provide a point of difference. Rather than focusing on complicated chains of transformations, the complex interaction of simple rules produces emergent phenomenon (Johnson 2009). By building a complex multifaceted system that processes many interacting inputs, relationships and connections between stimuli can create unforeseen forms and patterns. Examples range from buildings generated from data-driven environmental models to algorithms designed to explore social and organizational topics.

In Roudavski's discussion of morphogenetic case studies, he outlines that support for more flexible, differentiated cells and structures within a morphogenetic system creates further opportunity for interaction, differentiation and complexity within *"bottom-up"* cellular structures. He suggests the integration of tools for evaluating and adapting to local changes in environment, allowing the parametric model to become more responsive (Roudavski 2009). By bringing a number of interacting subsystems together, morphogenetic cellular architecture can further evolve towards producing robust, novel and emergent form (Schumacher 2009). Cynthia Ottchen considers the focus on purely material and physical properties in digital design process to be too narrowly focused, ignoring a wealth of other notions more suited to the field of architecture—social concerns, systems and programmatic organization. She writes that there is a tendency to use computers to explore quantitative data, and not use them to explore qualitative aspects of design, thus reinforcing a divide between parametric design techniques and traditional space planning methods, instead of being one and the same (Ottchen 2009).

This review of "bottom-up" approaches to design computing suggests some promise and is in contrast to what is broadly termed a "top-down" approach, where the computer is being used as a processor. Often this involves a procedural prescribed translation throughout a system, where the outcome can be predicted and the possibility of generating a novel solution is unlikely. Systems that are complex, rather than complicated, forgo processing and transformation, focusing instead on using simple rules that interact to produce unpredictable solutions. The higher the number of individual
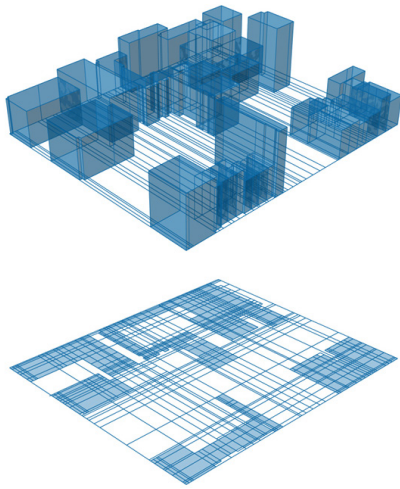
2 Site Delineation – dynamic site partitioning into discrete cells using Grasshopper
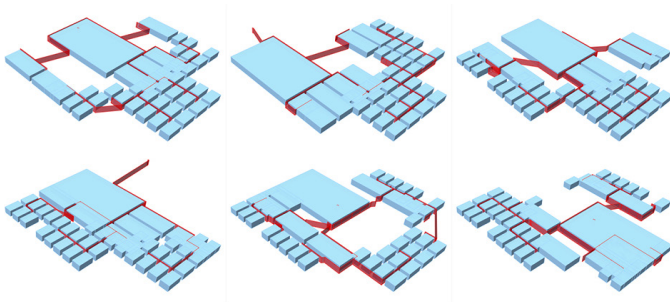
3 Dynamic Circulation -building circulation generated in Grasshopper using A*
   search algorithm

interacting elements, the further the computer outstrips the designer at processing these relationships. The more complex the system, the higher the chance of interesting, emergent architectural form.

The position explored in this paper is to attempt to produce a balance between a bottom-up and top-down approach to the use of the computer at the early design stages, with an emphasis on the organizational aspects of architectural design: site planning, movement patterns, space planning and environmental orientation. The focus is on combining existing generative techniques to enable multiple criteria to be generated simultaneously. However, the intent is not to provide a "hands free" system. As will be evident in the final iteration of the design case study, the level of control and feedback is in effect a combination of top-down and bottom-up

approaches, marrying the intent of the designer with the emergent capacity of complex systems. This interaction between designers and algorithms is captured by the phrase "selective interference."

EARLY WORK

"Problem decomposition" is an important consideration when transforming a complex computational process into a manage-able series of tasks (Burry, Scripting Cultures 2011). As such, the preliminary design phase of research has been broken down into a series of discrete experiments considering the organization and generation of space, and then producing a series of integrated tests. These early experiments were divided into three main areas of enquiry: space planning, site delineation and circulation.

Procedural space planning was explored through the use of the *Grasshopper* extension 'Galapagos,' a commonly used and freely available genetic solver (Rutten 2013). This was utilized to optimize relationships between points in space, using a matrix of rela-tionships derived from techniques explored by previous space planning researchers (American Institute of Architects 1981; Merrell, Schkufza, & Koltun 2010; Schneider & Koenig 2012) (Figure 1).

Delineating the site into discrete volumes was an important step to-wards dynamically creating a parsable map of an arbitrary urban site. The solution eventually settled upon, relying on Delaunay triangulation derived from the existing urban context as described by geodata (Figure 2). With the site delineated into a connected network of discrete vol-umes, site analysis of solar angles, building adjacencies and foot traffic can be translated into a consistent language of weighted cells.

The A* shortest walk algorithm, a common AI path-finding tech-nique was used to identify possible movement flows of people through the site (Jurgen, Wagner and Zweig 2009). Direct paths through the site, bottlenecks, and adjacencies to heavy foot traffic could all be determined from this technique. For building circu-lation, a recursive looping function using the A* algorithm was developed to create corridors incrementally, based on a predeter-mined spatial hierarchy (Figure 3).

When considered individually, many of the algorithms and tech-niques underpinning these experiments are well-documented and widely used. The focus of this research was not on producing a novel technique, but on developing a holistic approach that con-nected these existing approaches together into a cohesive design framework. It was through the interaction between these simple techniques that complex and emergent systems were developed.

DEVELOPED DESIGN WORK

The final stage of research was the strategy of bringing tributaries of information together at logical junctions to create a start-to-finish generative tool in a single *Grasshopper* definition. By formalizing and simplifying the intersections between the discrete processes, the algorithm becomes clearer and more responsive to change. This process, which we refer to as an "open box" structure, produced a wide variety of working concepts and provided a high level of control as a designer. This modular approach has precedent in approaches such as *Object Orientated Programming*, which in turn was informed by Christopher Alexander's Pattern Language (1977). In this context, 'Open Box' is a designer friendly term that captures an open-ended modular approach to design computing.

The first step in creating structure to underpin the algorithm was the decision to limit the typology of the design to a perimeter block. This decision was made both for programmatic and technical reasons, both as way of spatializing the space planning data and as a response to creating outdoor space in a high density urban environment.

In this system, programmatic requirements of the building are explicitly organized into a single list, a bar graph of volumes that shuffles in one dimension based on data from a spread sheet (Figure 4). The benefits are twofold, both lowering the probability space for the organizational algorithms and enabling a certain amount of organization at the start of the project without having to allow for all possibilities.

The organization of the perimeter block is laid out by the designer, as informed by daylighting and shortest walk algorithms representing the existing site context and movement patterns through the site. Different block organizations, be they fragmented or contiguous, can be quickly and intuitively explored by the designer.

The polyline is then propagated upwards, creating a large number of nodes that will define the building program (Figure 5). The area that makes up this container is then trimmed based on the spatial relationship between the nodes and existing building adjacencies, drawing the building heights to tend towards that of the existing context, and promoting density in shaded areas to maximize sunlight in the courtyard.

The programme is determined by spread sheet, imported and mapped to fill up the possible building space (Figure 6). A second evolutionary solver is then run, this time calculating adjacencies and layouts by organizing and shuffling the programme list to optimize layout based on required adjacencies described in the original matrix.
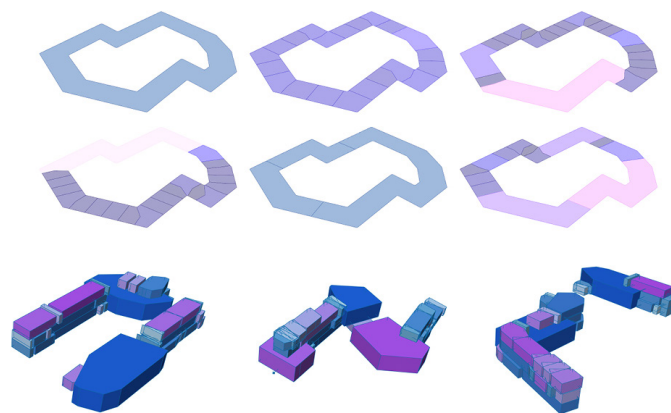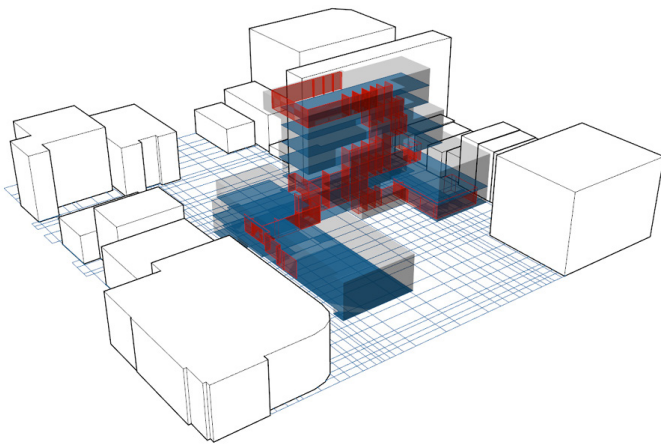

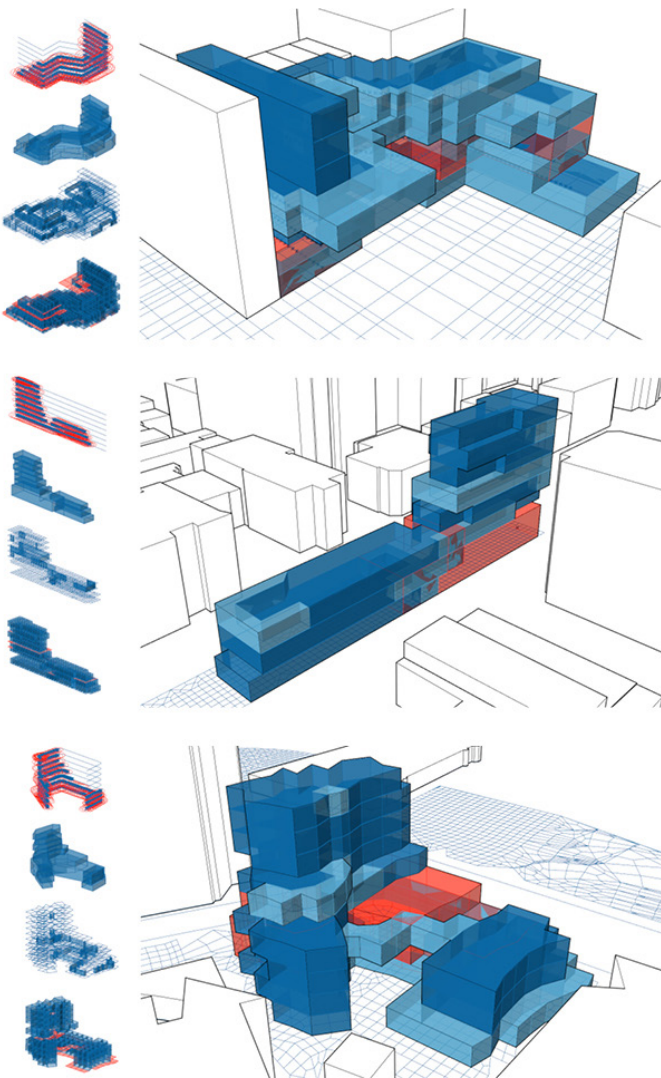4   Data Translation – information from a spreadsheet is converted into volumetric space
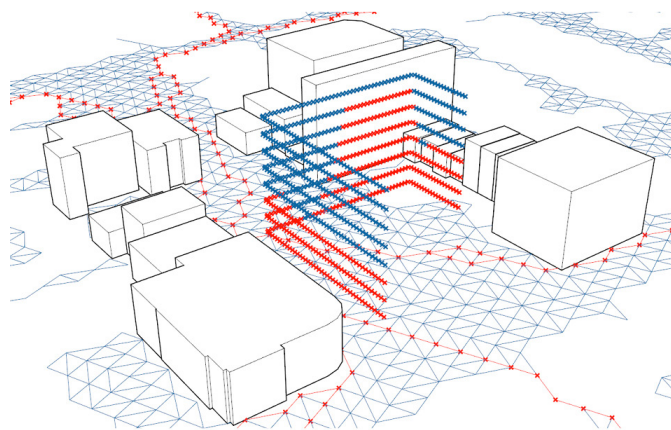
5   Building Generation Part 1 - perimeter block derived from surrounding context
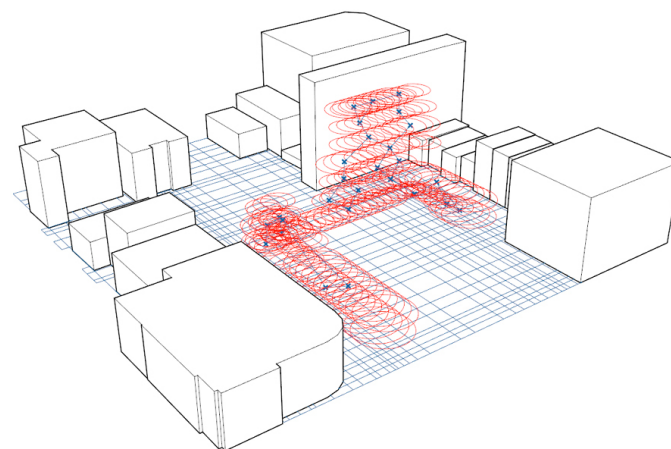
6   Building Generation Part 2 - genetic solver determines programmatic adjacencies

7　Building Generation Part 3 - abstract massing translated into contextual grid

8　Building Iterations - building iterations across three test sites

All of this information is then translated onto the site through a context-responsive rectangular mesh that determines the underlying structural grid of the building. These responsive grid cells allow the circulation and building envelope to respond intelligently to the conditions of the surrounding site, identifying discontinuities and existing volumetric relationships. After the programme has been spatialized, the looping shortest walk algorithms are run, creating corridors and connections that feed throughout the building based on a hierarchy of importance (Figure 7).

DISCUSSION

The objective of this research was to demonstrate both the feasibility of selective interference by the designer and a holistic approach to integrating algorithms and techniques. The final iteration of the algorithm successfully marries programme, typology and contextual information in order to generate different organizations of programme and form across different sites (Figure 8). More importantly, this experimentation has culminated in an overarching philosophy for organizing the structure of the parametric model: independent structures that interact at controlled intersections, an amalgamation of individual algorithms in a process dubbed "open box" design.

The approach of breaking the process down into discrete components and then carefully managing the interactions was initially just a way of controlling the scope of the project. In the end, it became the most important part of how the algorithm is structured, turning it from a tangle of interacting elements into a number of "macro-level" *Grasshopper* components with a clear structure (Figure 9).

An unforeseen and important consequence of this new process is that the architect has a higher degree of control over the generation process than was envisioned initially. While as a whole, the parametric model is highly complex, the outputs are clear and simple, each dictating a specific part in the process. As long as format for these outputs is maintained, each can be edited individually, allowing selective interference by the designer, and the system responds to these changes in ways understandable but unpredictable.

Change can be made surgically, a single modification to a single system, and the resultant changes are focused, but far reaching. Modifications to the programme will create differences in circulation and massing, but not to the broad outline of the building. The algorithm that controls circulation, for example, can be endlessly modified, shifting the organization of corridors independently from the programme (Figure 10). Once a circulation system has been determined, the structural grid could be shifted from 2400mm centres to 4000mm centres to test how that might change other conditions
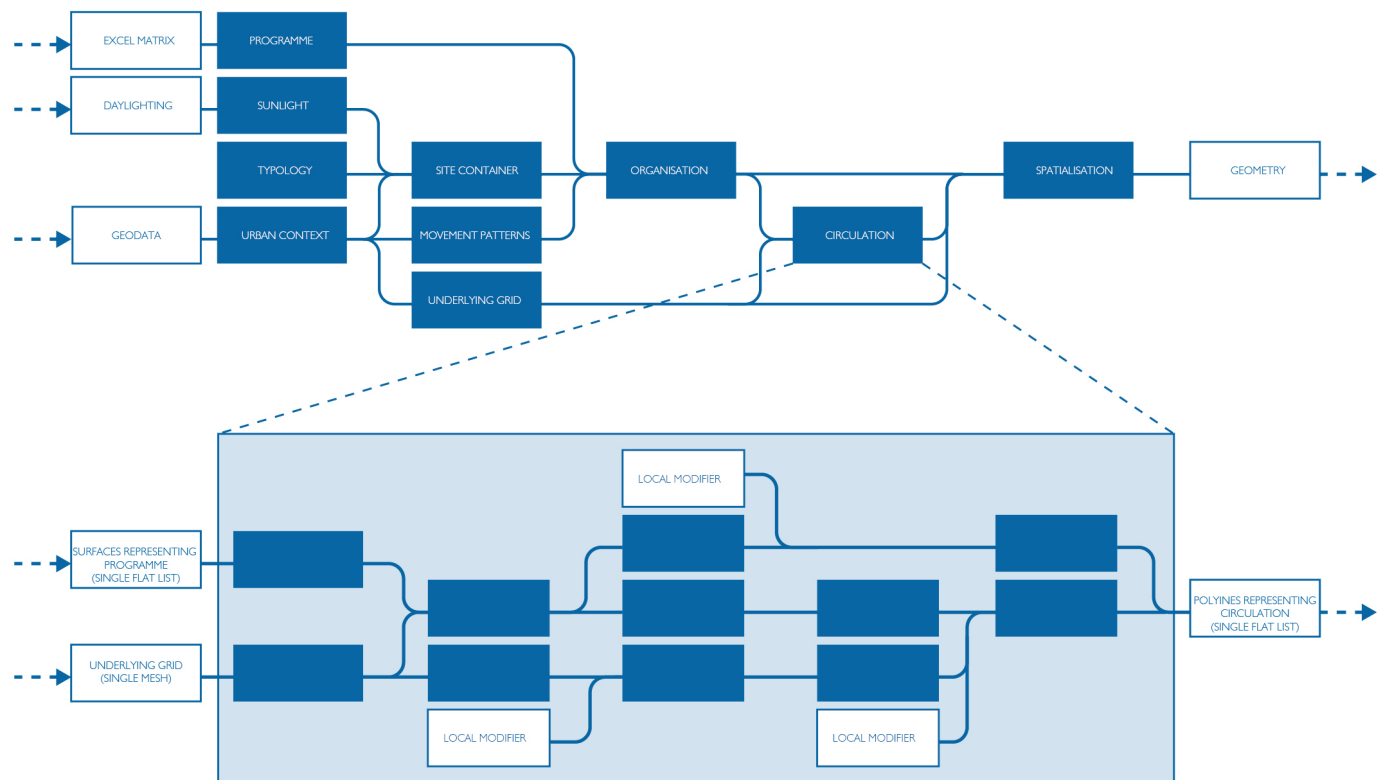
9 The "Open Box" - the meta-structure allows large scale internal changes to be recognised globally by the system
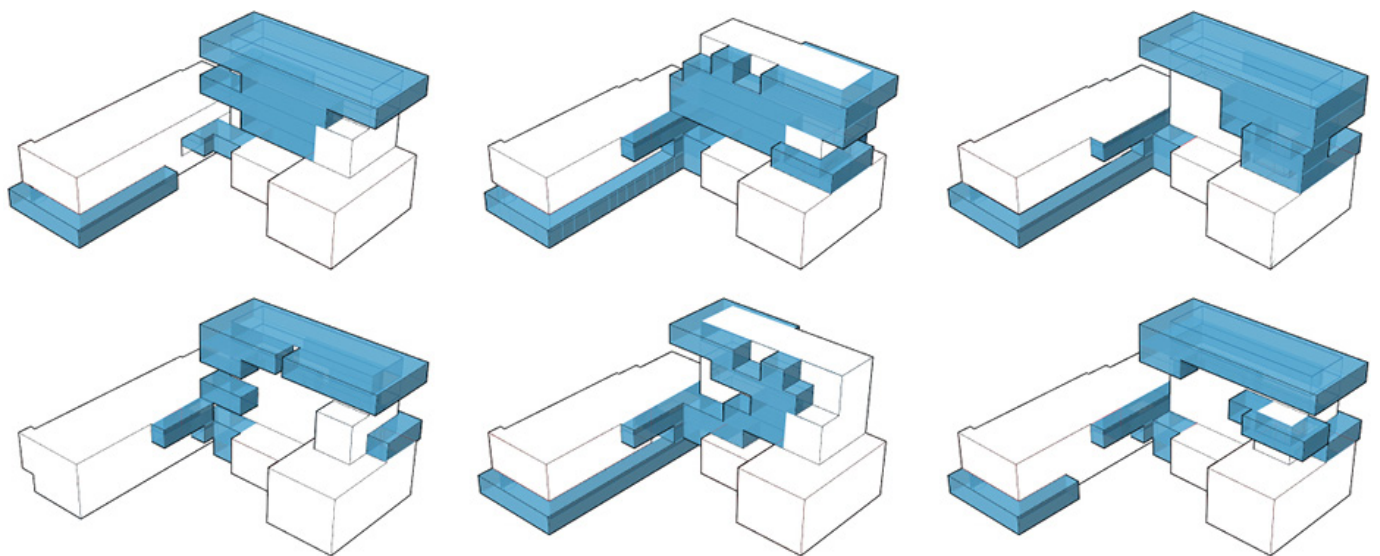
10 Localised Iteration -global changes to circulation, as affected by local inputs
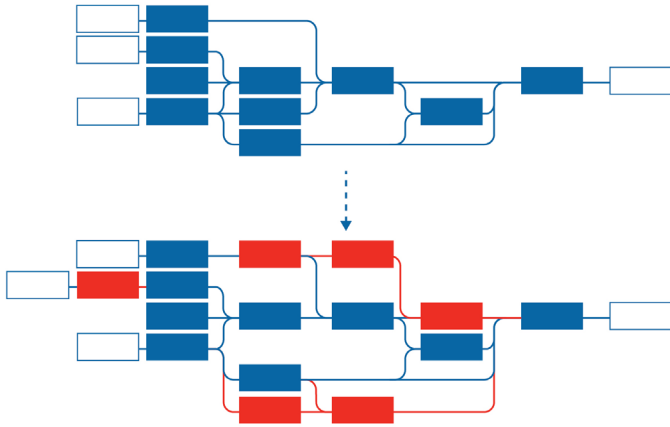
11 An Evolving Process – the simplification of inputs/outputs allows the "Open Box" meta-structure to evolve over a series of projects

of the building. Each individual algorithm can be tweaked independently, endlessly testing iterations without any major revisions to the rest of the definition and creating an environment of serendipitous discovery though rapid iteration and experimentation. The level of control and feedback inherent in this approach represents a cognitive shift away from the concept of a computer-driven parametric process and back towards a designer-driven model. Once a concept has been generated, the design process becomes that of changing individual parameters and systems, and watching the rest of the model react to decisions generatively. The system of encoded relationships allows for rapid iteration, without removing the agency of the designer.

GEOMETRY BOTTLENECK

The focus of this research was on the early part of the design process, creating a script that develops a spatially logical, conceptual building. The final outcome of this is a framework that outputs a set of polylines, points and surfaces representing floor plates, walls and structure organized in multiple stories. The *Grasshopper* information can be moved directly into *Revit* or *AutoCAD*, shifting the focus of the algorithm towards that of a three-dimensional sketch.

Alternatively, with appropriate organization and clever use of resources, the *Grasshopper* algorithm can be modified and augmented to focus on specific design elements. The algorithm can be extended and iterated on to explore specific aesthetic conditions that emerge during the generation process. The strength of the model is that it allows for endless embellishment and extension, eschewing the user-orientated, one-size-fits-all approach to parametric design for a reactive, open-ended design environment.

PARTNERSHIP BETWEEN DESIGNER AND COMPUTER

The final combination of algorithms produces an architecture that does not surprise with sweeping gestures, but with quiet, serendipitous moments–the interaction between circulation and a hall; the way a structural grid interferes with a facade; the shelter that forms under a cantilevered office. It is a partnership of rapid, informed, surgical iteration, allowing the computer to calculate relationships and then augmenting these interactions with your own gestures. The design process becomes a 'discussion' of iteration and tweaking, with your computer partner in which complex exploratory concepts can be iterated on in seconds. Through disagreement and evolution, a range of sketch design ideas emerge.

THE IMPORTANCE OF THE "OPEN BOX"

An "open box" workflow allows these user-oriented tools to be integrated into a complex architectural workflow without completely interfering with the rest of the structure. The focus on informed complexity, rather than overcomplicated simplicity allows the rest of the building to respond to these new tools as they become available, and gives the algorithm the capacity to evolve from project to project (Figure 11).

By building a holistic parametric framework, there is capacity to react to changes and advances in the medium as well as explore and expand different elements of the generative model from project to project. This creates an open, organic, holistic framework that evolves over time, incorporating new ideas and systems as they emerge into the public domain.

FUTURE WORK

The explicit and simplified parametric framework is designed around expandability. As such there is room for additional projects pertaining to specific aspects of the generation engine; expanded versions of the components that govern space planning and circulation could give further depth and nuance to the generator as a whole. At the time of writing, the beta versions of the *Grasshopper* plugins *"Space Syntax"* and *"Spiderweb"* are beginning to circulate, tools which could expand upon many of the space planning ideas explored during the preliminary design phase of this research. This work could represent a solid foundation for future exploration in the area of computer-aided space planning.

CONCLUSION

Working effectively within increasing complexity relies to some degree on a level of simplicity (Perony 2014). By keeping the interactions limited to very specific pathways, the interaction of

programme, typology and contextual information are used to generate a wide variety of conceptual options for urban high schools across any number of sites. By bringing the interaction between datasets to the forefront, not just the processing of them, and by focusing on fostering unpredictable behaviors through complex relationships, an informed and functional architecture can emerge.

This process provides constraints in such a way that the architecture evolves in a natural, predictable way that can still surprise and inform, as well as consistently producing viable, interesting iterations of buildings. The "Open Box" structure and the concept of "selective interference", have produced a wide variety of working concepts and provided a high level of control as a designer.

It is an unusual thing to look upon a design, for the first time, that clearly embodies your own design sense, but for which you never had a hand in the physical modeling. But allowing the computer to become an extension of your design process—imbuing design decisions into the structures that generate your geometry simply provides a different kind of control, a different kind of agency as a designer. By encoding your design decisions at such a low level, the designer is free to explore—modifying conditions at a local or global level and watching how those new ideas transform the building. By relinquishing a degree of control to the computer, the role of the designer shifts towards that of intuitive decision making and the design process becomes one of discovery and collaboration.

REFERENCES

Alexander, Christopher. 1977. A Pattern Language: Towns, Buildings, Construction. Berkeley: Oxford University Press.

American Institute of Architects. 1981. The Architect's Guide to Facility Programming. New York: Architectural Record Books.

Burry, Mark. 2011. Scripting Cultures. Chichester: John Wiley & Sons.

Coates, Paul. 2010. Programming Architecture. Abingdon: Routledge.

Davis, Daniel, and Brady Peters. 2013. "Design Ecosystems: Customising the Architectural Design Environment with Software Plug-ins." Architectural Design 83:124-131.

Frazer, John. 1995. An Evolutionary Architecture. London: Architectural Association Publishing.

Johnson, Neil. 2009. Simply complexity: A clear guide to complexity theory. Oxford: Oneworld Publications.

Jurgen, Lerner, Dorothea Wagner, and Katharina Zweig. 2009. Algorithmics of Large and Complex Networks. Berlin: Springer-Verlag.

Meredith, Micheal. 2008. "Never Enough (transform, repeat ad nausea)." In From Control To Design: Parametric/Algorithmic Architecture, edited by Tomoko Sakamoto and Albert Ferré, 6-10. Actar Birkhauser Distribution.

Merrell, Paul, Eric Schkufza, and Vladlen Koltun. 2010. Computer-Generated Residential Building Layouts. Stanford: Stanford University.

Oosterhuis, Kas. 2002. Architecture Goes Wild. Rotterdam: 010 Publishers.

Ottchen, Cynthia. 2009. "The Future of Information Modeling and the End of Theory." Architectural Design 79:23-27.

Perony, Nicolas. "Puppies! Now that I've got your attention, complexity theory." Filmed October 2013. TED Video, 13:45. Posted January 30, 2014. http://www.ted.com/talks/nicolas_perony_puppies_now_that_i_ve_got_your_attention_complexity_theory.html

Roudavski, Stanislav. 2009. "Towards Morphogenesis in Architecture." International Journal of Architectural Computing 7:345-374.

Rutten, David. 2013. "Galapagos: On the Logic and Limitations of Generic Solvers." Architectural Design 83:132–35.

Schneider, Sven, and Reinhard Koenig. 2012. "Hierarchical structuring of layout problems in an interactice evolutionary layout system." Artificial Intelligence for Engineering Design, Analysis and Manufacturing 26:129-142.

Schumacher, Patrik. 2009. "Parametricism - A New Global Style for Architecture and Urban Design." Architectural Design 79:14-23.

IMAGE CREDITS

All image credits to Christopher Welch (2014).

CHRISTOPHER WELCH is a recent graduate from Victoria University of Wellington's Master of Architecture program. His thesis focused on using computational design techniques to aid space planning for architectural design. Chris's digital work is output-oriented, treating specific parametric strategies as a means to an end rather than an all-encompassing philosophy, bringing varied and disparate techniques together with a focus on producing complex but feasible architecture.

JULES MOLONEY's research, creative practice and teaching span the fields of architecture, kinetic art and virtual environments. This hybrid practice is predicated on an equal mix of theory and intuitive engagement with digital technology. His book Designing Kinetics for Architectural facades: State Change (Routledge, 2011) articulates a framework for the design of kinetics from first principles. He is currently Professor of Interdisciplinary Digital Design and Head of School at Victoria University of Wellington.

TANE MOLETA engages research led teaching at Victoria University of Wellington's School of Architecture. He is currently involved with research projects in parametric design for timber structures; mass customization for medium density housing; mixed, virtual and blended reality as design tools and game engines in the design studio. He has recently published papers in CAADRIA (Computer-Aided Architectural Design Research in Asia), eCAADe (Education and research in Computer Aided Architectural Design in Europe), NIME (New Interfaces for Musical Expression) and AANZ (Art Association Australia and New Zealand).