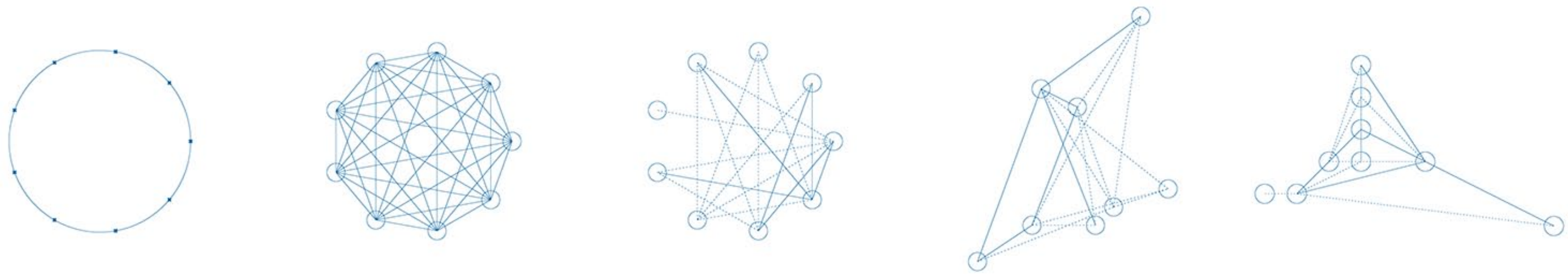
selective interference

EMERGENT COMPLEXITY INFORMED BY PROGRAMMATIC, SOCIAL AND PERFORMATIVE CRITERIA



A THESIS
SUBMITTED TO THE VICTORIA UNIVERSITY OF
WELLINGTON IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF MASTER OF
ARCHITECTURE

VICTORIA UNIVERSITY OF WELLINGTON
2014

CHRISTOPHER DAVID WELCH

Acknowledgements

The work presented in this thesis is the product of a long cycle of discussion, iteration, dead ends and epiphanies, and would not have been the same without the support and input of those around me.

Firstly I would like to thank my supervisors, Professor Jules Moloney and Tane Moleta for insight and direction, and for providing me with the opportunity to discover and deeply explore the ideas surrounding generative architecture over the last few years - I hope you two realise that this is all your fault.

Further thanks go to my classmates who made the heat of WG203 bearable, in particular Matthew McFetridge and Frano Bazalo for technical insights and enthusiastic discussion.

Endless thanks go to my friends and family for support, conversation and coffee, and especially to my parents Pamela Breek and David Welch, whose support has been unwavering since I started this journey.

I promise I'll find something else to talk about next year.

Contents

0.1 Abstract

Parametric design tools and visual programming languages are fast becoming an important part of the architects design process. A review of current literature notes that the barrier to entry into the medium is lowering while the power of the tools available is increasing. The purpose of this research is to use these emerging tools to explore complex architectural issues related to space planning and massing. This research aims to bring these aspects of the design process together to generate an architecture where programme and aesthetic are derived in equal measure by the architect and the computer.

The project began with a series of technical studies focusing primarily on space planning, massing, site analysis and circulation with the purpose of using an amalgamation of these techniques to develop into a final generative algorithm.

These ideas are explored through an open ended design process of iterative research and testing, self and peer review, development and critical reflection. The viability of the algorithm is then tested through the generation a number of test buildings, across variety of sites.

In order to provide a direction and author a degree of creative friction within the research process, the projects are framed around the development of a mid-size, urban sited secondary school.

The final algorithm 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. This process, described as an "open box" structure, produced a wide variety of working concepts and provided a high level of control as a designer.

0.2 Glossary of Terms

Architects operating within digital space have developed their own nomenclature, some of which is subject to multiple interpretations. This section clarifies my own interpretation of this terminology in the context of this thesis.

PARAMETRIC DESIGN

Though widely used to communicate a range of different ideas (Davis, Chapter 2 - The Challenges of Parametric Modelling, 2013), in this thesis specifically it is used in conjunction with a "Top Down" design process. This refers to an algorithm designed to produce specific predetermined geometry.

GENERATIVE DESIGN

a cyclical design process of translating creative ideas into computer code, as outlined in the book of the same name. (Lazzeroni, Bohnacker, Laub, & Grob, 2009)

FEEDBACK LOOP

Refers to the iterative process described in the definition of "Generative Design."

TOP DOWN

"Proceeding from the general to the particular." (Oxford Dictionaries, 2013) In the context of this thesis, this refers to an algorithm designed to produce a specific pre-established design idea.

BOTTOM UP

"Proceeding from the bottom or beginning of a hierarchy or process upwards; non-hierarchical." (Oxford Dictionaries, 2013) A design process led by algorithms designed to produce novel, emergent results.

SPATIALISATION

Refers to the process of taking abstract information related to a building and turning it into concrete, volumetric space that can still be understood by the algorithm.

BLACK BOX

"A complex system or device whose internal workings are hidden or not readily understood." (Oxford Dictionaries, 2013) This refers to generative algorithms that the architect has very little understanding or control of.

OPEN BOX

Refers to the system developed for the final iteration of this thesis, a structure of individual, interacting algorithm that are transparent and simple to edit.

Literature Review

1.1 Parametric Modelling as a Tool

As the construction industry has transitioned into the digital age parametric design techniques have become increasingly integrated into the day-to-day workflow of the designer. By working with pre-programmed parametric tools architects and designers can produce traditional architectural geometry at an increasingly rapid rate. Though computer automation has done much to increase the fidelity and ease of modelling geometry, the loss of control can have drawbacks. Robert Aish critiques the tendency of commercial Computer Aided Design software to create an environment where the role of the designer is simply manipulating pre-existing variables of highly defined parametric objects, an illusion of decision making that works against a delineated design process. (Aish, 2003)

Parametric tools that appear to be 'black boxes' of obscured processes are, in actual fact, complex webs of interacting relationships between parameters; by limiting the user to only having control over inputs and ignoring the relationships between them, the user is at risk creating a rubber stamp of another designers mental process. Aish challenges this paradigm by reverting, in his work, to a blank slate, building the parametric rules and systems, the creation and manipulation of geometric relationships removing the constraints of working at a higher level.

Understanding the construction of geometry inside a computer is not a superfluous skill. Scheurer and Stehling argue that a high level of mathematical and geometric understanding is required in order to truly utilise computational design techniques as a practitioner. The order of operations and relationships within the description of a parametric model can have huge influence not only at a conceptual stage, but ultimately on how a building can be constructed. (Stehling & Stehling, 2011) Parametric design is not simply about using pre-existing parametric tools to increase work-flow; in order to truly take advantage of the computer you need to be able to understand and control the processes themselves.

1.2 Parametric Modelling as a Design Method

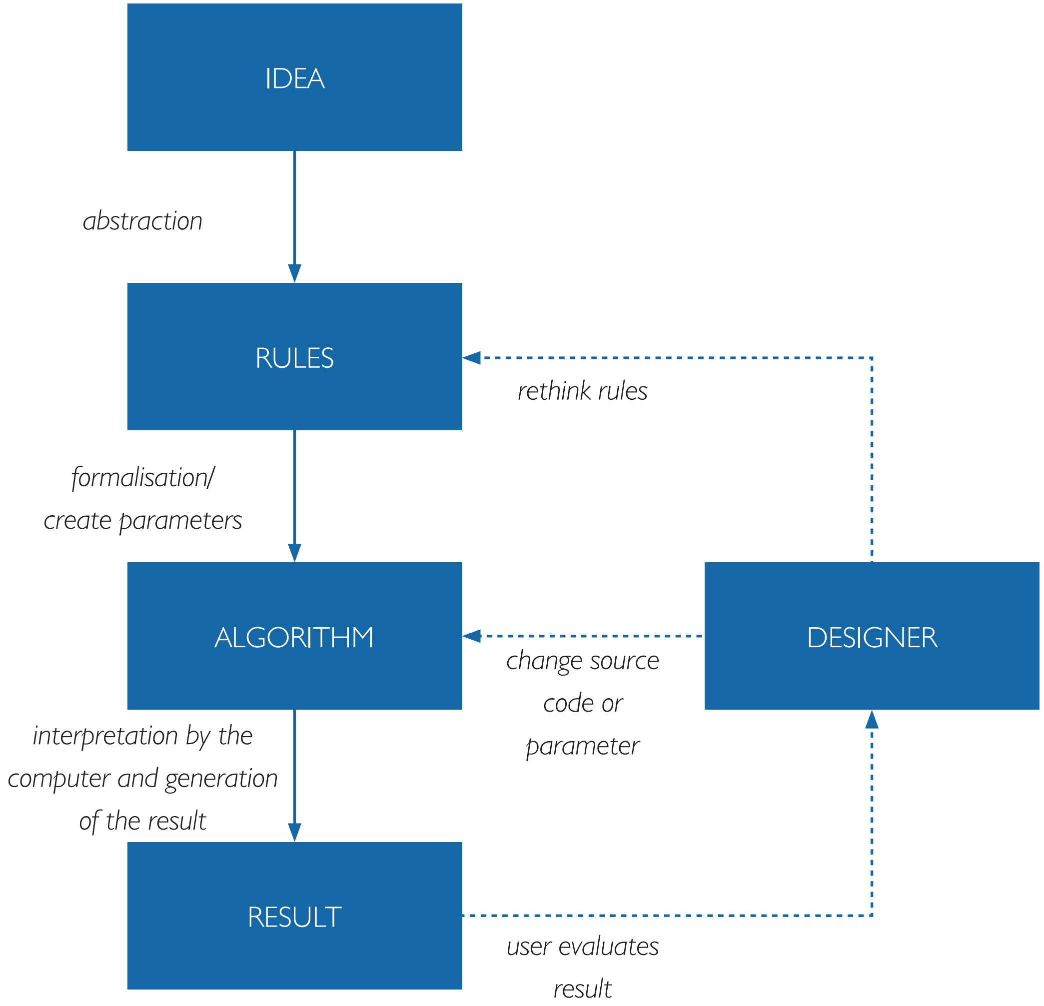
Conceptualising parametric objects simply as series of inputs and outputs is challenged by Burry, as it creates "an implied design process that appears to be the enemy of intuition." In his writing, Burry highlights the importance of considering the development of a parametric framework as a design tool, "a discussion between front end and back end" that allows the designer to interrogate the process of geometry generation at the same time as they develop geometry. (Burry, Between Intuition and Process: Parametric Design and Rapid Prototyping, 2003) If you design a parametric framework yourself, it becomes part of your iterative process, and gives you the opportunity to experiment quickly within applicable, useful boundaries.

In this mind-set, the computer shifts from being a representational tool towards being an active participant in the design process, an extension of the architect's mind that augments, and in some cases replaces, traditional design techniques. Kas Oosterhuis writes that he uses and works with the computer in a way that surprises him, his role being to adapt to a surprising, emergent system.(Oosterhuis, 2002) This requires having a large number distinct sets of data in a system interacting in such a way that the results are unpredictable.

The rapid feedback loop between parametric description and three-dimensional visualisation fosters an iterative environment where the process of developing a tool is as important as the tool itself, becoming a discussion between the designer and the computer about information, relationships, and feedback loops. The focus switches from geometry to processes. (Lazzeroni, Bohnacker, Laub, & Grob, 2009) The relationships that are forged within the data become the genome, and the geometry shifts as a result from the concrete towards a fluctuating sea of possibility.

fig. I

GENERATIVE DESIGN PROCESS



Note. diagram adapted from *Generative Design (p. 461)* by C. Lazzeroni et al.

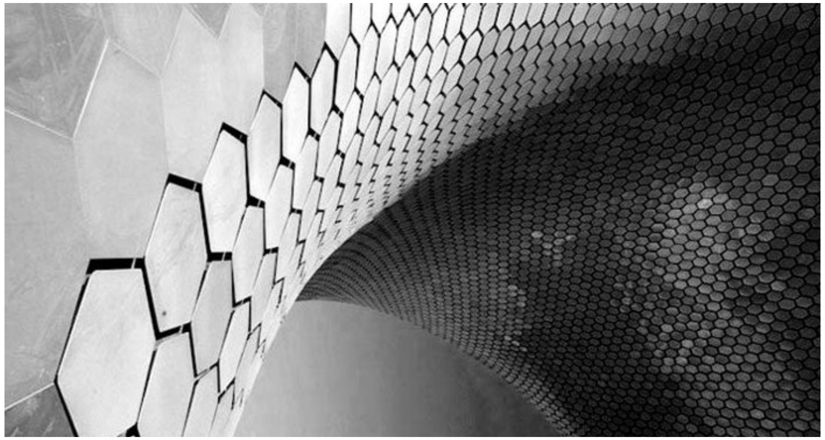
1.3 Existing Design Techniques

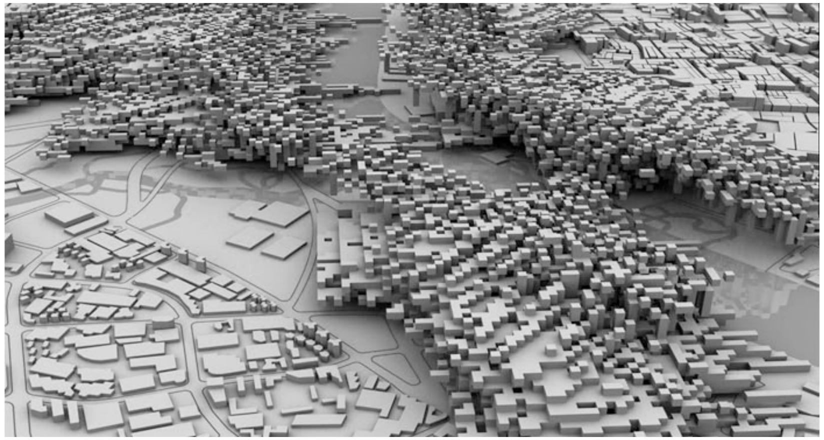
The discussion of where the computer can be used in the design process ranges from the early conceptual generator through to being almost purely as a digital fabrication tool. Mark Goulthorpe proposes that by using a 'top down' design approach and developing fabrication techniques for complex forms at the tail end of the of design-development phase, the resultant detailing and materiality is forced to become informed and innovative. Attempting to build the unbuildable "leads to the development of precise, rather than arbitrary fabrication techniques," as well as freeing one from the inherent conceptual limitations imposed by designing with existing techniques in mind. (Goulthorpe, 2003) By creating complex, arbitrary geometry and then developing fabrication systems within which to fit them forces you ask questions that you otherwise wouldn't have asked, and addressing these issues becomes a form of research in itself.

Conversely, by examining and emulating biological processes, Stanislav Roudavski suggests that generative design techniques can be used as a 'bottom up' approach to generating form and space. Dubbed 'digital morphogenesis,' the technique takes advantage of the computers innate ability to calculate massively complex relationships between parametric cells in order to explore an organic form finding process. (Roudavski, 2009) This approach focuses on using the computer as a design generator, with the sculptural nature and structure of the resultant forms a response to a few carefully created rules and relationships. Using the computer in this way revels in the idea that computation can be used to create an informed complexity that simply cannot be achieved through traditional design.

fig.2

TOP DOWN





BOTTOM UP

fig.3

1.4 Emergence in Complex Systems

There is a large focus in the industry today on using parametric design techniques as drivers for complicated form, using the processing of information as a design or fabrication technique and creating forms that would otherwise be impossible. 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?

Burry writes that "now [that] there is massive computer power and software cheaply available, most scripting has become nothing more than an onanistic self-indulgence in a cozy graphics environment. Endless repetition and variation, an elaborate geometrical scheme with no apparent social, environmental and technical purpose whatsoever." (Burry, Scripting Cultures, 2011) Useful spatial information can degenerate into noisy decoration if the data is simply being endlessly transformed.

While the 'top down' approach favours a designer-drive decision making process by nature, 'bottom up' generative processes have been used in a distinctly different way. Rather than focussing on complicated chains of transformations, a 'bottom up' approach uses the complex interaction of simple rules to produce emergent phenomenon. (Johnson, 2009) By building a complex multifaceted system that processes many interacting inputs, relationships and connections between stimuli can create forms and patterns that are impossible to produce otherwise. Examples of these stimuli this range from buildings generated from data driven environmental models to more subjective algorithms designed to explore social and organisational topics.

Rivka Oxman's work uses the parameterisation of performance criteria as a driver for design development, tweaking the response of geometry to specific stimuli in order to develop architectural form. (Oxman & Oxman, 2008) Ali Rahim writes that the interaction between performance criteria produces patterns and behaviours of the built space that could never be achieved from a top down approach. As well as resulting in unpredictable final forms, each space and surface of these 'performance buildings' is justified by responding to a specific stimulus. (Rahim, 2003)

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)

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 organisation. She writes that there is a tendency to use computers to explore quantitative data, and not use it 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) Focusing on more high level architectural considerations such as how a building sits in relationship to sight-lines, types of programme, and movement patterns can become a driver for architecture that is far more important than just as purely sculptural elements.

Aish and Schumacher independently identify the integration and interaction of different algorithms, techniques and scales as a possible avenue for future investigation. (Aish, 2003) (Schumacher, 2008) By bringing a numbers of interacting subsystems together, morphogenetic cellular architecture can further evolve towards producing robust, novel and emergent form. (Schumacher, 2008) Increasing complexity and eschewing “top down” control can aid the architect in producing surprising, emergent building systems.

In a situation where a generative design process is being driven only by a few complicated, transformative generators, it can become easy to predict where changes will occur as parameters are manipulated. Rather than analysing data to solve problems, the computer is being used to parse the data into predictable, complicated forms. This counteracts one of the key advantages of exploring the computer as a design tool: the ability for the design process to produce solutions that would not have been considered by the architect without those tools.

When data has a procedural prescribed translation throughout a system the outcome can be easily predicted the possibility of generating a novel solution is unlikely. The more complexity, the higher the number of individual interacting elements, the further the computer outstrips the designer at processing these relationships, and the higher the chance of interesting, emergent architectural form.

2

Design Proposition

2.1 Position

Complicated morphogenetic processes can be used to generate architectural form in a “top down” fashion. The computer is being used not as a generator, but as a processer. The danger remains that this processing focuses purely on the sculptural, ignoring the more notional and pragmatic concepts of architecture; the social, systemic and programmatic concerns that define a building.

Complex systems forgo processing and transformation, focusing instead on using simple rules that interact to produce unpredictable solutions. By intelligently organising and integrating a number of different generators, a ‘bottom up’ design process can be sculpted to eschew form and respond to truly architectural influences, allowing the computer to collaborate the architect, introducing emergent concepts in the early stages of the design process.

2.2 Research Question

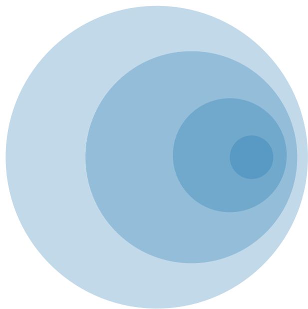
Complex interacting systems guided by simple rules can be used to produce novel, emergent solutions. Considering this, how can the interaction of programme, typology and contextual information be used to generate conceptual massing in architecture?

2.3 Scope

The goal of this project is to develop a holistic parametric model from distinct algorithms that overlap to generate self-organising, functional design proposals. Consequently, the focus is on developing a parametric framework for the interaction of inputs, not on developing a fully realised building to a high level of material detail.

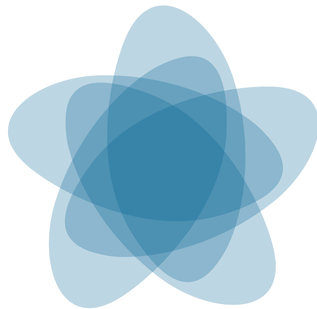
Though the research question is primarily focused on technology, the typology of the project has been restricted to that of an urban high school. This decision has been made because of the complexity of the aforementioned programme and site, creating complex problems that will force the planning and organisation of the building to respond intelligently, without simply camouflaging the fact that things are not working properly.

This approach to space planning and form finding treats specific parametric strategies as a means to an end, not an all-encompassing philosophy, bringing varied and disparate techniques together as they are required. The research is not focused on a particular problem-solving approach, but to integrate and explore different techniques that are presently available and seeing how the combination and layering of these techniques can be used to create viable architecture.



INHERITANCE + DEPENDENCE

fig.4



INTERFERENCE + SUPERPOSITION

fig.5

All parametric design carried out within this project is undertaken in Grasshopper3D, an architecture oriented parametric modelling tool. (Davis & Peters, Design Ecosystems: Customising the Architectural Design Environment With Software Plug-Ins, 2013) The strong focus on 3D architecture and design allows for a low level of coding knowledge, simple debugging and fast iteration that will greatly speed up the pace at which work can be undertaken.

2.4 Methodology

There are three generic approaches to design research: research about design, research through design, and research for design. Each of these strategies can feed into each other throughout the design process, and can be utilised to help identify and develop new knowledge. (Downton, 2003)

The majority of the project will consist of research through and for design; a process that will test the effectiveness of the parametric, interaction driven design process at all scales of the project.

Research for design will be undertaken throughout the project in the form of a series of short technical experiments exploring various aspects of the form finding and spatial analysis that will inform the parametric modelling process. Research will continue in parallel throughout the design process, as new technical opportunities and problems are identified and dealt with.

Research through the design of a series of iterative tests early in the project will help identify gaps in technical knowledge that will need to be addressed by research and experimentation before design development begins. By analysing and exploring each element that makes up the parametric model separately, the overall form can be driven by the choices of the architect. In this case, the main design decisions are derived from exposing or repressing interactions between elements in the matrix.

Transforming the abstract, quantitative data processed by Grasshopper3D into a representation of physical form requires subjective designer input. Reflection and critical analysis will be required in weighing up whether design decisions are informed or merely a technical compromise. Reflective iterative design research will be required to codify design decisions and produce an informed generative design process.

3

Preliminary Design

3.1 Discussion of Research Method

"Problem decomposition" is an important consideration when transforming a complex computational process into a manageable 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 organisation and generation of space, and then producing a series of integrated tests. Though creating these individual algorithms was a relatively straightforward process, amalgamating them into a cohesive whole proved to be a challenge. Building on this work, work transitioned into controlling and intelligently connecting these individual processes into a working framework.

3.2 Space Planning

Exploratory testing began by evaluating the viability of using genetic algorithms as a space planning technique. Genetic algorithms are useful for exploring and optimising arrays of variables by selectively sampling large probability spaces and weighing different iterations based on a predetermined 'fitness factor.' This can theoretically produce viable solutions for open ended problems, leading for them to be extensively explored as a tool for floor-plan optimisation and spatial organisation. (Rutten, 2013)

Inside Grasshopper3D, different nodes on a grid were fed a matrix of relationships to one another derived from techniques explored by previous space planning researchers. (Schneider & Koenig, 2012) (Merrell, Schkufza, & Koltun, 2010) (American Institute of Architects, 1981) The Grasshopper3D extension 'Galapagos' was then utilised to optimise the nodal relationships into a setup that had the least overlap and distance between different programmes.

Over a series of iterations it was observed that the algorithm generally produced subjectively optimal results if the variables that governed the fitness factor were carefully set and the probability space was narrow, focusing on only a few interacting datasets. Having too many variables produced noise, and the nuances of complex space planning were very hard to represent as weighted numerical values. Conversely, when the probability space was small enough to produce optimal results every time, there was generally an obvious solution to the problem anyway, negating the need for the tool.

fig.6

EVOLUTIONARY SPACE PLANNING
(SIMPLE)

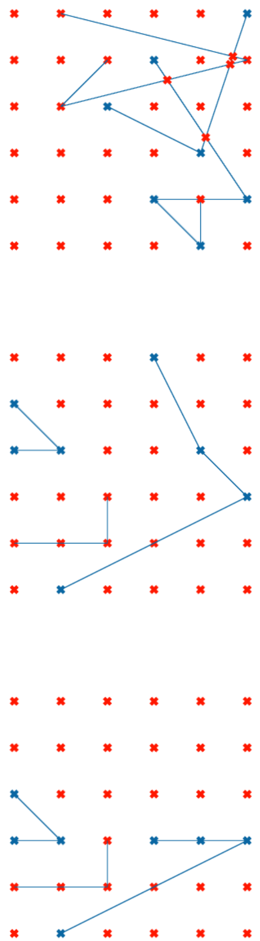


fig.7

EVOLUTIONARY SPACE PLANNING
(SIMPLE + WEIGHTED)

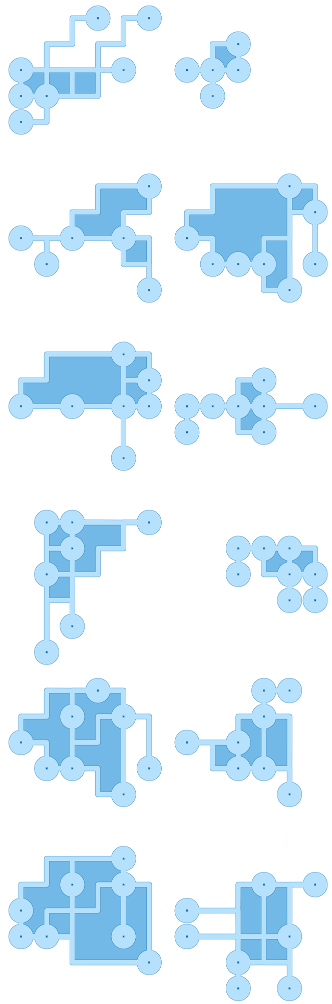
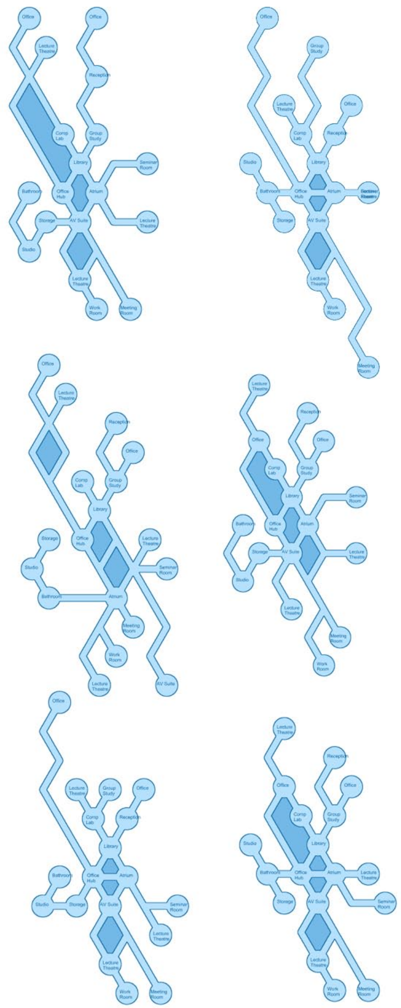


fig.8

EVOLUTIONARY SPACE PLANNING
(COMPLEX + WEIGHTED)



3.3 Massing

Experiments into genetic space planning were extended by applying the same algorithms to form finding, ignoring the spatial relationships between the spaces and instead focusing on minimising surface area and logically organising different sized spaces.

These tests, though they regularly produced optimised results, were less successful when considered as a small part of a larger process. The technique used to minimise the distance between the spaces was very simple and it takes a large number of genetic iterations to reach a foregone conclusion, rather than generating novel, unforeseen solutions.

As an alternative approach, tests were conducted by using a physics based plugin, "Kangaroo." The connecting lines between the different spaces, what was previously the fitness factor, were turned into compressive physical forces, which gave better, faster results and allowed much greater freedom in terms of modifying the form of the massing using different values in the physical simulation. (Piker, 2013)

fig.9

EVOLUTIONARY MASS PACKING

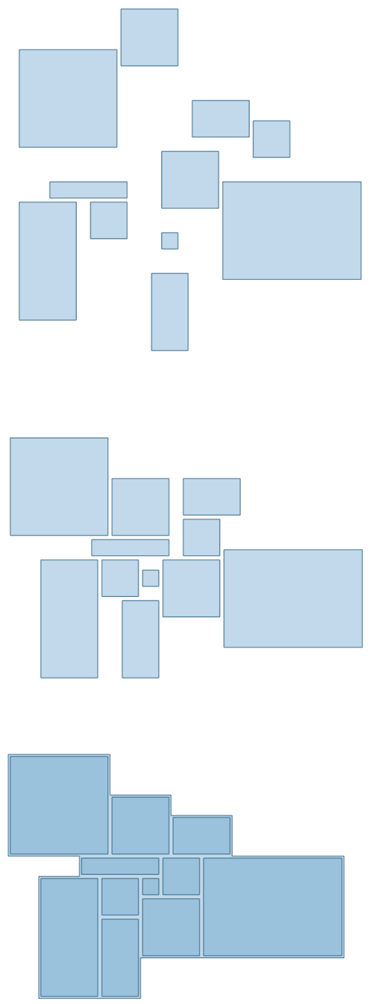


fig.10

PHYSICAL MASS PACKING

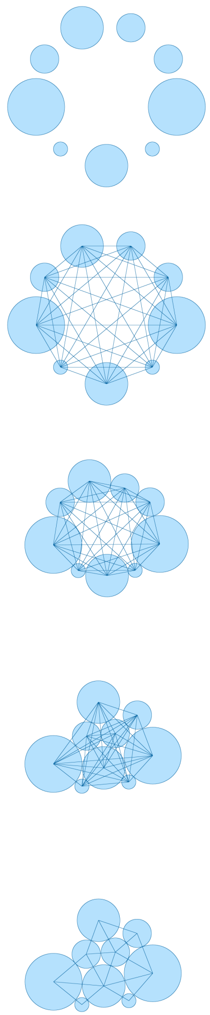
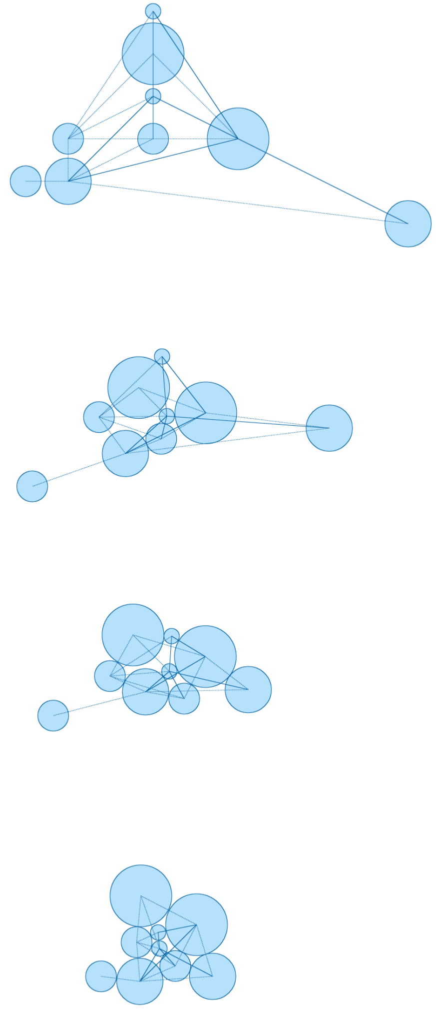


fig.11

COMBINATION MASS PACKING



3.4 Spatialisation

Massing and space planning up until this point had been explored separately – they have been purely diagrammatic tests exploring specific elements of the form-finding process, Several tests were undertaken exploring different techniques for translating these diagrams into real space in order to make these diagrams relate to one another.

Transforming the data in a way that maintained both its organisation and relative scale proved to be difficult. Initially, this was attempted using a 'flat" structure, snapping elements of the graph onto an underlying grid.

Though this made the spaces visually and organisationally simpler to work with, the lack of interaction between spaces did not feel compatible with the initial research direction. Expanding on the idea of creating noisy, overlapping spaces, further tests started to explore the concept of hierarchy, layers and levels within the translation from diagram to real space. These tests began to produce interesting and controllable results but the resulting spaces were quite removed from the initial volumes and locations determined by space planning. A different approach was needed in order to maintain the logic required to make the resultant building have an informed structure.

fig. 12

GRID SIMPLIFICATION

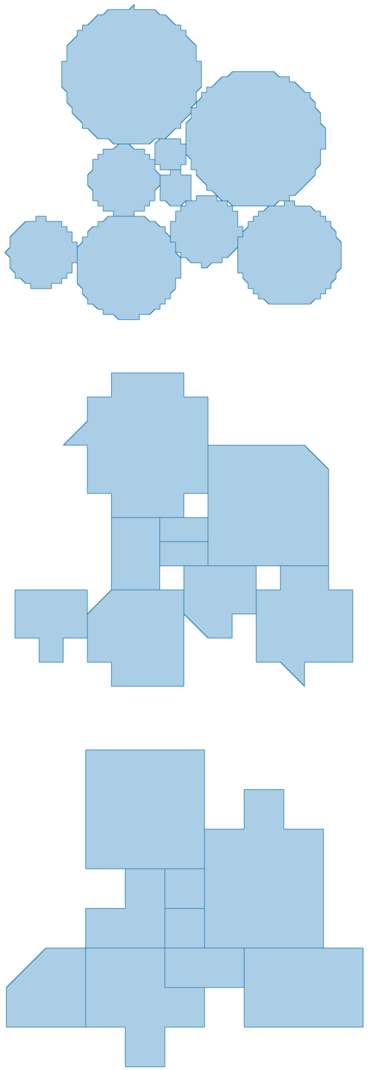


fig. 13

GRID LAYER SELECTION

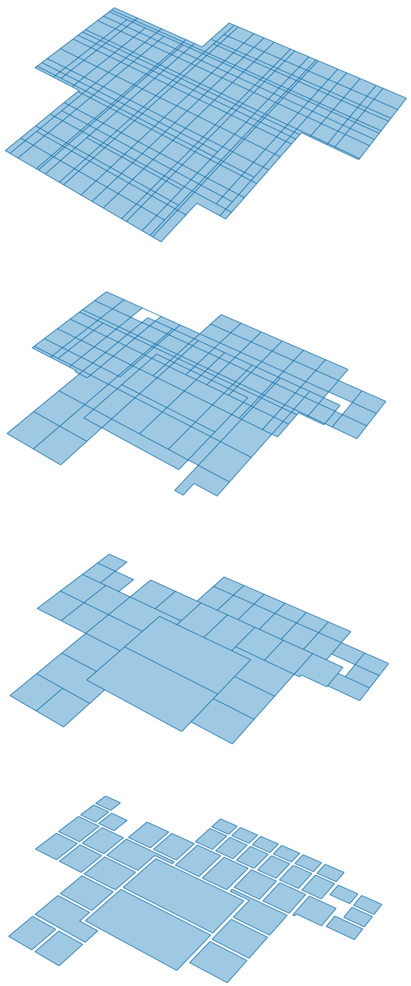
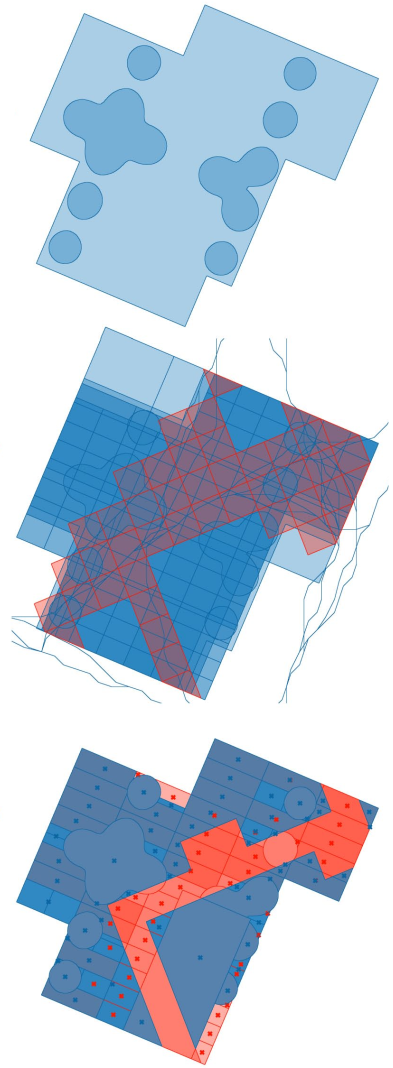


fig. 14

MULTISHAPE LAYER SELECTION



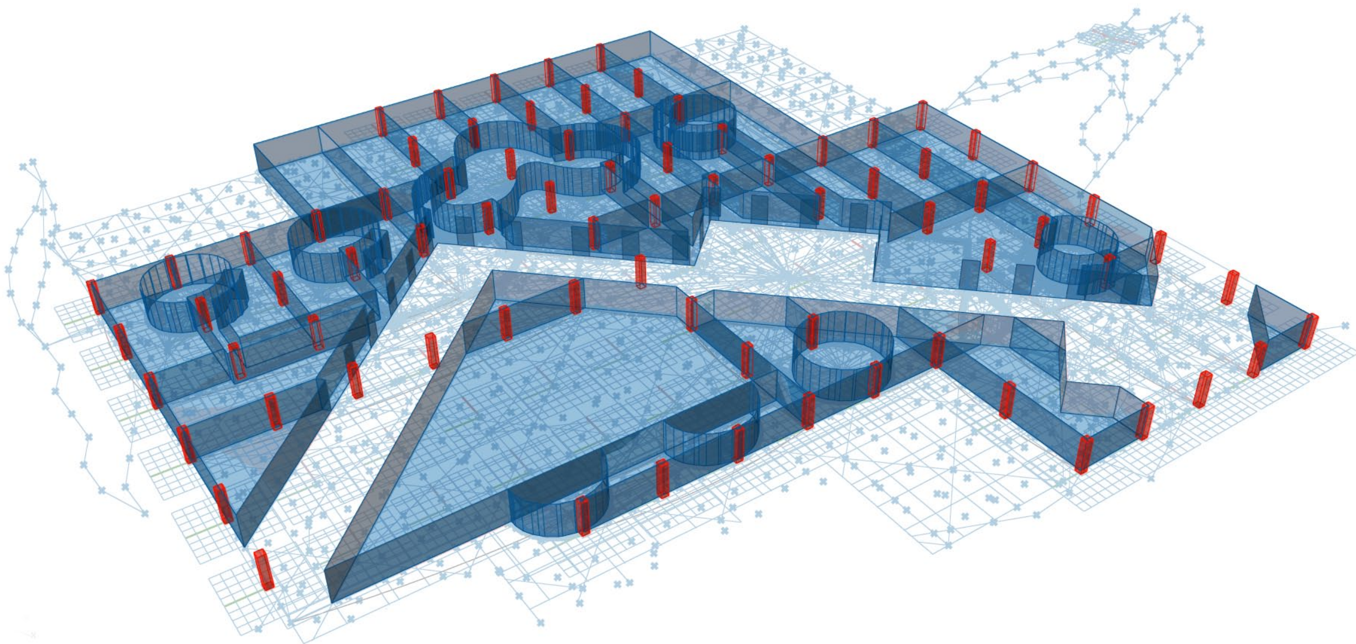
3.5 Integration Test I

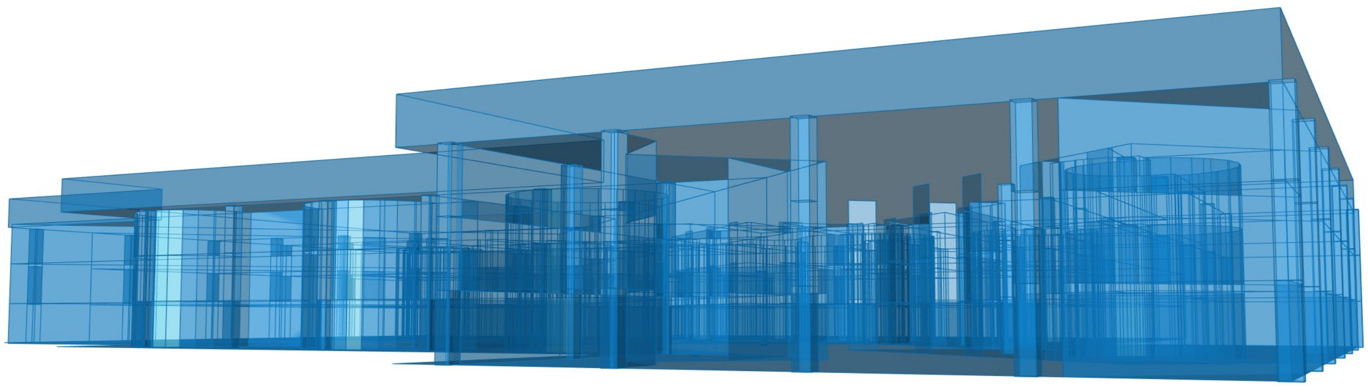
The individual space-planning algorithms were integrated together in preparation for the three month presentation milestone. This prototype built around using the interaction and overlapping of different layers of hierarchical information informed by genetic algorithms to generate a complex, informed system.

Using this system, loosely defined circulation spaces, structural grids, different scales of programme and breakout spaces all come together in layers. The final algorithm consisted of independent parametric systems, organised into a layered hierarchy that allowed for a degree of control on the designer's part, and produced novel, though chaotic, spatial conditions.

fig.15

ITERATION I SCHEMATIC DIAGRAM





ITERATION I PERSPECTIVE

fig.16

3.6 Critical Reflection

Though the breadth of technical studies was well received, the feedback on the first iteration was mixed. Notably, the reviewers were critical of the focus on two dimensional planning which ignores the critical factor of volume in my space planning.

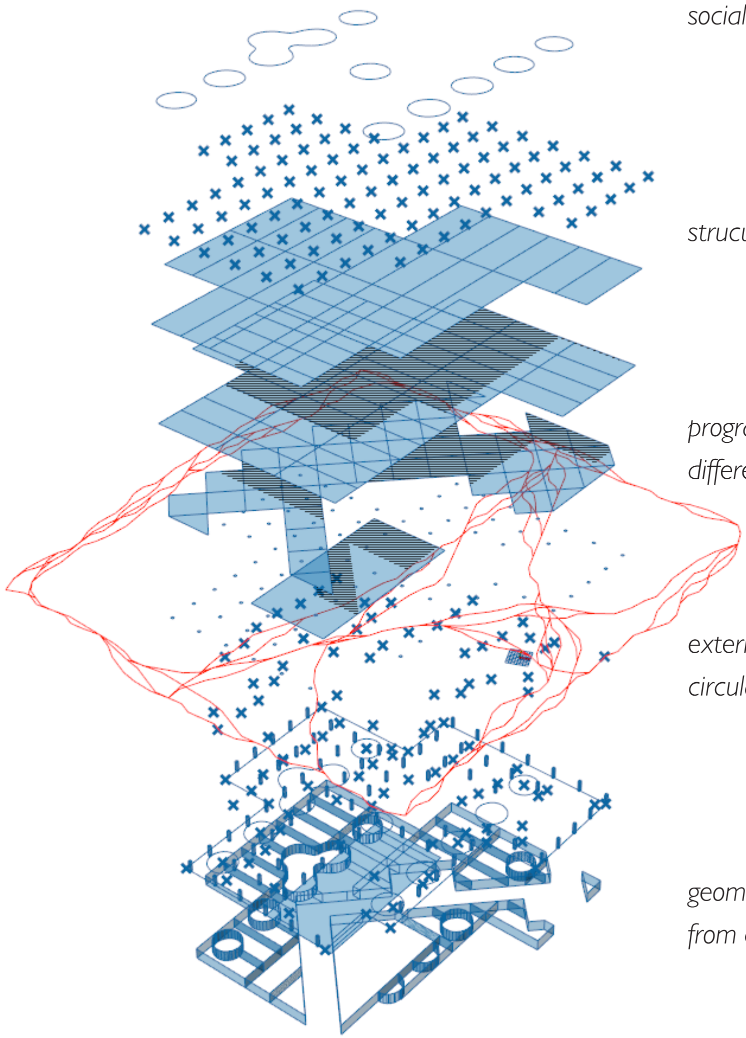
In retrospect, the pressure of producing something identifiable as a building led to the transformation of the space planning graphs directly into 3D space without fully considering the architectural potential embedded within the resulting outcomes of this process. Directly transforming the scaleless, representational diagram into geometry “forces the same expression at all levels of the project,” rather than naturally feeding into an informed piece of architecture. (Umemoto & Reiser, 2006) At a stage where the data driving the building was diagrammatic and loosely defined, turning the diagram into concrete and recognisable architectural had a hugely limiting effect on the process conceptually.

This carried the realisation that the pressure to squeeze geometry out of a set of data creates a scenario where the work you develop becomes compromise between the designer and the computer, rather than a series of informed, relevant choices. Working forward from here, the focus shifted towards more clearly evaluating the value of the diagrams and extending the amount of information available by focusing more on environmental and social conditions of the site,

Quantifying and abstracting elements of site required a range of different techniques; each one focused on a different aspect of the interaction with the surrounding environment. The overall goal of these techniques was to generate enough parsable data that the algorithm could weight different areas of world space against each other, allowing for the transference and spatialisation of the pre-existing space planning layouts.

fig.17

ITERATION I LAYER BREAKDOWN



3.7 Site Analysis

The A* shortest walk algorithm, a common AI path-finding technique was used to identify possible movement flows of people through the site. (Jurgen, Wagner, & Zweig, 2009) Direct paths through the site, bottlenecks, and adjacencies to heavy foot traffic could all be determined from this technique. The main issue with using this technique was weighing the data against aesthetic notions of site - a narrow alleyway may be picked up as a main thoroughfare, ignoring swathes of open footpath.

Some of the more promising visual feedback taken from these explorations was in the form of vector analysis, which used simulated magnetic attractor fields to influence different aspects of the site, resulting in images of data highlighting areas of rest and movement. However, extracting useful information for further computation proved difficult – patterns, though visually discernable, were difficult to isolate and extract as usable information. The same issues were encountered when exploring day lighting; using day lighting to highlight possible site layouts at a fine grained level proved to be easy to visualise, but hard to spatialise in a data driven manner.

Transferring purely diagrammatic information and transferring it into tangible delineated spaces proved to be a major challenge - how do you get an algorithm to define parcels of land, besides arbitrarily?

To answer this, a system was developed that generates a rectangular mesh based on the existing urban context in an attempt to utilise the surrounding site context as a method of demarcation. While this did create an underlying structure for future work, the data needs to be carefully balanced and tweaked in order to inform the architect's intentions.

fig.18

SHORTEST WALK SITE ANALYSIS

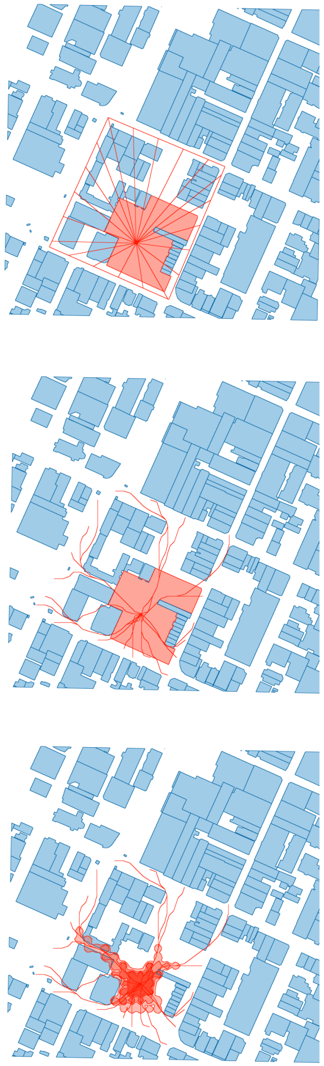


fig.19

POINT GRID SITE ANALYSIS

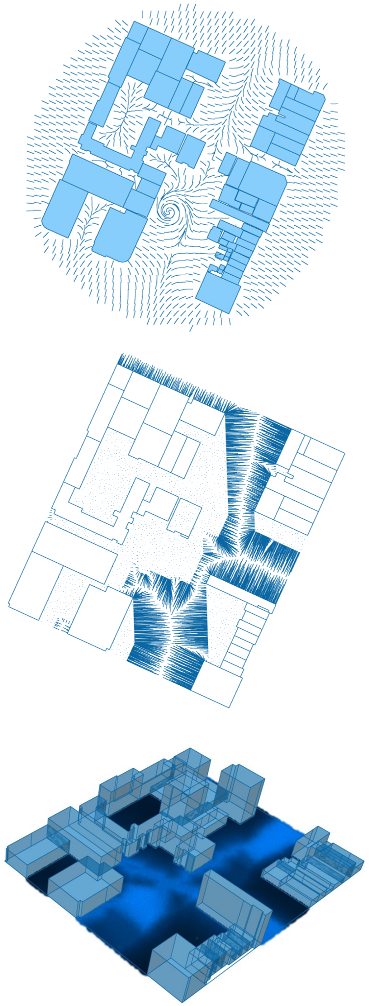
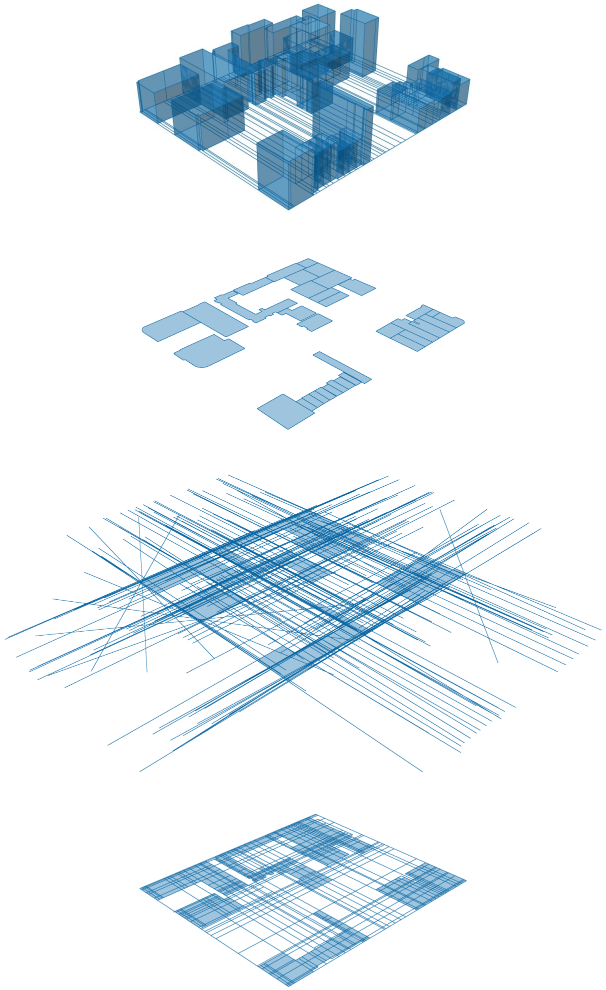


fig.20

CONTEXT DRIVEN SITE DELINIATION



3.8 Connection

After exploring the integration of the A* search algorithm in site analysis, a series of tests were undertaken using it as a technique for connecting defined spaces in logical ways. The shortest walk algorithms proved to be very fast at determining the shortest possible route between different regions, though initially it was difficult to get them to behave in a useful way; the proposed routes were too direct, ignoring each other and creating overly extensive circulation systems - very disconnected from the concept of informed simplicity.

This was addressed by using the Grasshopper3D plug-in "Hoopsnake, a recursive looping function that creates corridors incrementally, based on a predetermined spatial hierarchy. This produced a more efficient result, building only as many corridors as necessary, and creating new, interesting spatial relationships of its own.

fig.21

CIRCULATION ALGORITHM
(SIMPLE)

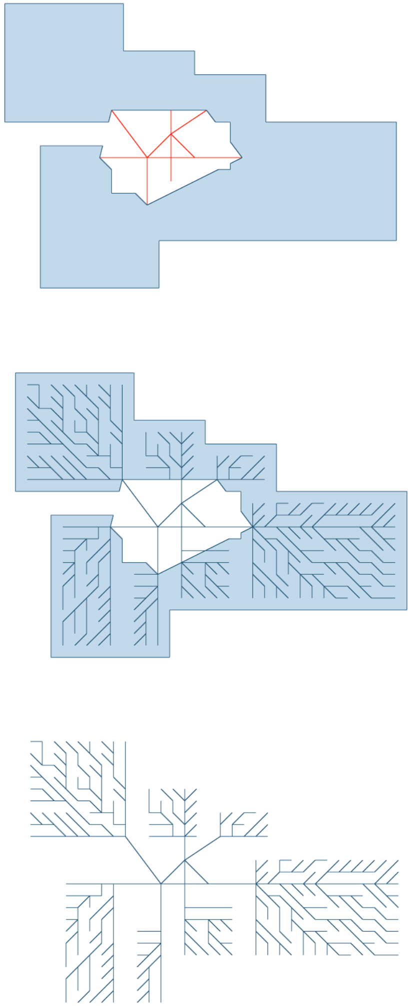


fig.22

CIRCULATION ALGORITHM
(RECURSIVE)

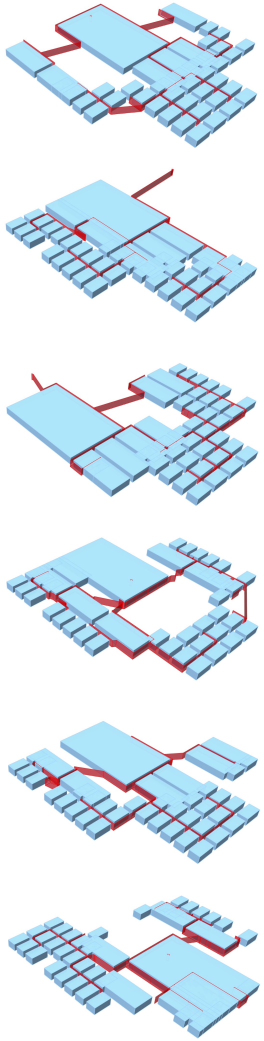
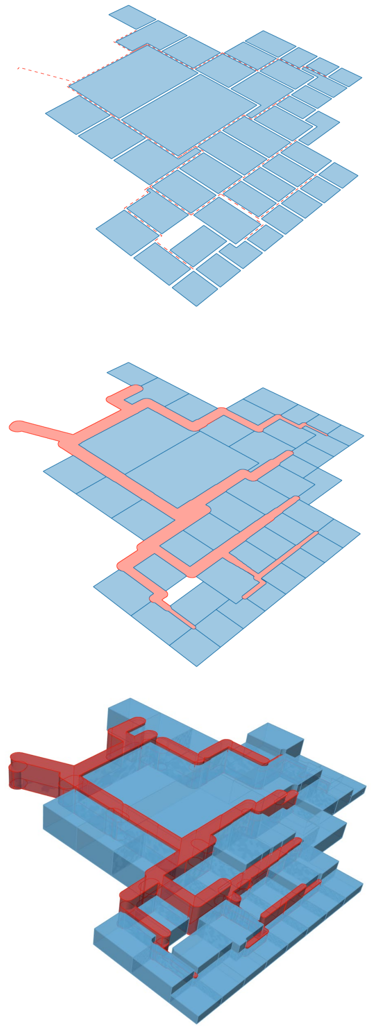


fig.23

CIRCULATION ALGORITHM
(RECURSIVE + VOLUMETRIC)



3.9 Vertical Propogation

A major barrier between creating multi-storey buildings in Grasshopper were the organisation and structure of lists and data. Looking into ways of structuring lists within grasshopper allowed for the possibility to integrate double and triple height spaces into the design, as well as separating the concepts of floor plates and volume.

This was explored using the same bottom up, procedural “Hoopsnake” definition used in the circulation experiments. A generative, looping approach was used to inform each floor as to the programme of the floor below. Though this work produced interesting results, the structure and scale of the data management in order to get it running was far beyond what I expected. The organisational algorithm itself has its uses, but the generative aspect of the tool was clumsy, and slow to iterate design outcomes, requiring that for each tweak that the simulation is run over and over. This highlights a key limitation of grasshopper - loops and recursion are very hard to integrate into the parametric work flow. Moving forward, attempts would be made to develop a flat, deterministic approach to vertical propagation.

fig.24

VOLUMETRIC SPACE PLANNING (LINEAR)

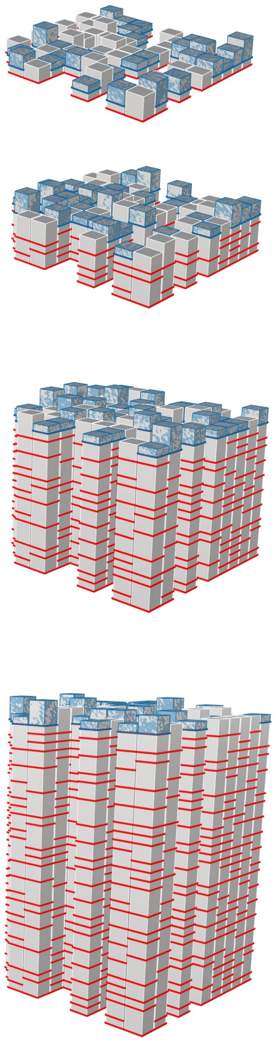


fig.25

PLANAR SPACE PLANNING (NON-LINEAR)

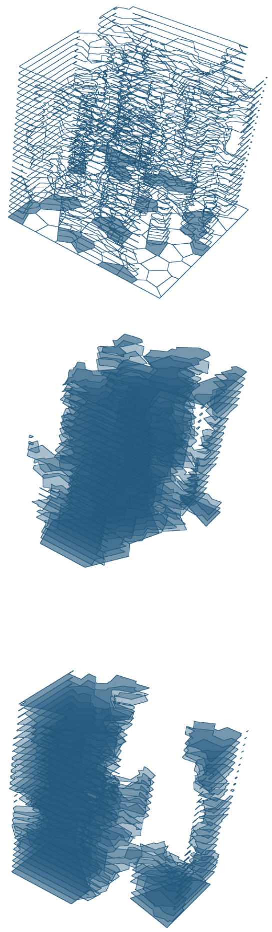
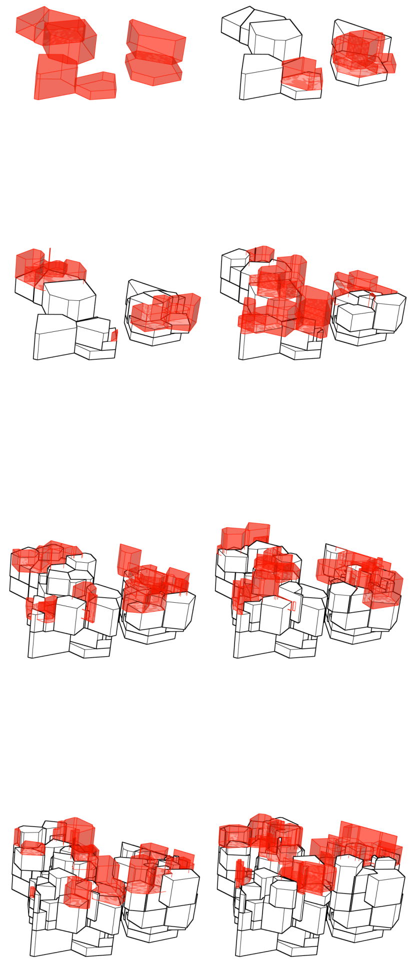


fig.26

VOLUMETRIC SPACE PLANNING (NON-LINEAR)



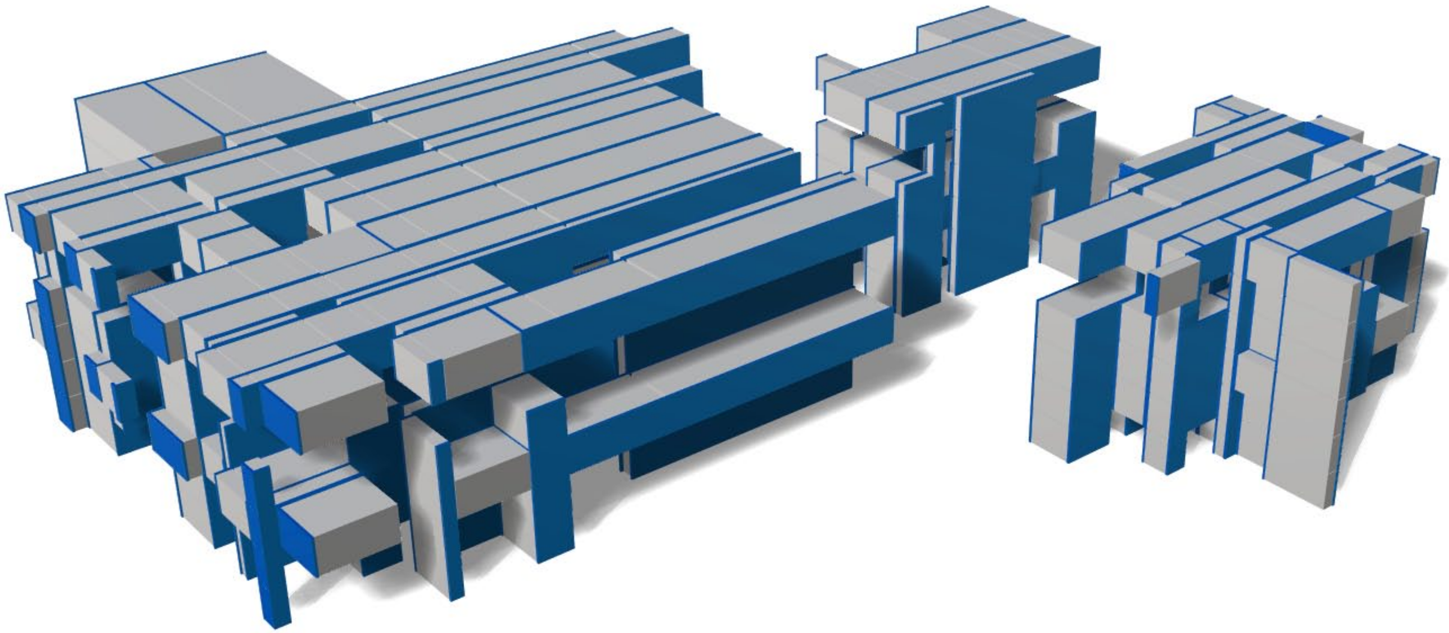
3.10 Integration Test 2

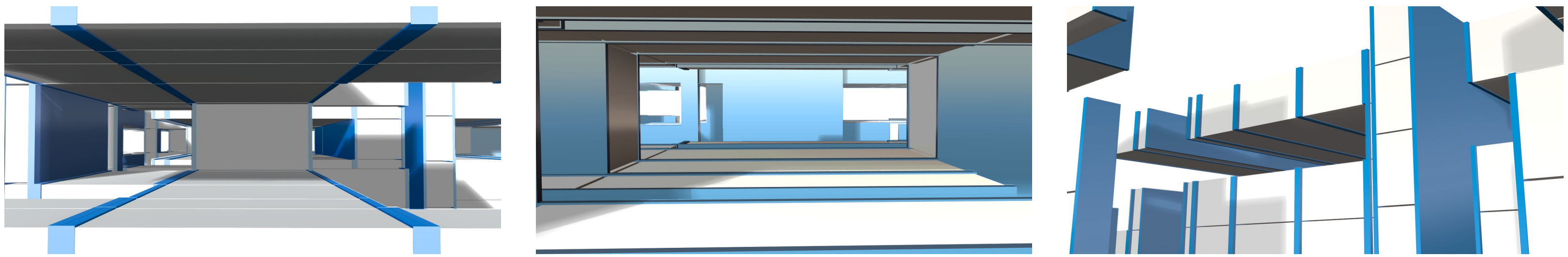
The second major iteration was an attempt to transfer my space planning information into a 3D space - one that appreciated context and adjacencies and placed buildings on the site. Integration of these techniques into the existing algorithm created a definition that could generate a space with height, volume, floor space, connectivity and programme, allowing it to be evaluated in three dimensions.

Evaluating the resultant forms from these experiments reveals a tangle of interference with very little apparent logic or structure. The spatial diagrams that inform the model have a pre-set, highly organised hierarchy; these relationships are lost in the noisy interaction with other elements of the program. This led to the re-evaluation of using a parametric model with a flat hierarchical structure.

fig.27

ITERATION 2 GEOMETRY GENERATION





SELECTED EXTERIOR PERSPECTIVES

fig.28/29/30

3.11 Critical Reflection

In order to be useful, the definition needs a way of abstracting the programmatic information of the building into a format that is simple enough to spatialise without losing or distorting it completely.

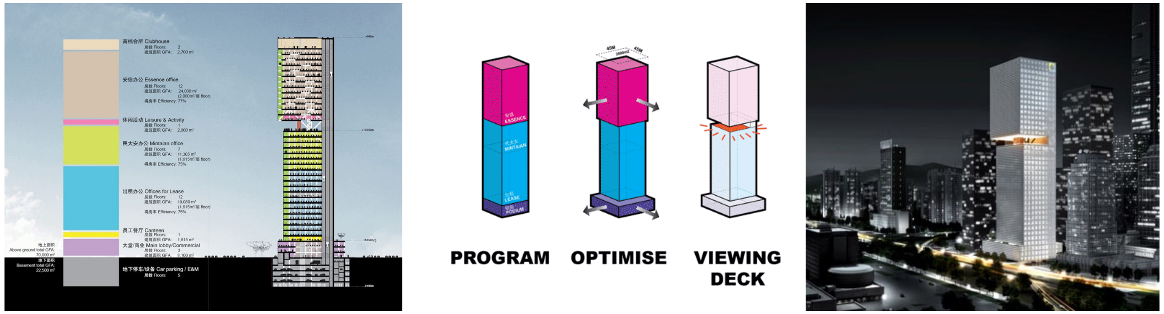
The firms OMA and BIG both use techniques for structuring programmatic diagrams as a part of the design process that, though they are designer driven decisions, still contain elements of a procedural nature that could be incorporated into a wholly digital work-flow.

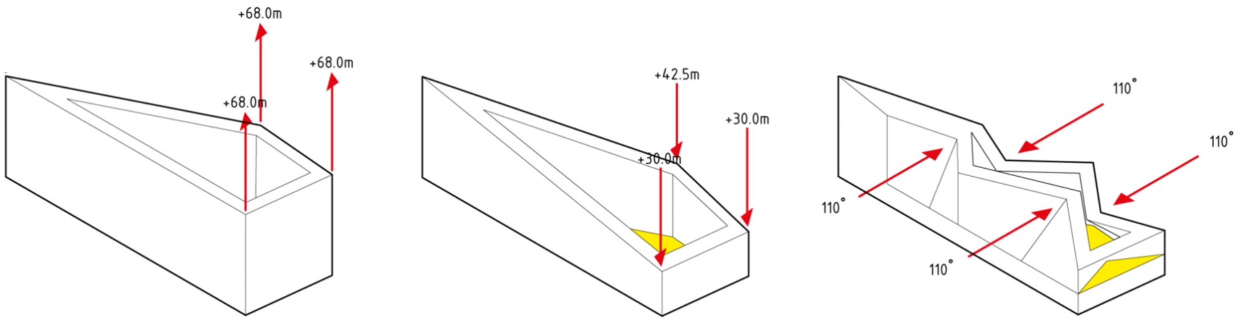
The BIG design process focuses on developing form through a series of simple formal decisions that incrementally consider the relationship between the built form and a small number of other stimuli - the site is broken down, and considered step by step, as a way of simplifying the design process. (Bjarke Ingels Group, 2010) If one were to parameterise these it would form a clear hierarchy of steps, compared to the relatively flat hierarchy that has been implemented in this project thus far.

Regarding OMA, the most important parallel between their work and a digital process is the firm's attitude technique of evaluating space as a data-scape, a graph, breaking down the brief and taking the control of the data at an early stage and developing the volume of the building as an almost direct representation of said data. (Price-Ramus, 2006) By formalising and sorting the data with a preconceived spatial element, the organisation of the programme is retained throughout the design process - a technique that could be used to decrease level of 'noise' that exists in the current iteration of this project.

fig.31

OMA DESIGN PROCESS





BJARK INGLES GROUP DESIGN PROCESS

fig.32

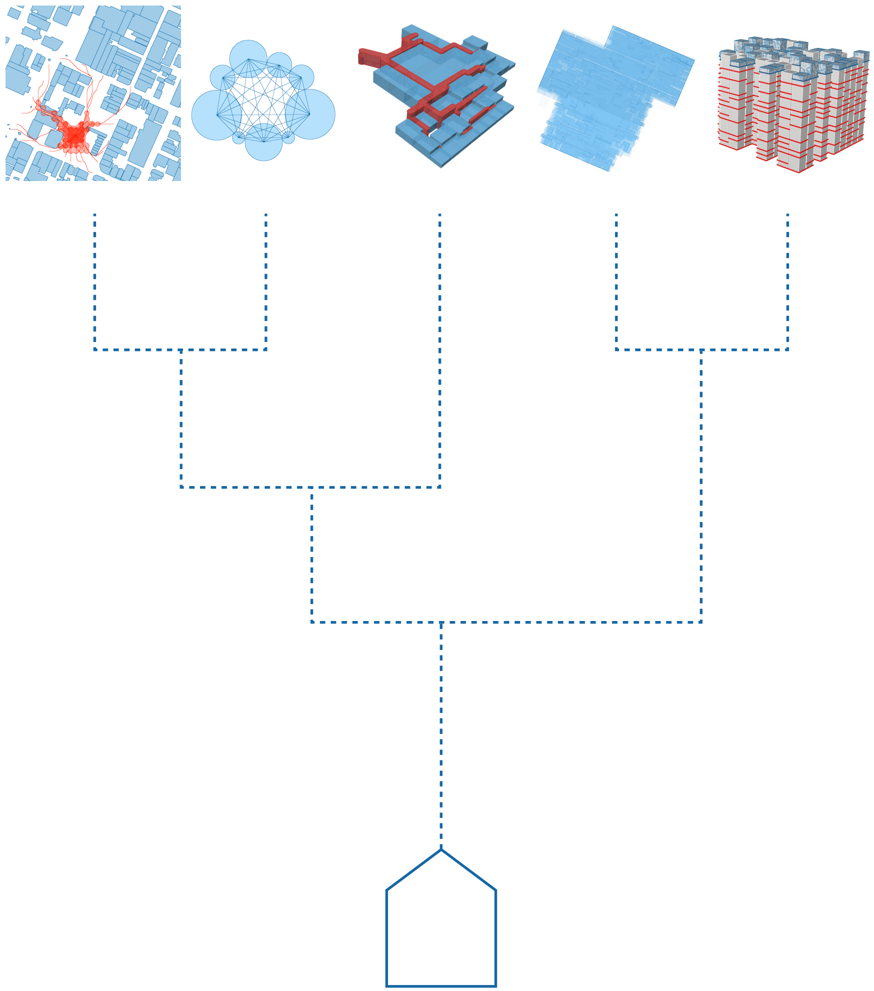
3.12 Moving Forward

The original strategy driving the development of this algorithm was to create a number of discrete processes and then create coherent architectural through integrating these processes together in a flat structure. The consequence of this procedure is that it is difficult to identify the useful relationships from the noise of unintended interactions.

Integrating a formal hierarchy and purposefully bringing relevant elements together in an ordered way better acknowledges the role of the designer in the process, improving control while still allowing discrete algorithms to create a compounding diversity of outcomes.

fig.33

REVISED CONCEPTUAL PROCESS



4

Developed Design

4.1 Discussion of Research Method

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 Grasshopper3D definition. By formalising and simplifying the intersections between the discrete processes the algorithm becomes clearer, and more responsive to change. This process, described as an "open box" structure, produced a wide variety of working concepts and provided a high level of control as a designer.

4.2 Limitation of Typology

The first step toward bring a semblance of structure into 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 spatialising the space planning data and as a response to creating outdoor space in a high density urban environment.

The space planning data, in this iteration, is explicitly organised into a single list, a bar graph of volumes that shuffle in one dimension based on data from a spread sheet. The benefits are twofold, both lowering the probability space for the organisational algorithms and allowing for a certain amount of organisation at the start of the project without having to allow for all possibilities further down the road.

This represents a shift in thinking towards developing the broad strokes of the building, and allowing for the focus of the parametric design tool to shift towards small-scale tasks better suited to calculation.

fig.34

DATA TRANSLATION
(SIMPLE)

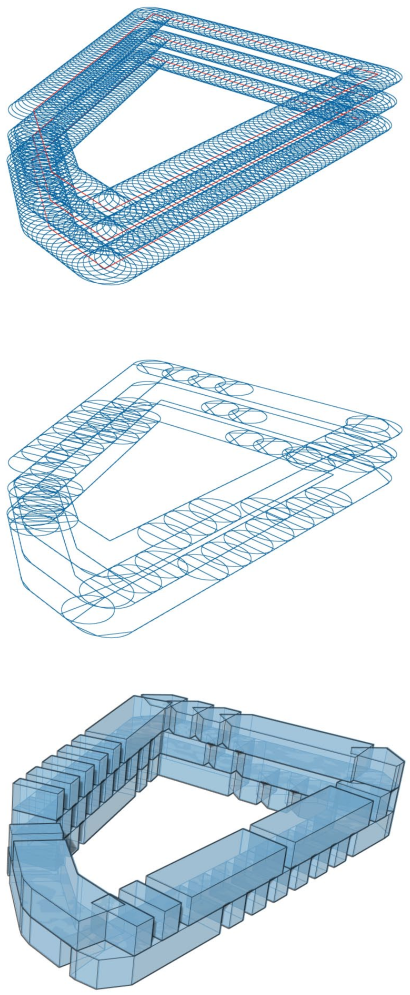


fig.35

DATA TRANSLATION
(PROGRAMMATIC)

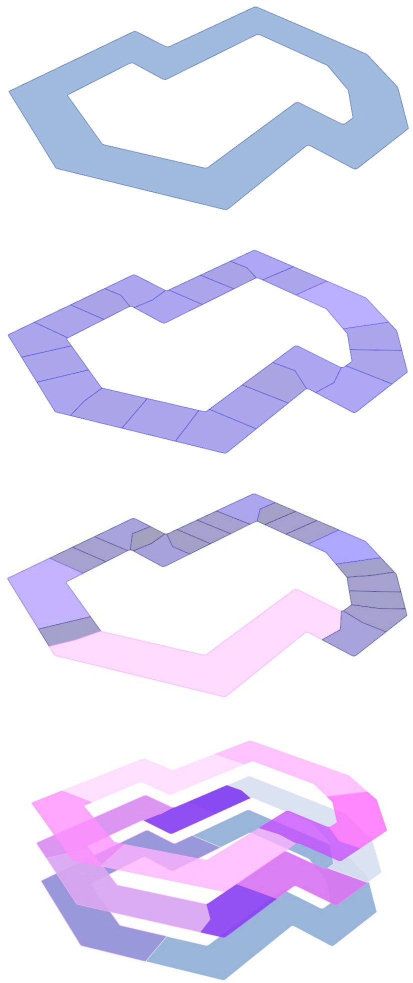
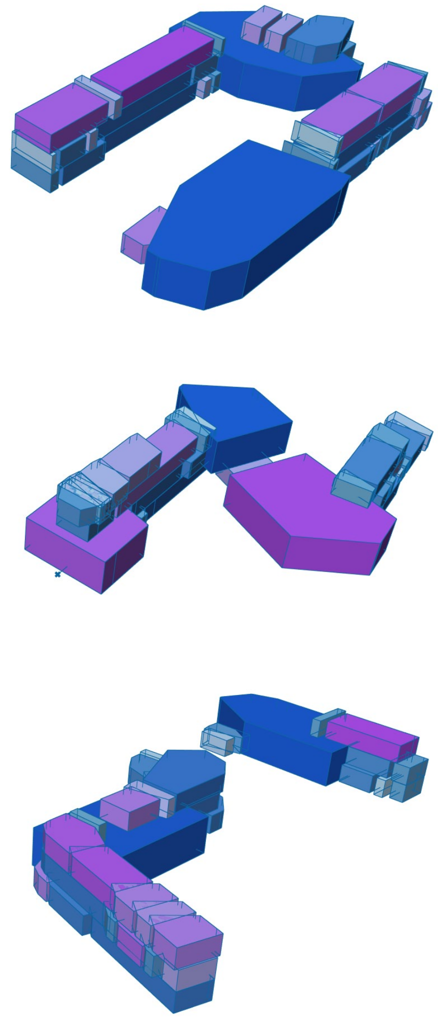


fig.36

DATA TRANSLATION
(VOLUMETRIC)



4.3 Structural Grid

A key challenge over the course of development was the question of how to transform diagrammatic information into 3D, logical spaces. After experimenting with multiple different techniques, a system derived from three dimensional meshing techniques was chosen. A mesh is derived from grid points extrapolated from the underlying building 'graph,' and then used to parcel out the site into individual units of space. These grid points can be derived from a number of sources - for example, points derived from building itself, site context, sight lines, movement of individuals, or abstract patterns.

Breaking the building down into discrete, itemised spaces creates a clear transition between diagram and real space, while also allowing the structural grid of the building to be determined, and influence the aesthetic of the building. A square typology will result in a regular underlying grid, while triangular, fractal meshes, and curvilinear meshes will impart their own unique aesthetics.

This procession of spatialisation is a way of putting all the logic of previous tests into a shared language of floor plates, walls, edges, roofs and volumes. Being able to finally clearly translate between the abstract and the concrete, the conceptual mass can now be considered in relation to the surrounding site context.

fig.37

PROGRAMME SPATIALISATION
(ARBITRARY MESH)

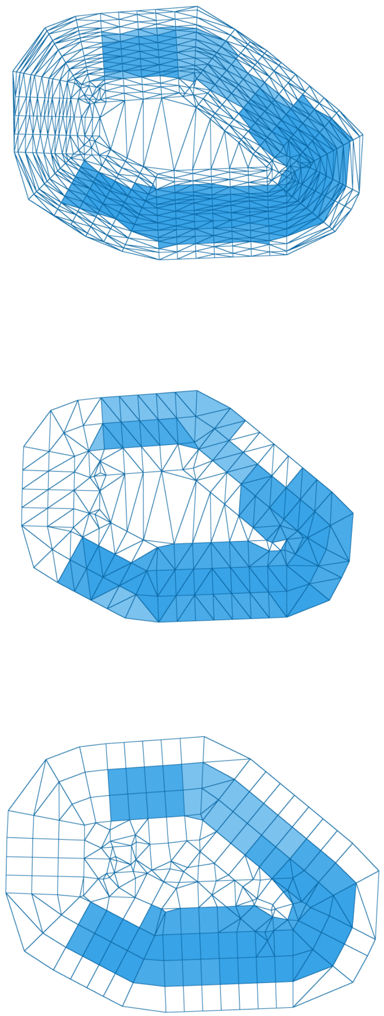


fig.38

CIRCULATION SPATIALISATION
(ARBITRARY MESH)

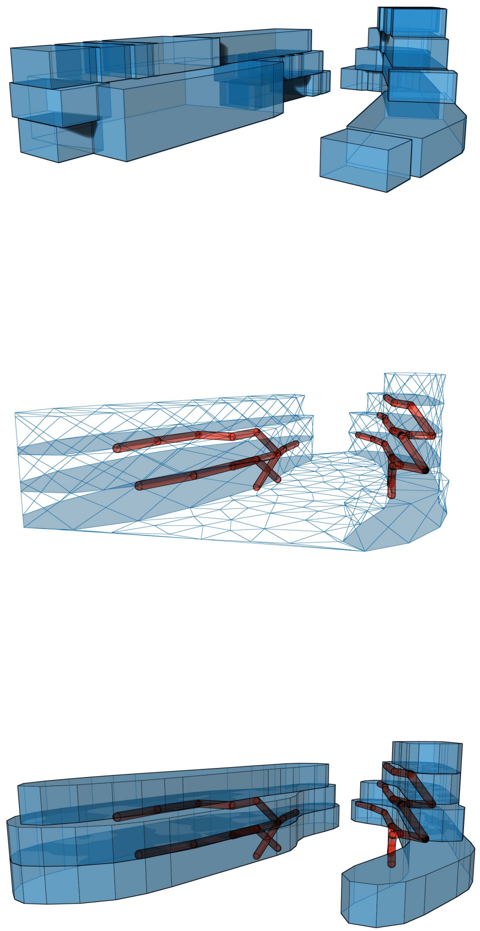
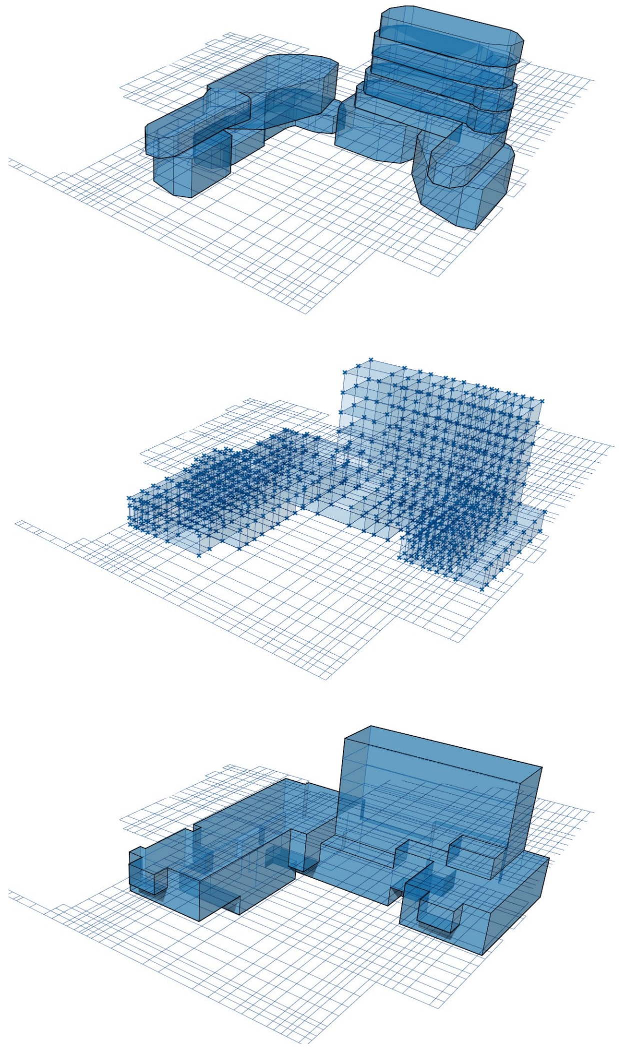


fig.39

PROGRAMME SPATIALISATION
(CONTEXTUAL MESH)



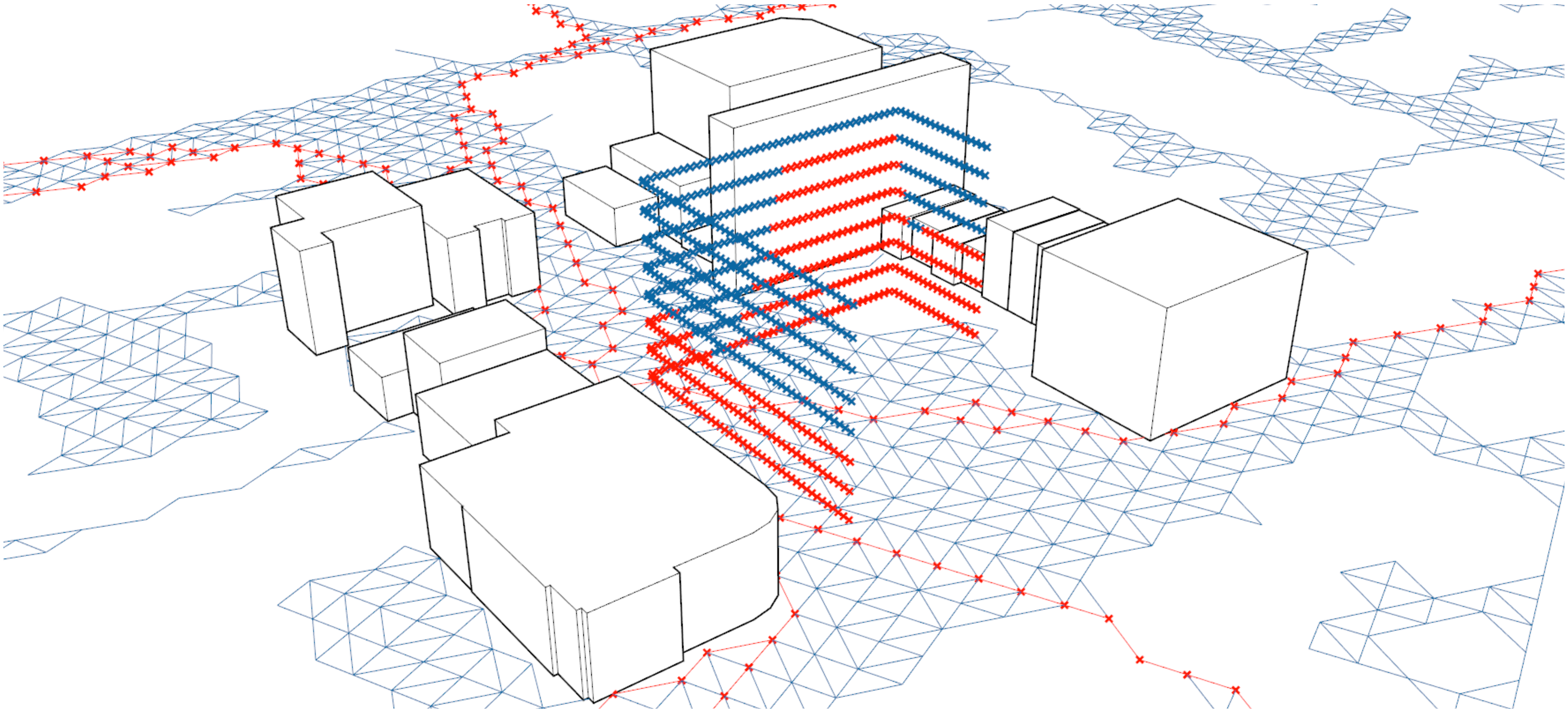
4.4 Integration of Site Context and Typology

The building footprint of the perimeter block is determined by a genetic solver that organises a polyline on the site based on repelling forces and shortest walk algorithms representing the existing site context and movement patters through the site. Using this formula, the polyline lays itself around the perimeter of the space, maximising coverage while minimising the number of points.

The polyline is then propagated upwards, creating an overgenerous list of nodes that will define the building program. 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 maximise sunlight in the courtyard.

fig.40

BUILDING GENERATION
STAGE 1



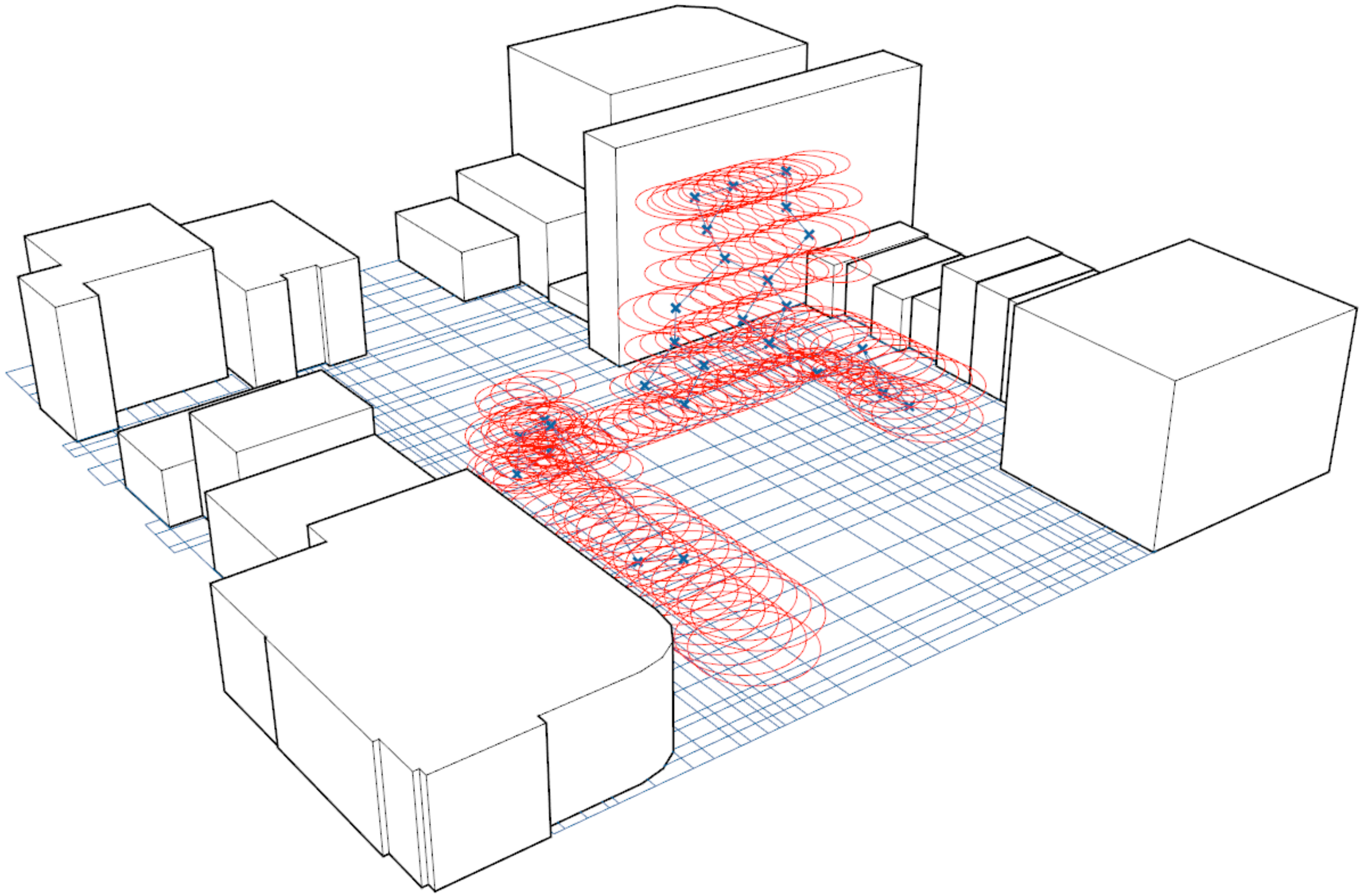
4.5 Integration of Typology and Programme

The programme is determined by spread sheet, imported and mapped to fill up the possible building space. A second evolutionary solver is then run, this time lcalculating adjacencies and layouts by organising and shuffling the programme list to optimise layout based on required adjacencies described in the original matrix. Double height spaces (the school hall and gym) are calculated first, allowing for the single height programme to propagate without interference.

All of this information is then translated onto the site through the lens of a context based rectangular mesh that determines and underlying grid. This grid allows the circulation and building envelope to respond intelligently to the conditions of the surrounding site, identifying discontinuities and existing volumetric relationships.

fig.41

BUILDING GENERATION
STAGE 2



4.6 Integration of Programme and Grid

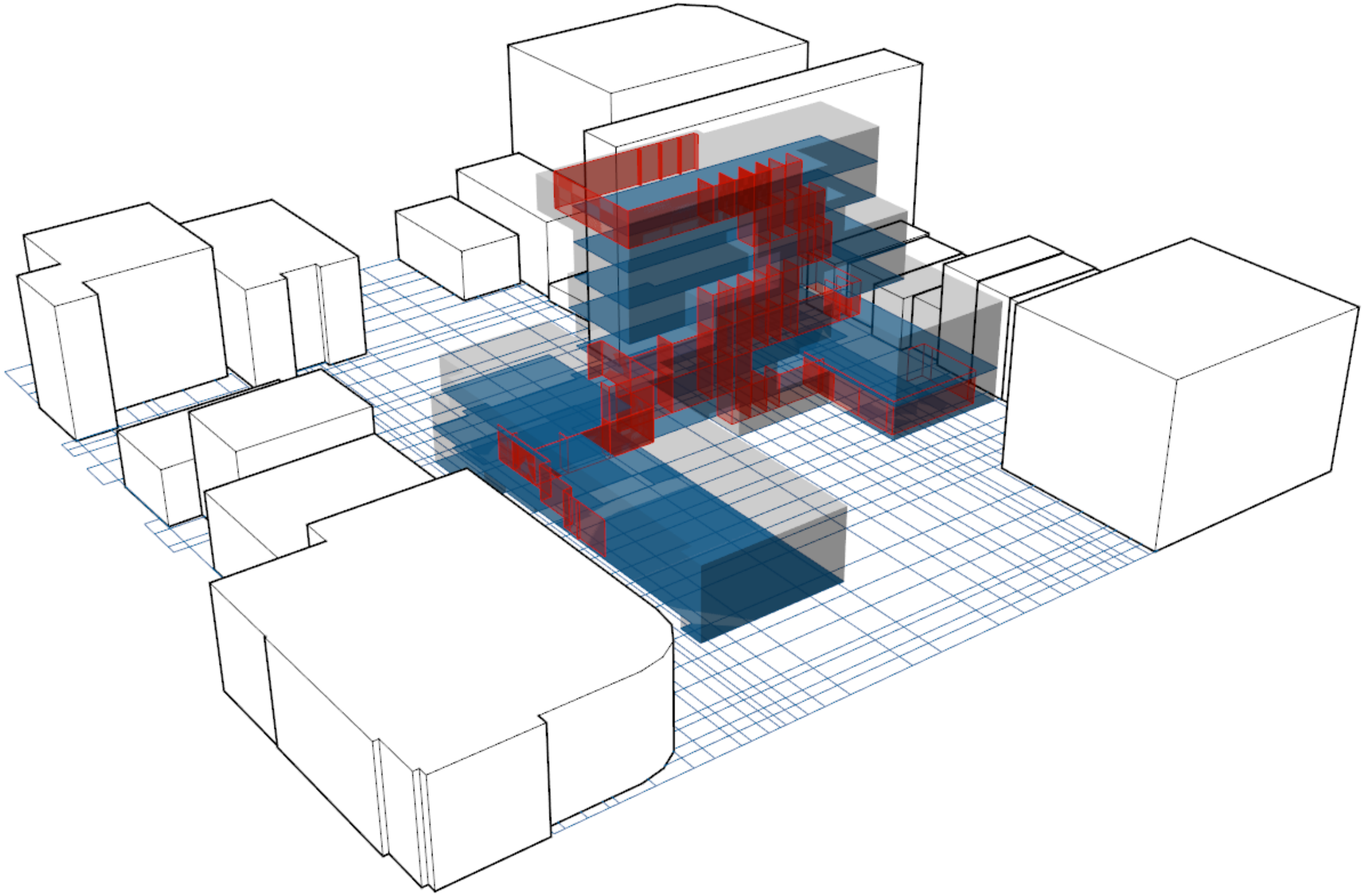
Once the volumes of the programme are loosely organised, it is directly translated onto the site using the independently developed underlying grid, in this case, a structural grid based off of the original perimeter polylines.

After the programme has been spatialised, the looping shortest walk algorithms are run, creating corridors and connections that feed throughout the building based on a hierarchy of importance.

Types of programme can be set to have different levels of interaction with the corridor system, so for example circulation lines will run through an atrium area but not through a classroom block or gymnasium. This can be determined by the designer in the excel matrix that defines the programme. By incorporating this kind of control, emergent and logical solutions can be found for circulation issues, such as elevated walkways running through double-height areas.

fig.42

BUILDING GENERATION
STAGE 3



4.7 Final Site Tests

With the algorithm integrated into a single Grasshopper3D definition, the iterative process can be run through seamlessly from start to finish without having to manually transfer information between files. Three distinct sites with their own unique spatial conditions were chosen as testing grounds in order to test the diversity of iterations that the algorithm produces.

16-180 BUTE STREET

A large open lot surrounded on three sides by a number of properties of different densities and heights, as well as complex foot traffic through the site.

THE KENT TERRACE MEDIAN STRIP

A unique spatial challenge, a long stretch of parkway in the centre of a major road.

88-94 JERVOIS QUAY

Chosen for the unique circular shape of the park as well as the coarse urban density, as well as the circulation challenges posed by having a highway on one side and the Wellington Civic Centre on the other.

For each of these sites, 20 to 30 iterations were generated. After the initial generation, two iterations for each site were analysed further, exploring the specific conditions that played a role in the generative process. The selection criteria for a successful iteration focused on the existence of novel or intriguing design elements that emerged naturally from the generation process.

BUTE STREET

fig.43

KENT TERRACE

fig.44

CIVIC SQUARE

fig.45

4.8 Bute Street Iteration 1

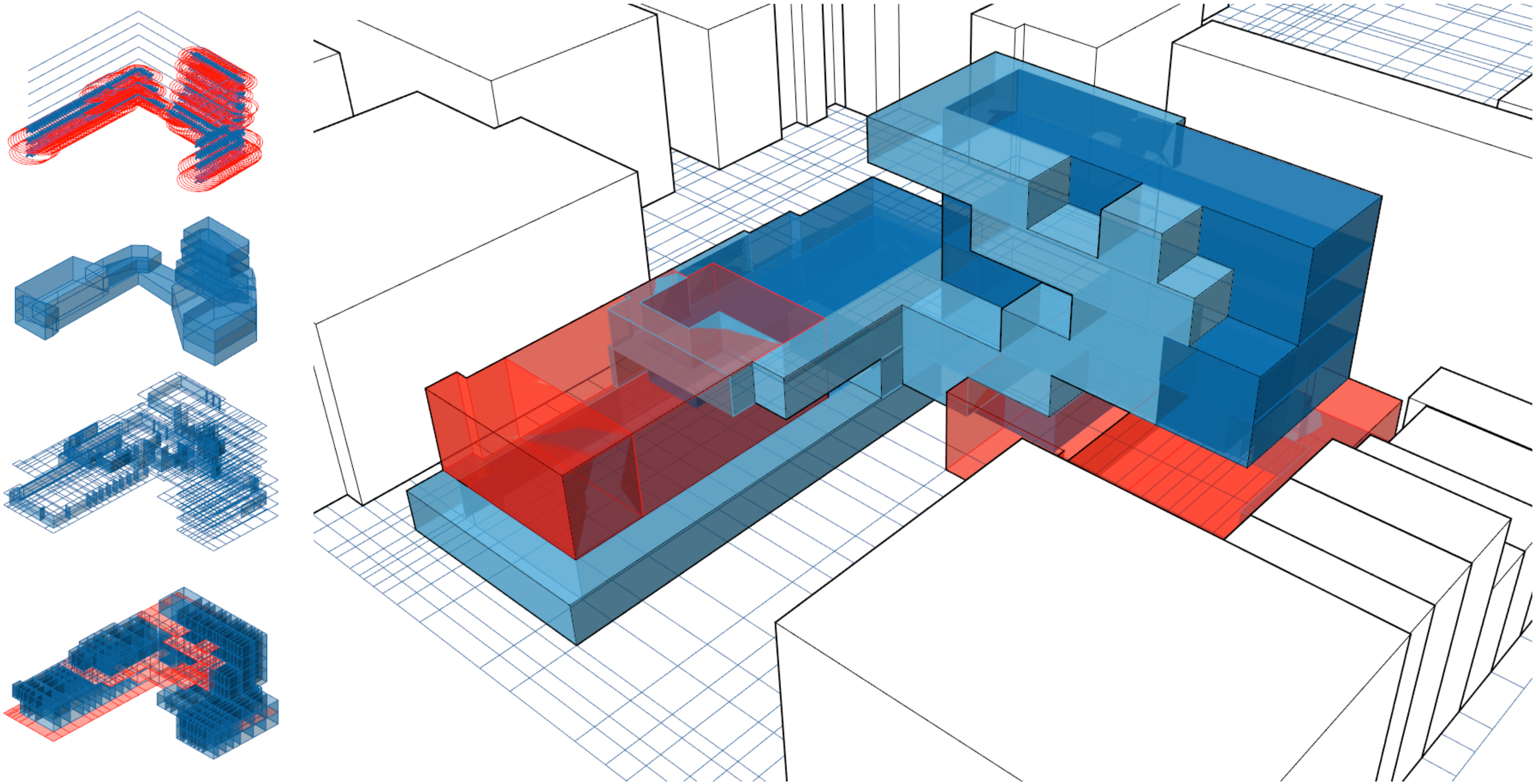
While all the iterations generated for evaluation use the same base algorithm, each iteration highlights different elements of the toolset.

The massing of iteration 1 is driven largely by the heights of the surrounding urban context, coalescing into two distinct tiers of height. The upper tier rises to match the tall thin apartment complex that flanks the northern edge of the site, while the lower tier is defined by the lower contextual buildings that dominate the rest of the site.

This provides a large uninterrupted container for the programme to propagate through. The double height spaces, the hall and gym, occupy this space, allowing uninterrupted classrooms and offices fill the rest of the container. The nexus of the circulation system sits at the central point of the L-shaped mass, and cascades up the southern face of the building to avoid interfering with the building volumes on the north and east sides.

fig.46

ITERATION 1: SOUTHEAST PERSPECTIVE



4.9 Bute Street Iteration 2

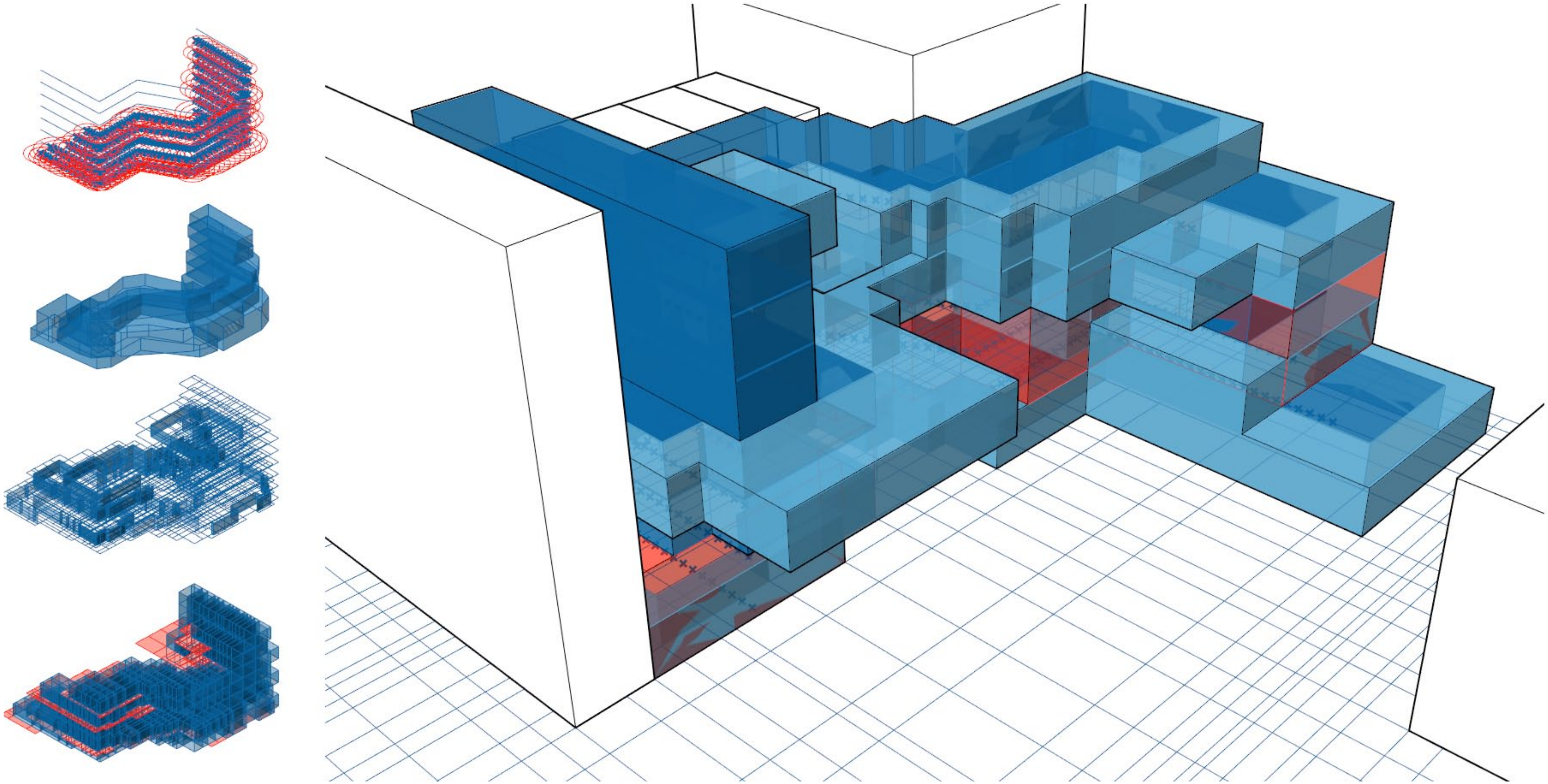
The second iteration has a much closer relationship to the urban context than its predecessor, which causes the massing of the building to cascade gradually down from the high northern apartments, towards a single-storey frontage on the southern edge of the site.

Due to a reduction in floor space, the programme is split in two distinct tiers. The gym and hall take up most of the lower space, with the rest of the programme distributed on the upper stories.

At the termination point of the circulation system sits a large cantilevered space that looks out over the site in all directions. This is a serendipitous event that has resulted due to an interaction between the scale of the contextual grid and the chosen circulation route. If this emergent space is well received, the designer can modify the algorithm to explicitly try to create these spaces, otherwise the behaviour can be removed. This process of refinement becomes part of the design process.

fig.47

ITERATION 2: NORTHWEST PERSPECTIVE



4.10 Kent Terrace Iteration 1

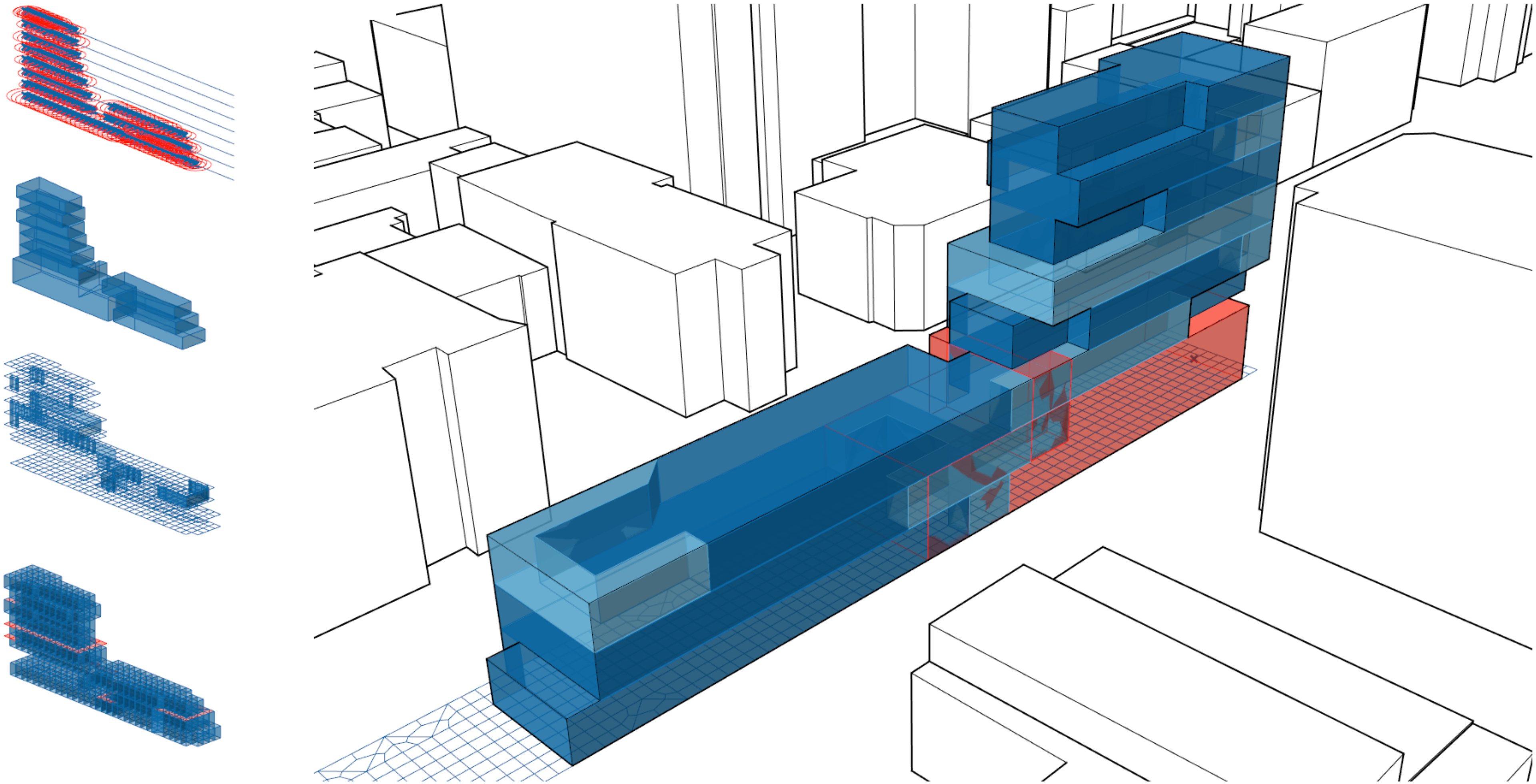
The narrow site along the centre of Kent Terrace serves as a stress-test for the genetic algorithms that organise the programme and the circulation system that connects them. The underlying grid that determines the buildings structure is highly regular, determined mainly in this context by the linearity of the road. The building has pushed up against the northern end of the site, the form matching the heights of the buildings on the east side of the median strip.

The gym and hall space, which would normally take up most of the lower floors, have been superimposed on top of one another by the genetic solver – this means that the space 'cost' of losing the hall was higher than removing any other elements of the programme, and thus was determined to be the optimal solution.

The circulation has failed to propagate on several levels, but it has formed a series of interesting relationships on the ground and first floor, creating a double height space at the centre of the building which, feeds into the first three levels in a compact space.

fig.48

ITERATION 1: SOUTHEAST PERSPECTIVE



4.11 Kent Terrace Iteration 2

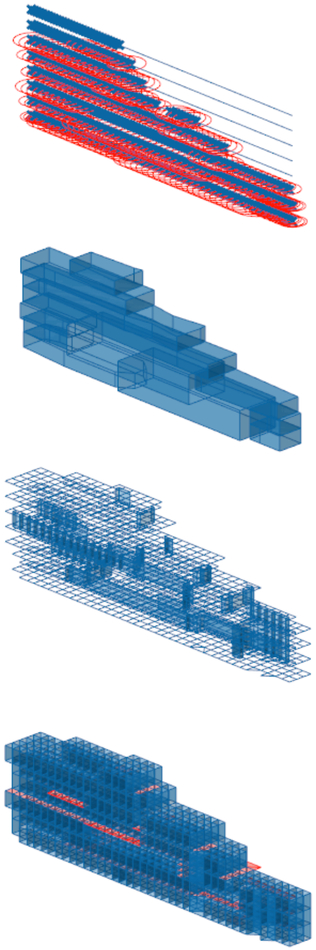
By loosening the adherence to the surrounding building heights, the second iteration spreads out and evenly distributes itself across the entire site. The circulation algorithm fares better here, organising the buildings circulation structure to take advantage of each level of the tiered roofs generated by the massing plan.

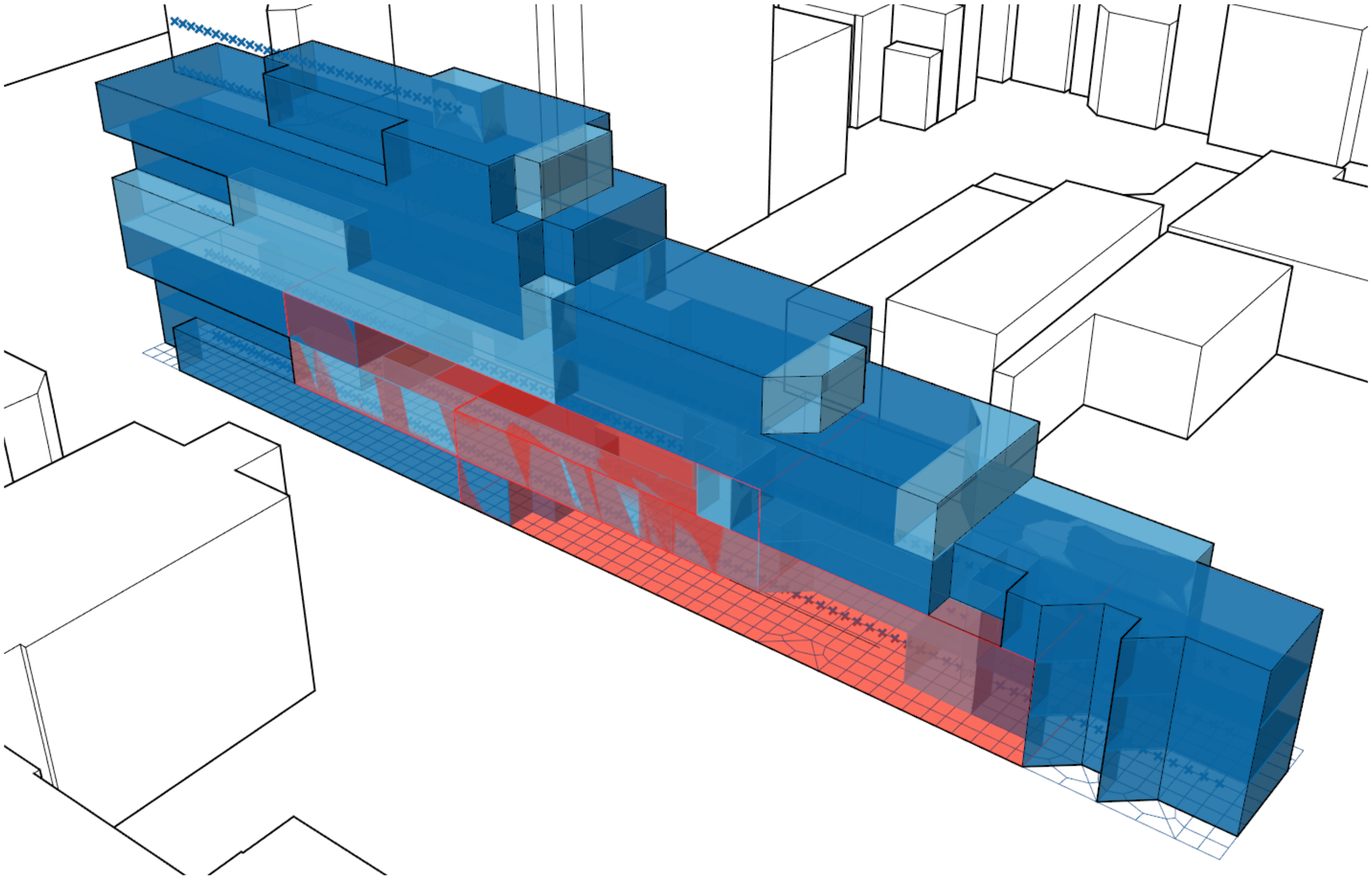
The extra space has provided room for both the gym and hall, but the two spaces overlap in a tiered configuration reminiscent of a lecture theatre. Additionally, this layout highlights the different permeability rules that can be applied to the circulation system – the circulation system passes through the middle floor of this triple-height space, linking the north and southern ends of the building.

As an aesthetic consideration, several triangular extrusions have been generated along the south-western edge of the building, creating an interesting pattern. By simply changing the setting in the grid algorithm to generate a triangular grid, this could be propagated throughout the form in seconds. This amount of reactivity promotes an environment of rapid and unfettered design exploration.

fig.49

ITERATION 2: SOUTHWEST PERSPECTIVE





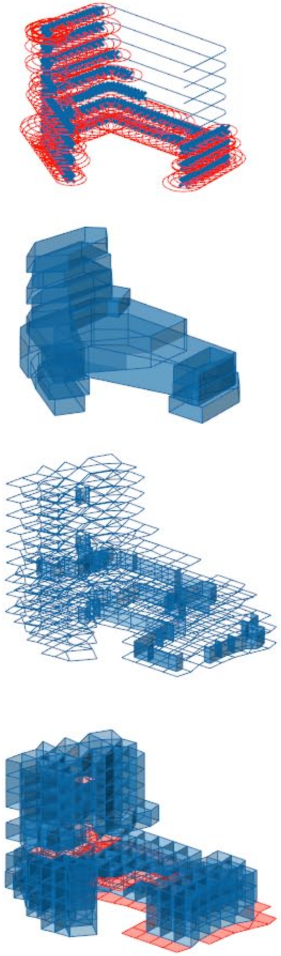
4.12 Jervois Quay Iteration 1

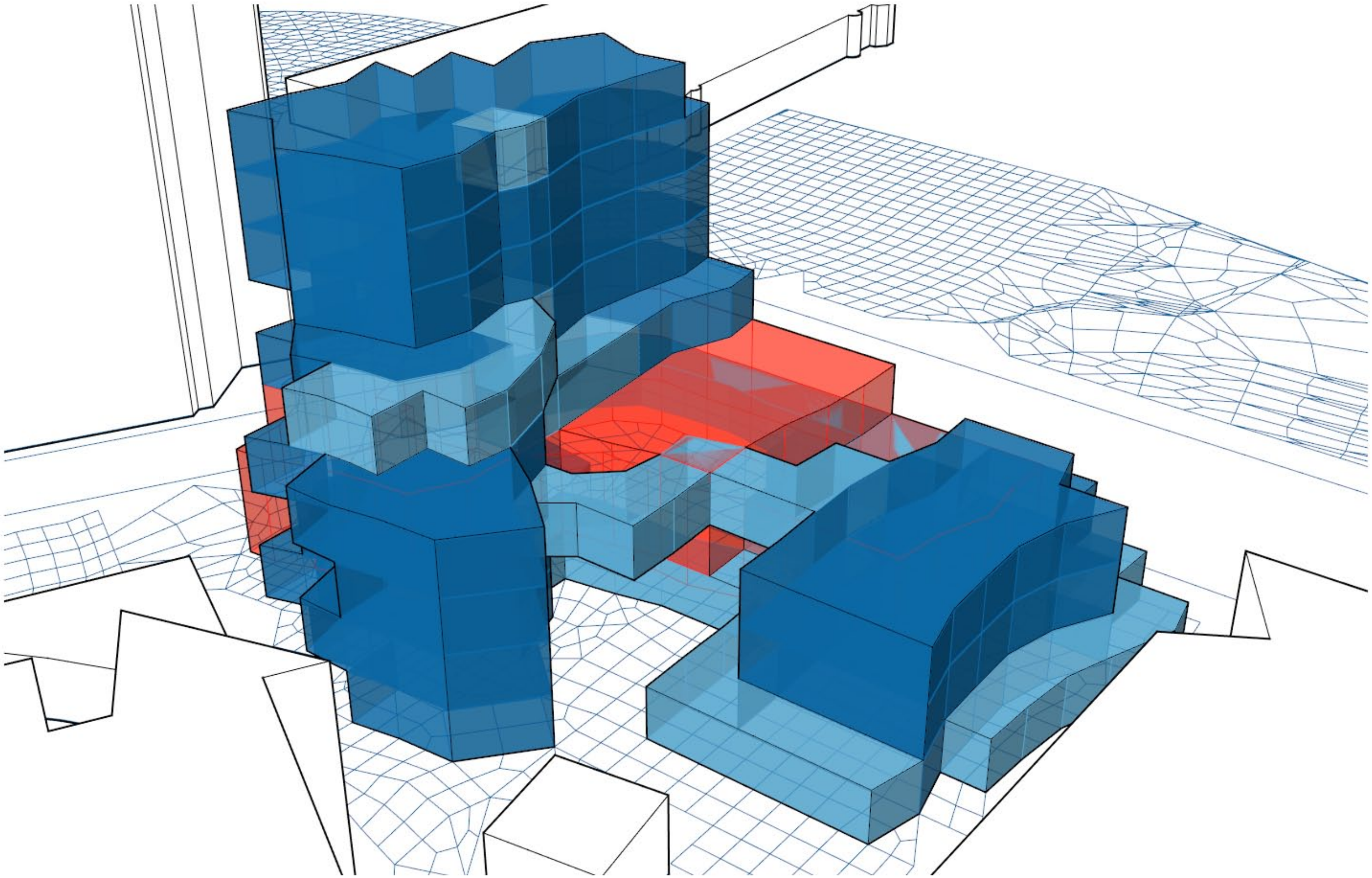
The urban context that surrounds Jervois Quay is highly irregular, revealing the influence of the surrounding building footprints on the underlying grid. The buildings send out “ripples” that interfere with one another to inform the relationship between any generated geometry and the surrounding context – a rectilinear grid produces rectilinear buildings, while more esoteric patterns cause the geometry to become less predictable.

Evidence of this can be seen in the southern block of this iteration, where the south façade pulls back from the neighbouring building while the rest of the block remains largely rectilinear. On the western edge of the building, the interaction between massing and grid has created a large wedge shaped block that wraps the form back in on itself, which in turn causes the circulation system to cantilever itself out over the central courtyard. The algorithm can also identify the large roof area provided by the hall and southern block as a possible outdoor circulation area, and ensures that the system provides easy access to it.

fig.50

ITERATION 1: SOUTHWEST PERSPECTIVE





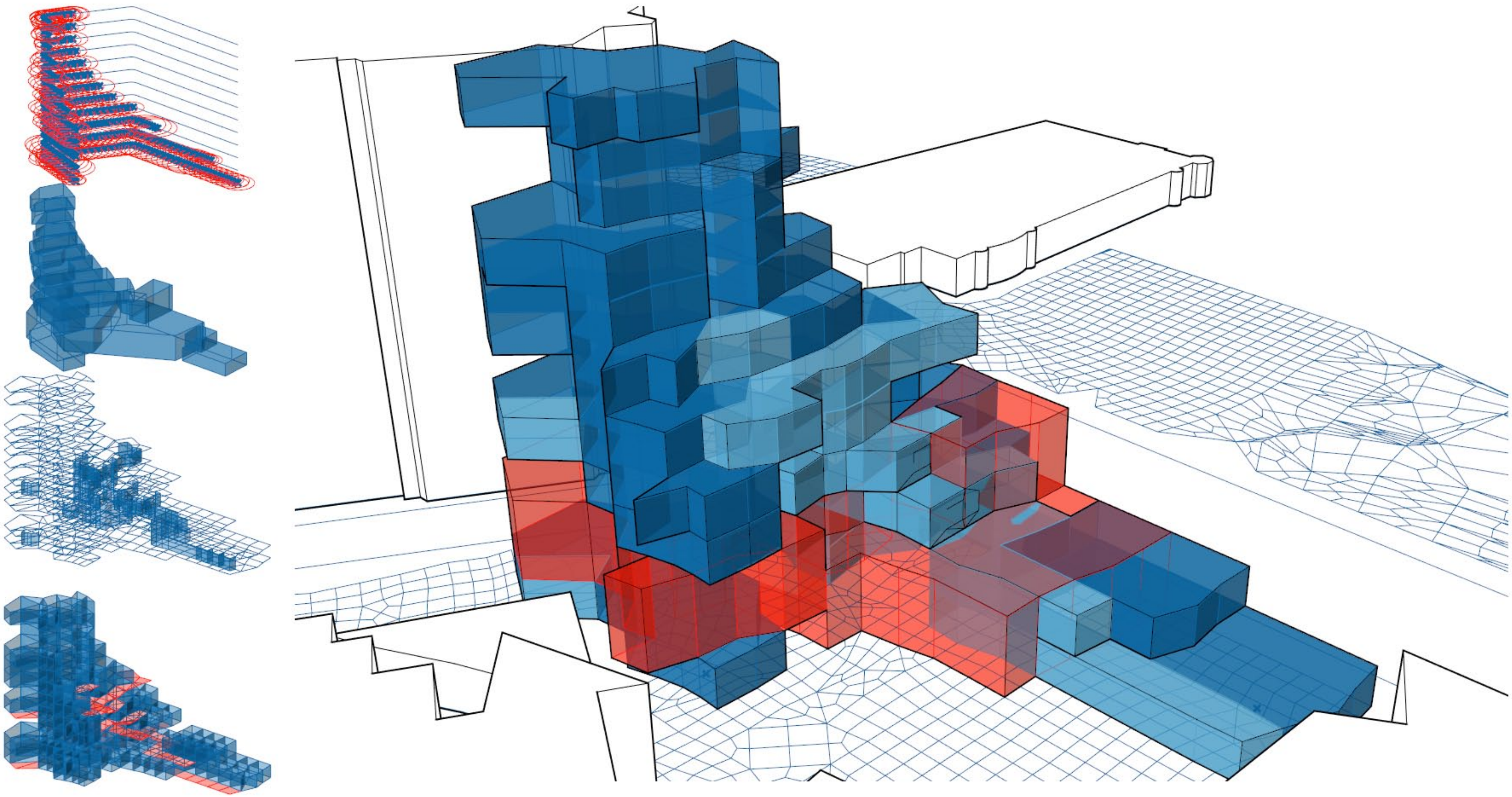
4.13 Jervois Quay Iteration 2

Where the first iteration engages with the various buildings that make up Civic Square, this iteration pulls back to the street edge and effectively disengages itself from the surrounding context. The many-tiered building is denser than other iterations, and the underlying grid creates a chaotic waterfall of form that tapers down to a single storey southern edge. The majority of the building is pressed up against the road edge, with the building rising to match the existing context at the northern tip of the site.

The juxtaposition between iteration 1 and 2 serves to show how important the input of a variety of data types is to the building generator. Without any contextual information the building resolves itself around only a few inputs, in this case the grid structure and a single building height reference.

fig.51

ITERATION 2: SOUTHWEST PERSPECTIVE



4.14 Critical Reflection

This final iteration of the algorithm successfully marries programme, typology and contextual information in order to generate different conceptual organisations of programme and form across different sites. More importantly, this experimentation has culminated in an overarching philosophy for the organising 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.

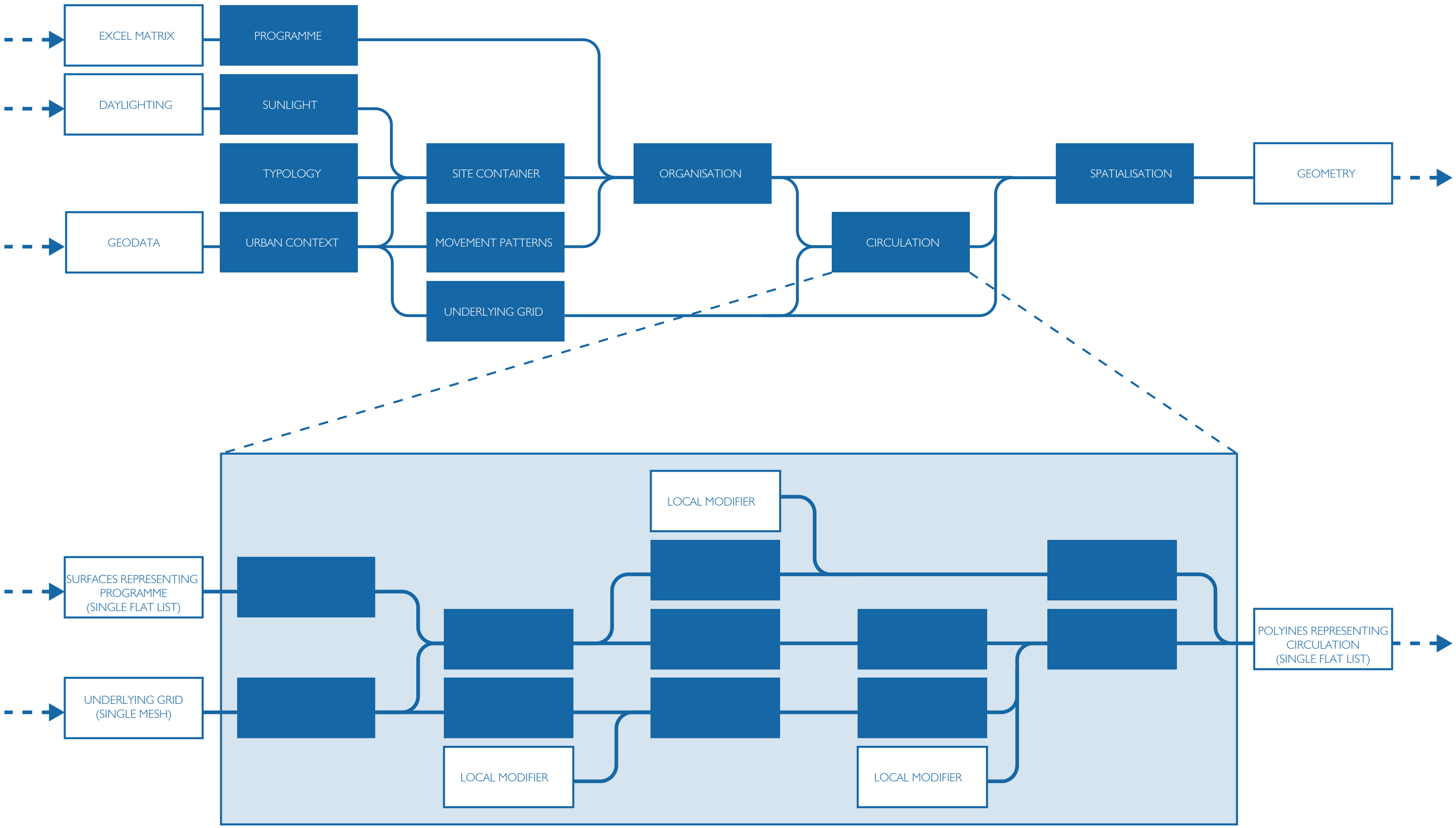
The DNA of the building is far from generic, and the algorithm could not be described as a ‘tool.’ The forms it produces are recognisably personal – teased over months to reflect my own design sense. The bold circulation lines and orthogonal shapes reminiscent of my undergraduate work are married with an additional complexity and exuberance to them that is initially hard to dissect – design decisions that contain a strange amalgamation of machine and human logic.

The approach that has been developed here, breaking the process down into discrete components and then carefully controlling the interactions, began simply as 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.

An unforeseen and important consequence of this new process is that the architect a much higher degree of control over the form than initially envisioned. 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 are maintained each can be edited individually as to the architect’s wishes, and the system responds to these changes in an understandable, though not predictable, way.

fig.52

"OPEN BOX" META-STRUCTURE



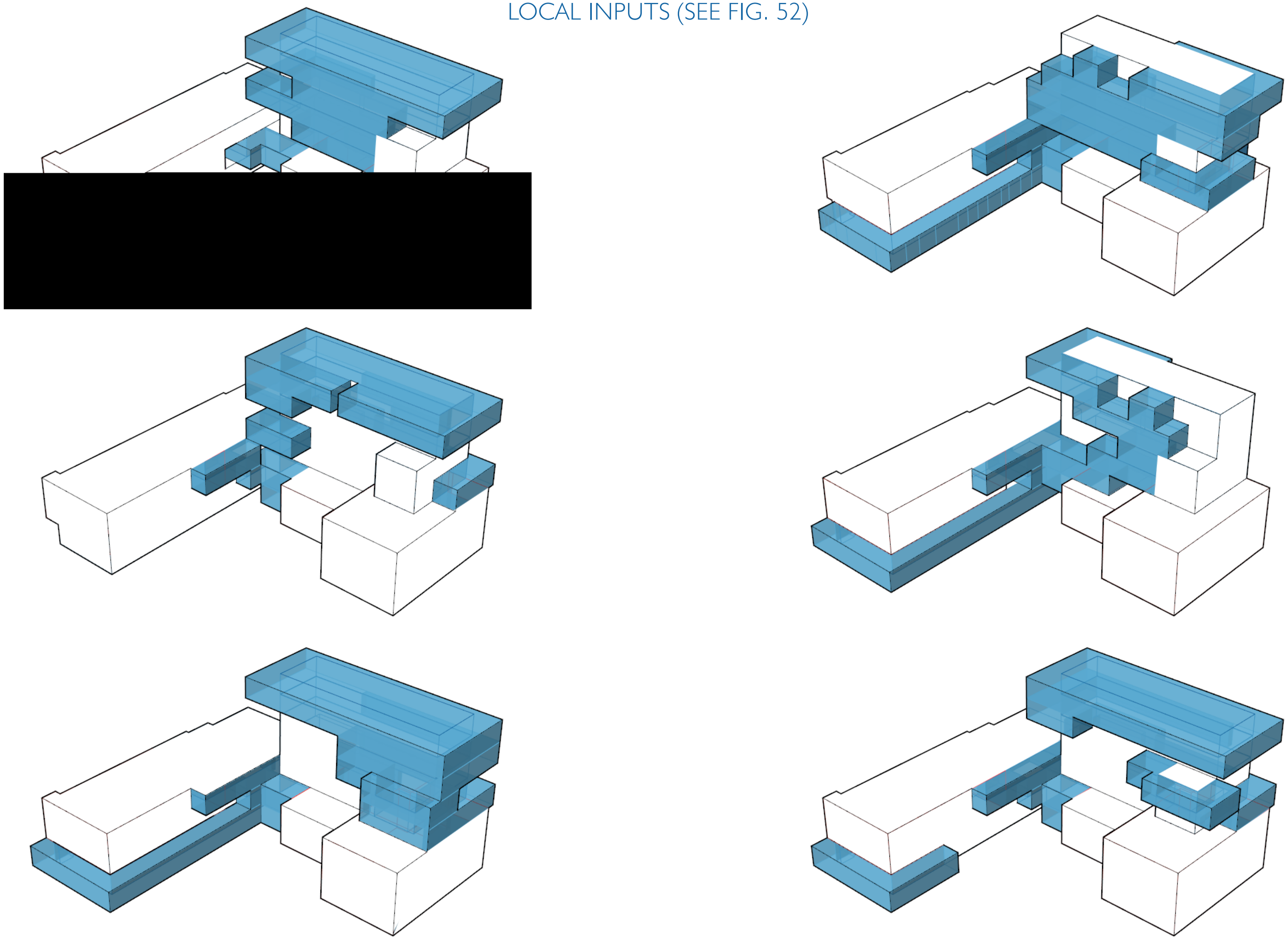
simplification of inputs/outputs allows large scale internal changes to be recognised globally by the system

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 (Figure 52), for example, can be endlessly modified, shifting the organisation of corridors independently from the programme (Figure 53). 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 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, personally, 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 your decisions generatively. The system of encoded relationships allows for rapid iteration, without denying the importance of the designer.

fig.53

GLOBAL CHANGES TO CIRCULATION, AS AFFECTED BY LOCAL INPUTS (SEE FIG. 52)



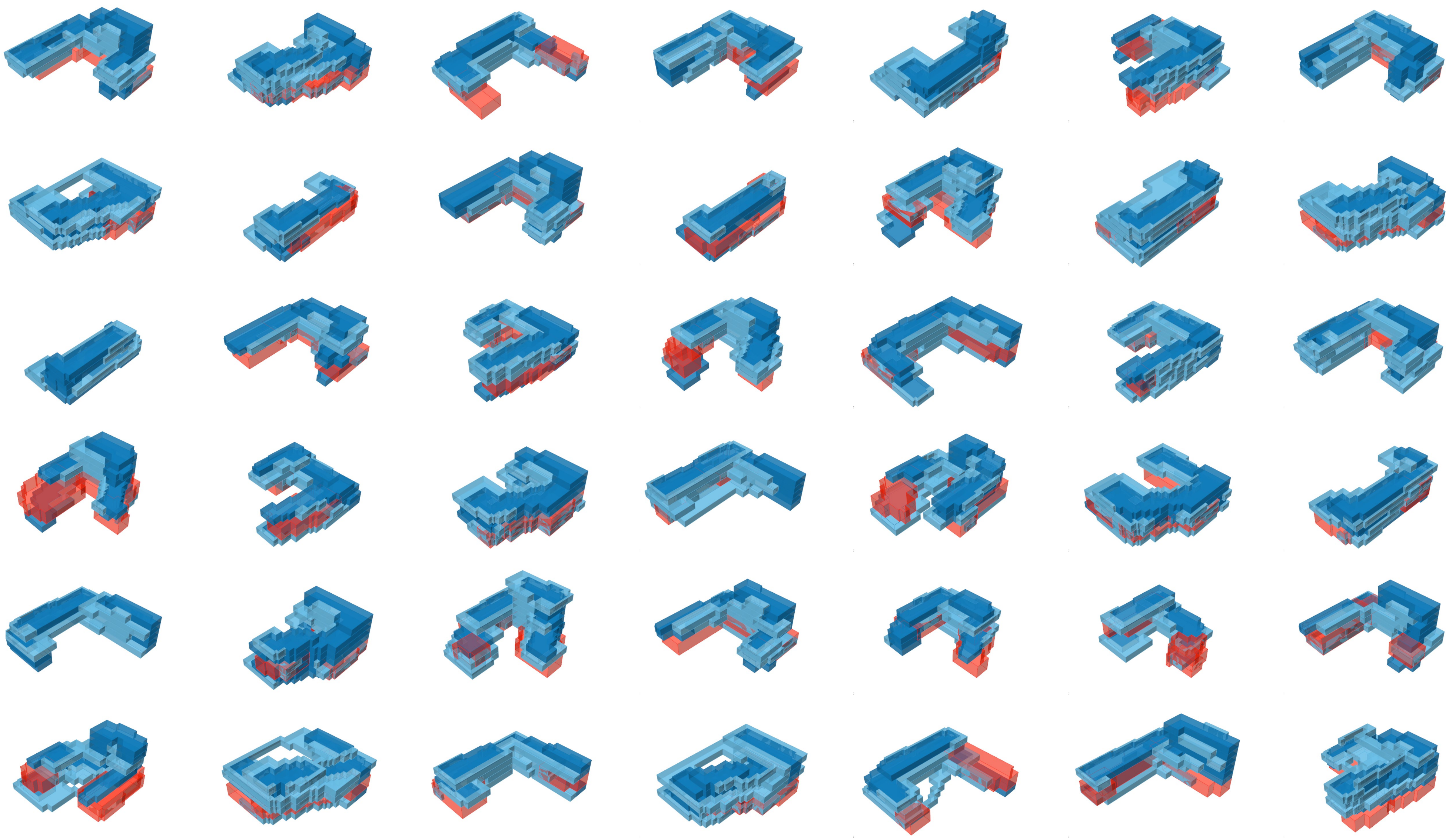
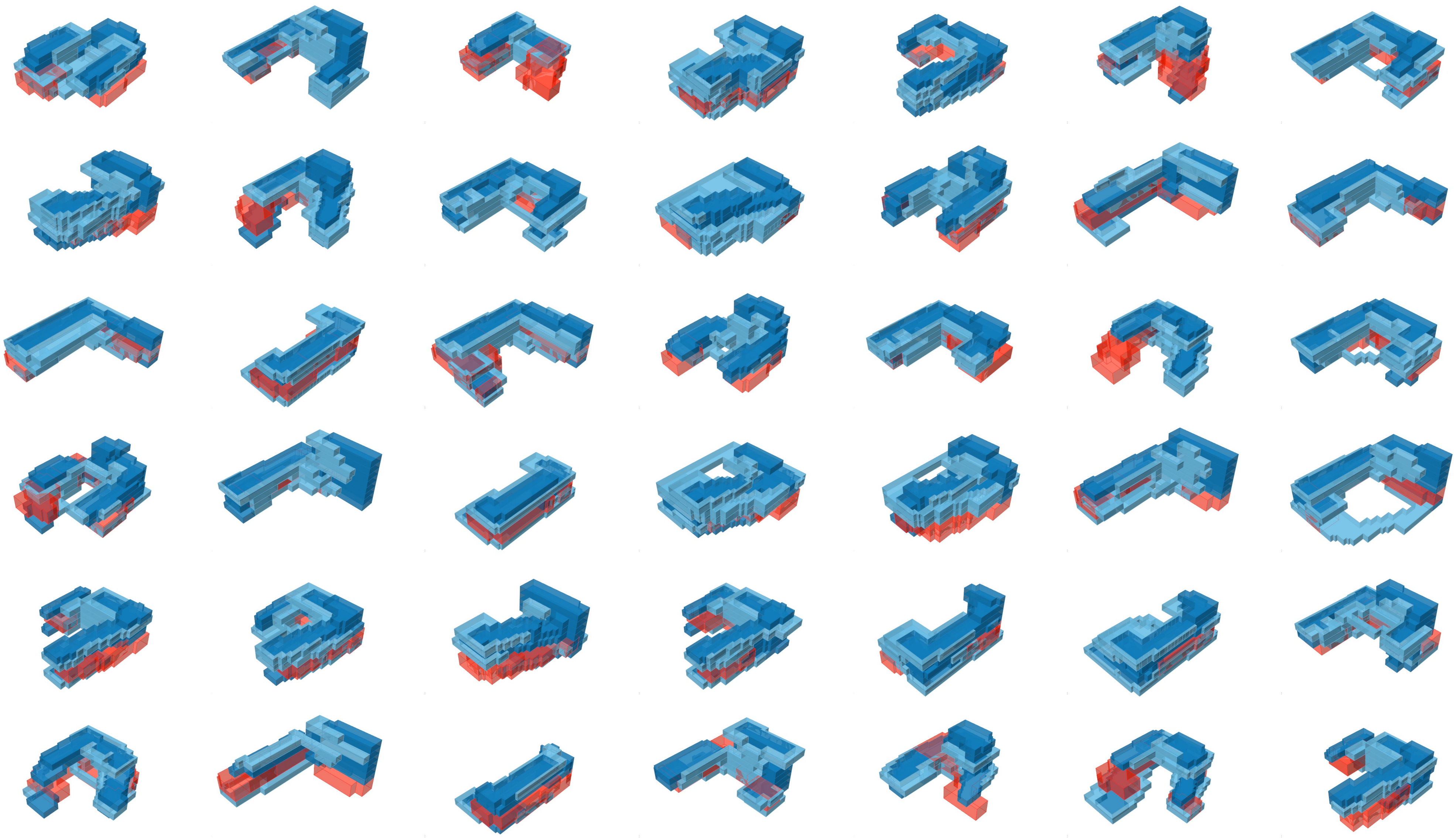


fig.54



5

Discussion and Conclusion

5.1 Geometry Bottleneck

The focus of this thesis was on the early part of the design process, creating a script that develops a spatially logical conceptual building that produces novel results. The final outcome of this is a framework that outputs a set of polylines, points and surfaces representing floor plates, walls and structure organised in multiple stories.

While this information can theoretically be taken through to a fully resolved building, a production bottleneck is revealed when attempting to describe the detailed geometry out of the abstract information. This is due to a number of factors, including limitations and inconsistencies of the grasshopper tool and the difficulty of creating parametric geometry that works consistently across any number of context-specific 'cases'. It is difficult and time consuming to parametrically quantify what would otherwise be intuitive, manual design decisions. When working digitally, what would otherwise be the flick of a pen or the twist of a module becomes a vast network of heavy coded instructions.

This represents both challenges and opportunities for the designer. Developing reactive, intelligent geometry can actually harm the design process, relying on generic cases and making it difficult to integrate spur-of-the-moment ideas into the development process. As a design strategy, considering the spatial information generated by the algorithm purely a sketch to be worked over mitigates the desire to produce fool-proof, endlessly complicated parametric components. Furthermore, it is a simple process to move the Grasshopper information directly into Revit or AutoCAD in real time, shifting the focus of the algorithm towards that of a three dimensional sketch.

Alternatively, with appropriate organisation and clever use of resources, the Grasshopper algorithm can be modified and augmented to focus in 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 environment.

5.2 A Marriage of Designer and Computer

Over the course of this project it has become increasingly apparent that an undirected computer will not produce innovative or interesting form if left to its own devices. It is purely a tool - an important and powerful one, but a tool nonetheless. The computer as a design tool relies on an informed designer controlling the flow of information between different parts of the machine, allowing different algorithms and calculations to inform each other in intelligent ways. Complex exploratory concepts can be iterated on in seconds, rather than through hours of sketching or manual drawing.

The "open box" model 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 that you cannot, fast, and augmenting these interactions with your own gestures. The design process becomes a discussion with your own work, iteration, tweaking, conversing with your computer partner, disagreeing, arguing, evolving, and building something richer because of it.

If considered as an organism, the "open box" produces the genotype of the building, the instructions that dictate the underlying organisation and structure of an organism. The work-flow and decision making process in developing the geometry of the building, the phenotype, is another process entirely. While the genotype will imply a certain aesthetic in it structure, it is the role of the designer to expose and repress elements of this aesthetic. This is a process that feeds back into the genome, the same process of cyclic design research that produced the genome in the first place.

5.3 The Importance of the "Open Box"

When as a designer you are presented with a series of closed options to tweak, there is a certain propensity to only use the options that are put in front of you. This can be seen inside the community surrounding Grasshopper itself, where a glut projects focusing on metaballs, voronoi cells and UV surfaces fight for attention. These "black box" tools represent highly structured, simple ways of transforming abstract data into specific geometric forms.

The designer cedes control for simplicity. The looser the structure, the more fidelity you can have in controlling and changing relationships, the more informed your design can be. Creating a framework as open and modular as possible allows for structure while still giving the user a huge amount of control, and allows for constant updating as technology improves.

Buildings are not closed systems. They exist in complex, fluctuating environments and must, in their design, respond to any number of inputs and limitations. Noisy complexity can be generated from a small set of simple rules, but informed complexity cannot. Informed complexity relies on the organisation and interpretation of a large amount of data, requiring more than a few token routines or quick experimentations using existing tools.

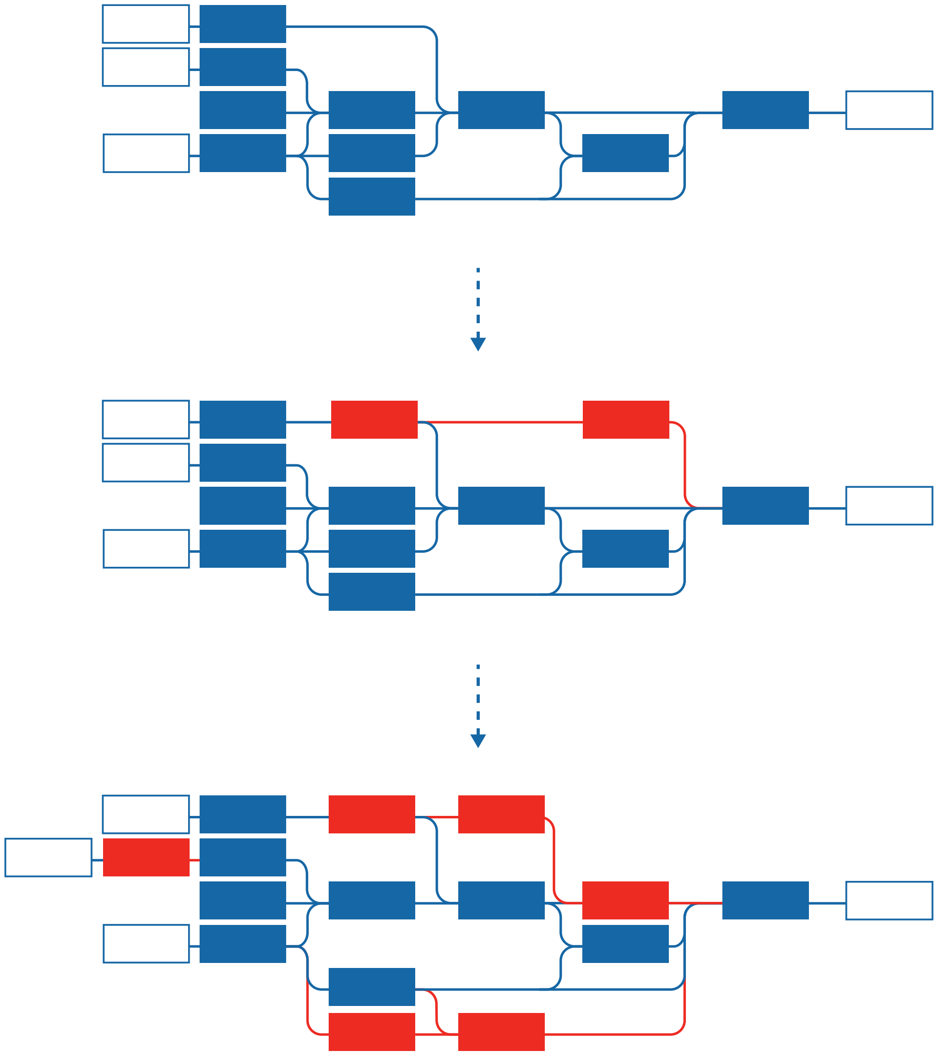
The goal of the "open box" structure is to augment the architect, revealing avenues and options so that the architect can make the best decisions and the best buildings possible. As an end user, and in a profession with many other areas to focus on besides the digital realm, there will not always be time to explore parametric design to its fullest extent. In fact, Grasshopper makes it far easier to mash up existing techniques rather than moving from first principals. By building a holistic parametric framework, an ever expanding model within which you work, you can react to changes and advances in the medium, as well as explore and expand your model from project to project. At the start of each project, rather than being forced to begin from scratch, a chance to explore some small corner of the digital work flow, the tool becomes an evolving ecosystem of thought. This creates an open, organic, holistic framework that evolves over time, incorporating new ideas and systems as they emerge into the public domain.

In the current age of open-source software, a technique can go from being practically impossible to being easily accessible in a matter of weeks. While there are vast wells of complex spatial and mathematical ideas being explored outside of Grasshopper3D, the integration of these tools into the platform is what makes them a viable option of exploration for the majority of end users. (Davis & Peters, Design Ecosystems: Customising the Architectural Design Environment With Software Plug-Ins, 2013) Tools like Karamba, Galapagos and Weaverbird highlight this trend, opening up design options that previously would have required a great deal of technical knowledge to implement into a project.

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 over- complicated 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.

fig.55

"OPEN BOX" STRUCTURE OVER TIME



5.4 Future Work

The "open box" organisation 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 Grasshopper3D 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 thesis. These plug-ins are explored extensively by Frano Bazalo, a fellow Victoria University Master's student, in his thesis "Responsive Algorithms: An investigation of computational process in early stage architectural design," which serves as a fantastic companion piece to this thesis. This total, this work represents a solid foundation for future exploration in the area of computer-aided space planning.

5.5 Conclusion

In response the research question of how the interaction of programme, typology and contextual information be used to generate conceptual massing in architecture, a complex "open box" design structure has been developed that produces responsive, novel conceptual designs in a marriage of designer input and computer processing.

The integration of parametric design tools into the architectural design process opens up many new avenues of exploration. By processing large amounts of information quickly and accurately, unique spatial conditions can be developed that would otherwise be impossible or highly time consuming. However, treating computers purely as a processor of geometric complexity limits the true potential as a design generator. By using these emerging tools to explore more complex architectural issues related to space planning and massing, the computer can truly engage and contribute to the architectural work-flow.

Complexity relies on simplicity. A complicated machine can fail – a complex machine is simple enough to respond to changes. (Perony, 2014) By keeping the interactions limited to very specific pathways, the interaction of programme, typology and contextual information be 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 behaviours 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. This process, described as an "open box" structure, produced a wide variety of working concepts and provided a high level of control as a designer.

It is an unusual thing, to look at for the first time on a design that clearly embodies your own design sense, without ever having a hand in the physical modelling of it. 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 power. 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.

6

Appendices and Bibliography

6.1 Bibliography

Aish, R. (2003). Extensible Computational Design Tools for Exploratory Architecture. In B. Kolarevic, Architecture in the Digital Age - Design and Manufacturing (pp. 243-253). New York: Spon Press.

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

Bjarke Ingels Group. (2010). Yes is More. Koln: Evergreen.
l

Burry, M. (2003). Between Intuition and Process: Parametric Design and Rapid Prototyping. In B. Kolarevic, Architecture in the Digital Age - Design and Manufacturing (pp. 147-163). New York: Spon Press.

Burry, M. (2011). Scripting Cultures. Chichester: John Wiley & Sons.

Davis, D. (2013, September 20). Chapter 2 - The Challenges of Parametric Modelling. Retrieved from Daniel Davis: http://www.danieldavis.com/thesis-ch2/

Davis, D., & Peters, B. (2013, March). Design Ecosystems: Customising the Architectural Design Environment With Software Plug-Ins. Architectural Design, pp. 124-132.

Downton, P. (2003). Design Research. Melbourne: RMIT Publishing.

Goulthorpe, M. (2003). Scott Points: Exploring Principals of Digital Creativity. In B. Kolarevic, Architecture in the Digital Age - Design and Manufacturing (pp. 163-181). New York: Spon Press.

Johnson, N. (2009). Chapter 1: Two's company, three is complexity. In N. Johnson, Simply complexity: A clear guide to complexity theory (pp. 3-18). Oxford: Oneworld Publications.

Jurgen, L., Wagner, D., & Zweig, K. (2009). Algorithmics of Large and Complex Networks. Berlin: Springer-Verlag.

Lazzeroni, C., Bohnacker, H., Laub, J., & Grob, B. (2009). Generative Design: Visualise, Program and Create with Processing. New York: Princeton Architectural Press.

Meredith, M. (2008). Never Enough (transform, repeat ad nausea). In T. Sakamoto, & A. Ferré, From Control To Design: Parametric/Algorithmic Architecture (pp. 6-10). Actar Birkhauser Distribution.

Merrell, P., Schkufza, E., & Koltun, V. (2010). Computer-Generated Residential Building Layouts. Stanford: Stanford University.

Oosterhuis, K. (2002). Architecture Goes Wild. In K. Oosterhuis, Computers are the New Extension of our Bodies (pp. 144-153). Rotterdam: 010 Publishers.

Ottchen, C. (2009, March). The Future of Information Modelling and the End of Theory. Architectural Design, pp. 23-27.

Oxford Dictionaries. (2013). Black Box. Retrieved from Oxford Dictionaries: oxforddictionaries.com/definition/english/black-box

Oxford Dictionaries. (2013). Bottom Up. Retrieved from Oxford Dictionaries: oxforddictionaries.com/definition/english/black-box

Oxford Dictionaries. (2013). Top Down. Retrieved from Oxford Dictionaries: http://www.oxforddictionaries.com/definition/english/black-box

Oxman, R., & Oxman, R. (2010, July). The New Structuralism: Design, Engineering and Architectural Technologies. Architectural Design, pp. 15-21.

Perony, N. (2014, January). Puppies! Now that I've got your attention, complexity theory. Retrieved from TED: ted.com/talks/nicolas_perony_puppies_now_that_i_ve_got_your_attention_complexity_theory.html

Piker, D. (2013, March). Kangaroo: Form Finding with Computational Physics. Architectural Design, pp. 136-137.

Price-Ramus, J. (2006, July). Behind the design of Seattle's library. Retrieved from TED: ted.com/talks/joshua_prince_ramus_on_seattle_s_library.html

Rahim, A. (2003). Designing and Manufacturing Preformative Architecture. In B. Kolarevic, Architecture in the Digital Age - Design and Manufacturing (pp. 199-217). New York: Spon Press.

Romero, F., & Ramos, A. (2013, March). Bridging a Culture: The Design of Muso Soumaya. Architectural Design, pp. 66-69.

Roudavski, S. (2009). Towards Morphogenesis in Architecture. International Journal of Architectural Computing, 7(3), 345-374.

Rutten, D. (2013, March). Galapagos: On the logic and limitations of generic solvers. Architectural Design, pp. 132-135.

Schneider, S., & Koenig, R. (2012). Hierarchical structuring of layout problems in an interactice evolutionary layout system. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 129-142.

Schumacher, P. (2009, July). Parametricism - A New Global Style for Architecture and Urban Design. Architectural Design, pp. 14-23.

Stehling, F., & Stehling, H. (2011, July). Lost in Parameter Space? Architectural Design, pp. 70-79.

Umemoto, N., & Reiser, J. (2006). Fineness and the Macroscale. In N. Umemoto, & J. Reiser, Atlas of Novel Tectonics (pp. 122-123). New York: Princeton Architectural Press.

6.2 List of Figures

6.3 List of Illustrations

Figure 23. Recursive shortest walk analysis of internal pathways using the natural pathways created by spatial boundaries and then creating volumetric corridors using circular extrusions.

6.3 Appendix A: Preliminary Design B-Sides

Much of the preliminary design phase of this project was spent searching for a method of quantifying transforming abstract information into 3D geometric space. This was a long process of experimentation and iteration, and a large amount of content was not directly relevant to the thesis. This section highlights a number of these studies in order to provide an insight into the breadth of the research undertaken.

fig.56

SHORTEST WALK CORRIDOR SPANNING TESTS

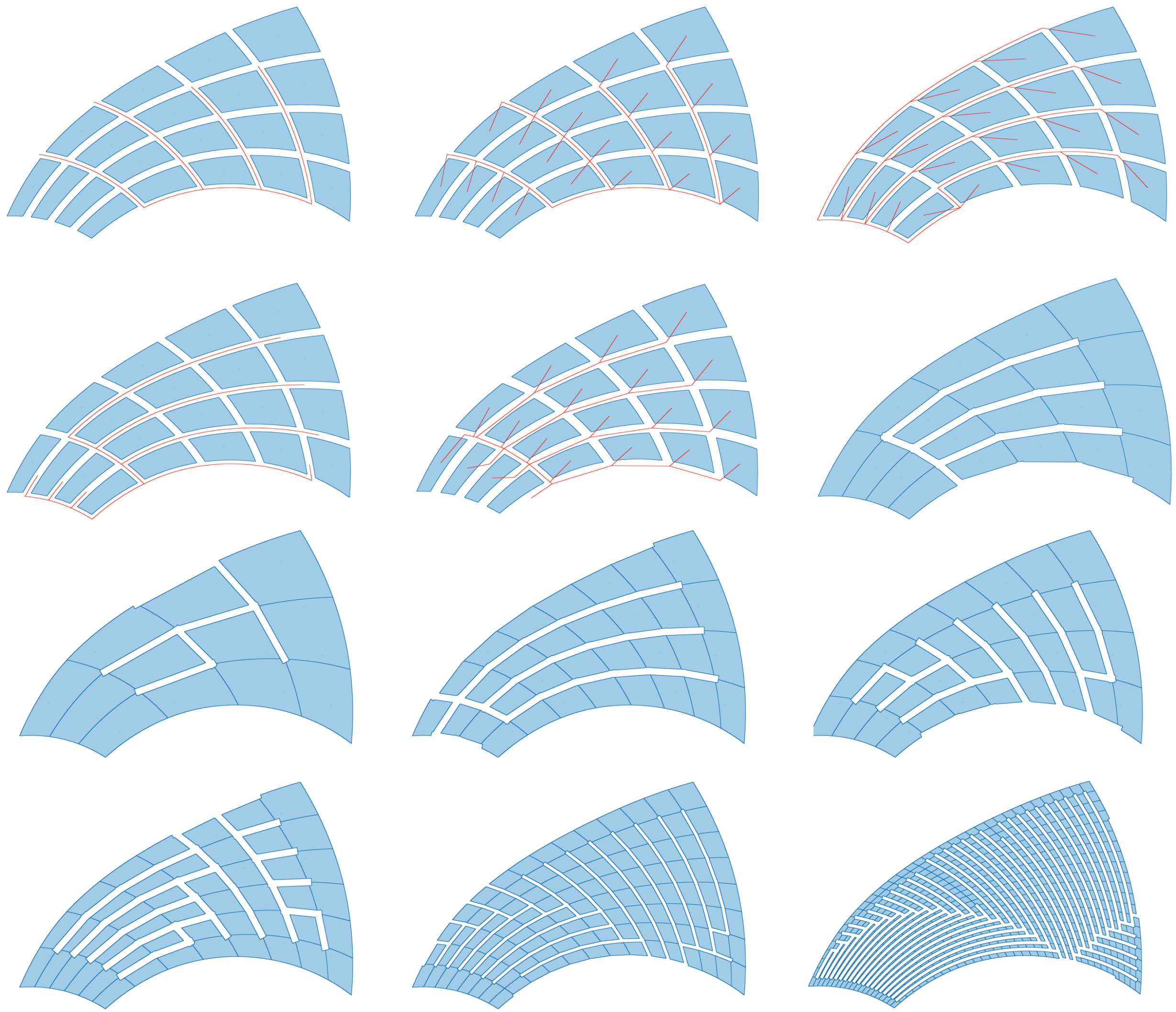


fig.57

GOLAPOGOS DRIVEN GRID-OVERLAY TEST RESULTS

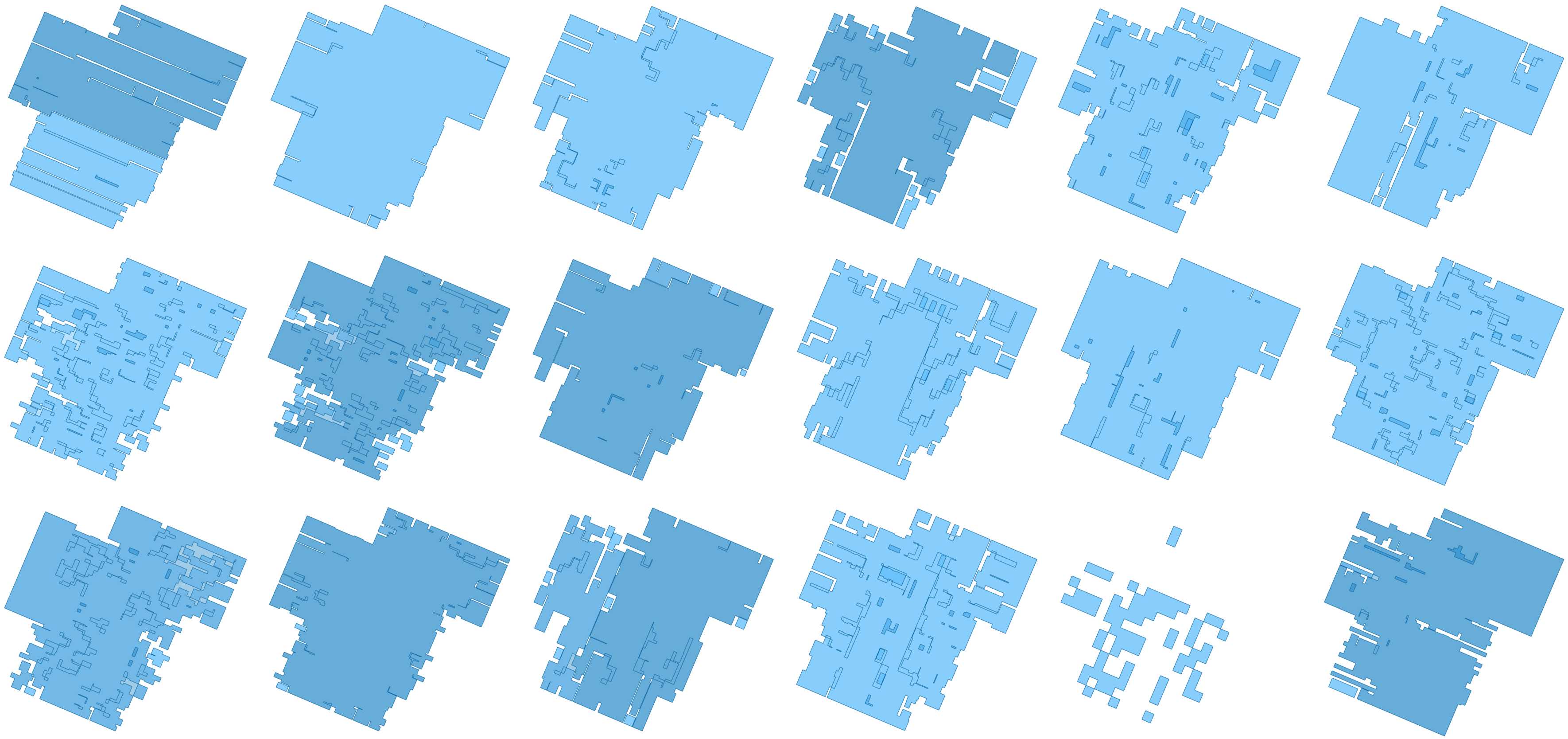


fig.58

'MINIMUM SPANNING TREE' OF ARBITRARY MESH

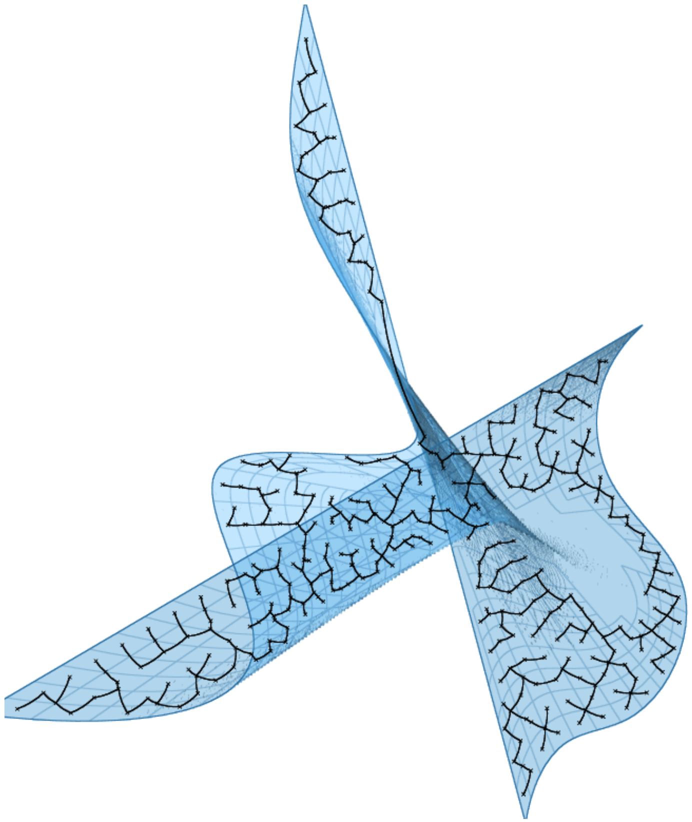


fig.59

POLYLINE CULLING BASED ON ORIENTATION/EDGE EXPOSURE

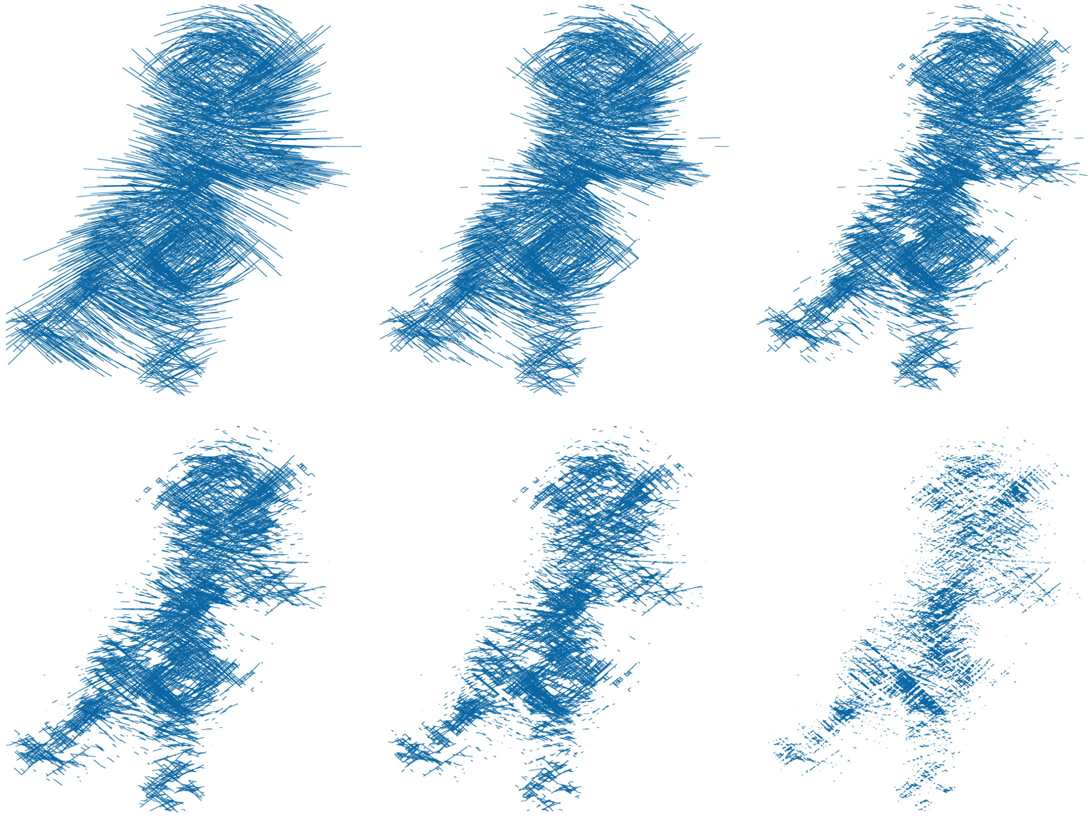
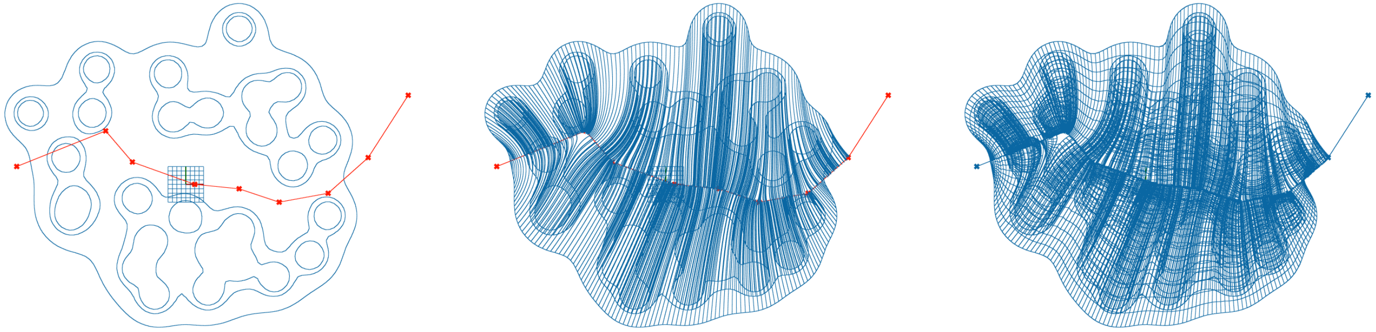
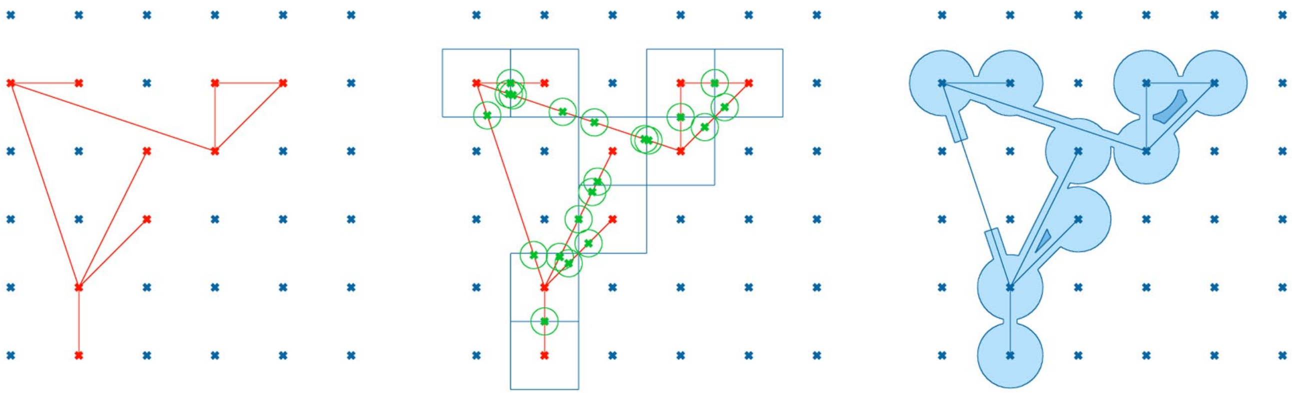


fig.60

UNDERLYING GRIDS GENERATED FROM WEIGHTED ZONES



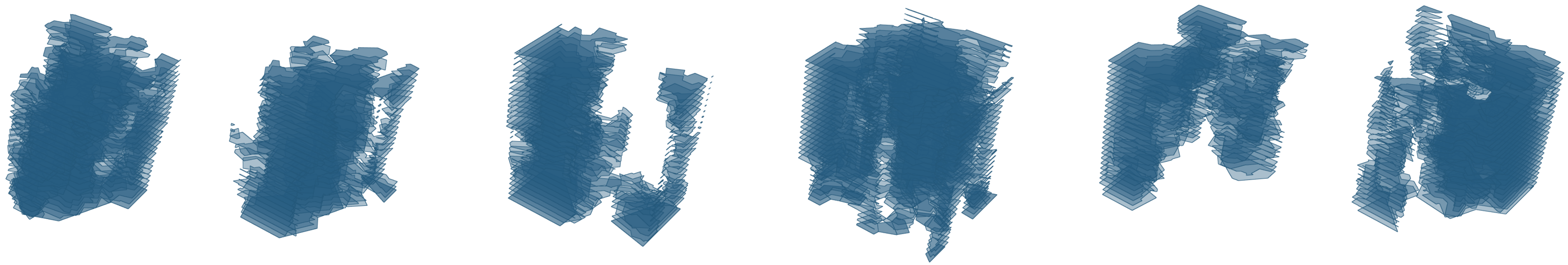


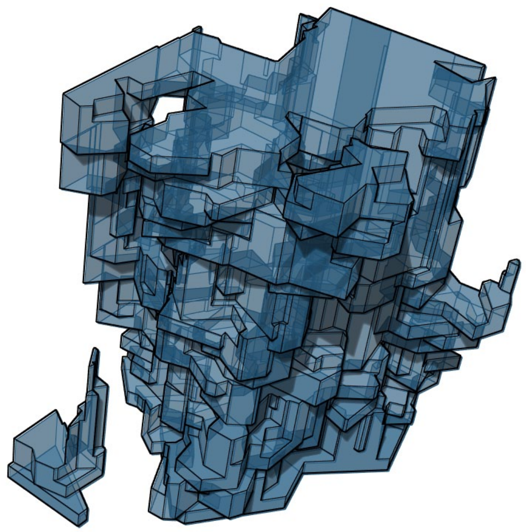
NODE BASED ENVELOPE BORDER IDENTIFICATION

fig.61

fig.62

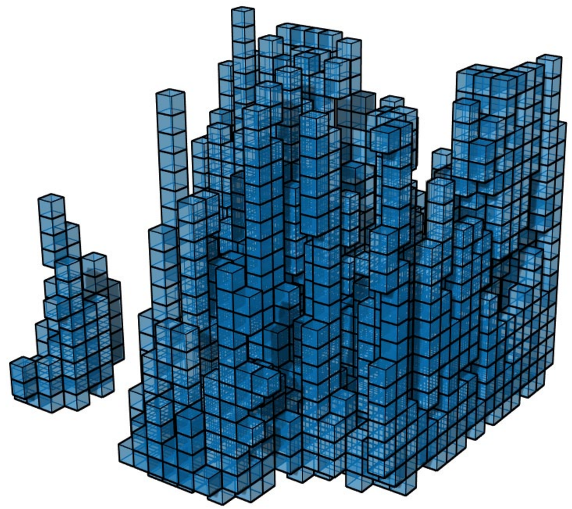
RECURSIVE FLOOR GENERATOR AVOIDING SIGHTLINES AND FOOT TRAFFIC





VOLUMETRIC MULTISTOREY GENERATED FLOORPLAN

fig.63



VOXEL-BASED REDUCTIVE FLOOR GENERATOR

fig.64

6.5 Appendix B: Developed Design B-Sides

The decision to shift the focus of the thesis away from producing a resolved building was made quite late in the design process. After a series of iterations were produced, it was determined that the geometry generation process was limiting the algorithm as a conceptual generator – these reasons are outlined comprehensively in chapter 5.1. This section explores the various conceptual buildings developed over the course of the project.

fig.65

ARBITRARY GRID BASED GEOMETRY GENERATION

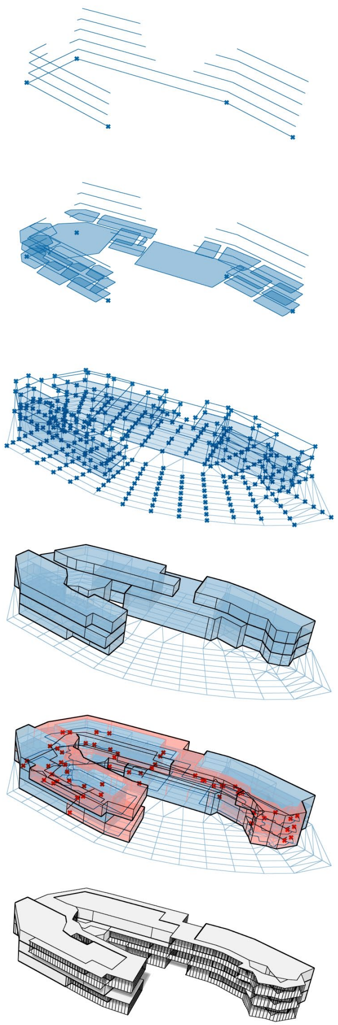


fig.66

UNIFIED EXTERNAL SKIN USING SMOOTHING TECHNIQUES

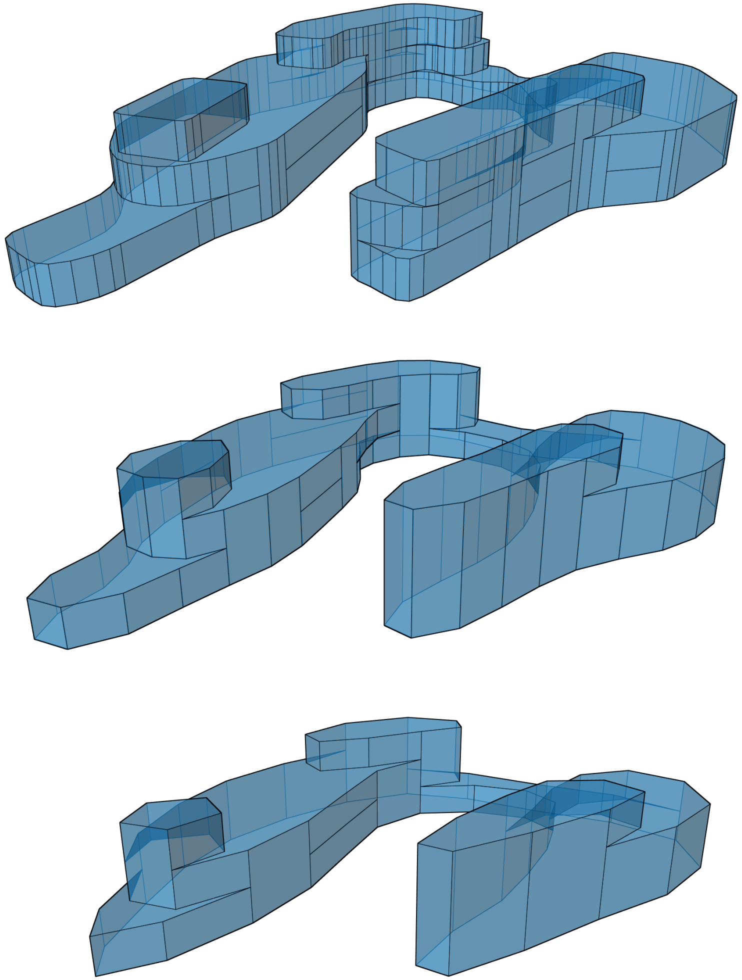


fig.67

GRID BASED GENERTION UTILISING SMOOTHING

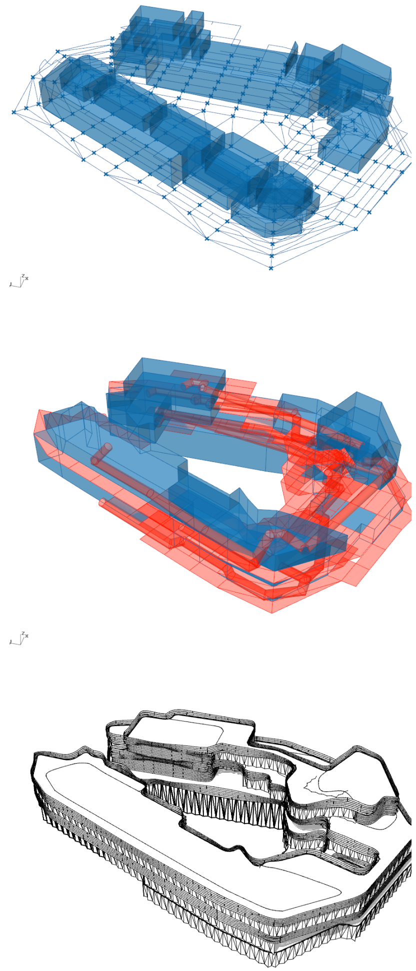


fig.68

EARLY CLAY RENDER TESTS

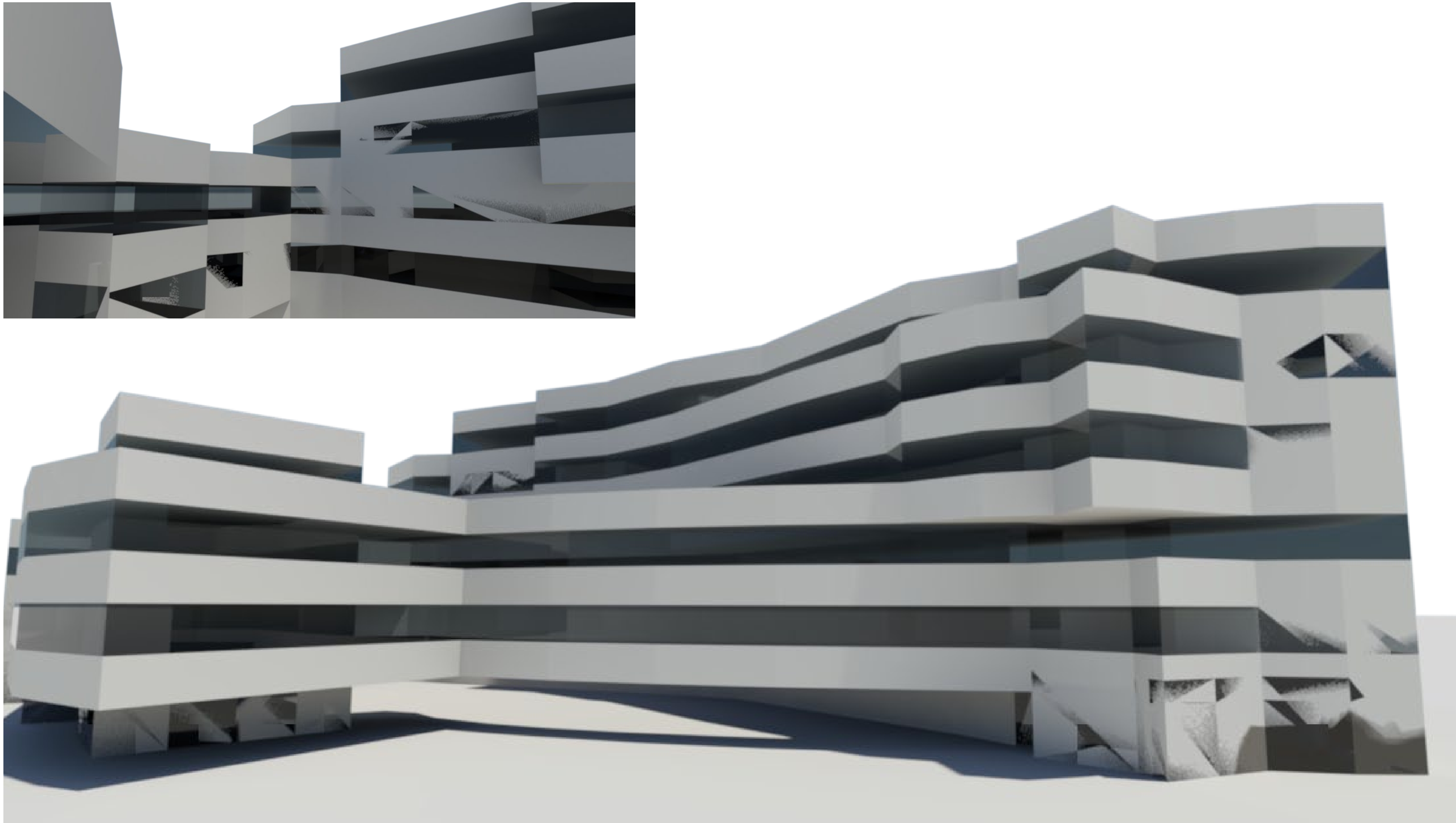


fig.69

DEVELOPMENT OF EXTERIOR CLADDING SYSTEM

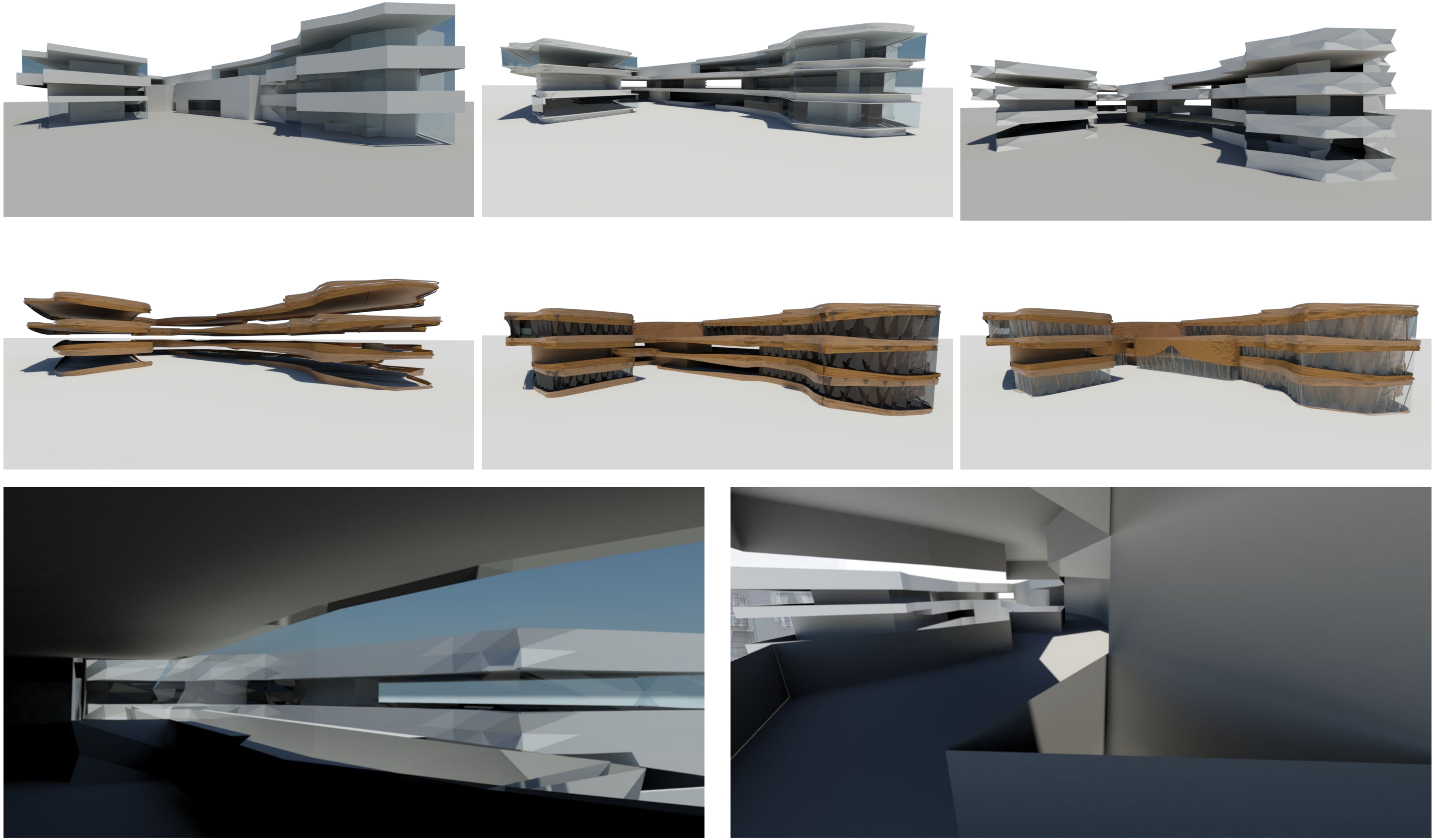


fig.70

SHIFT TO REGULAR CONTEXT-DRIVEN BUILDING GRID

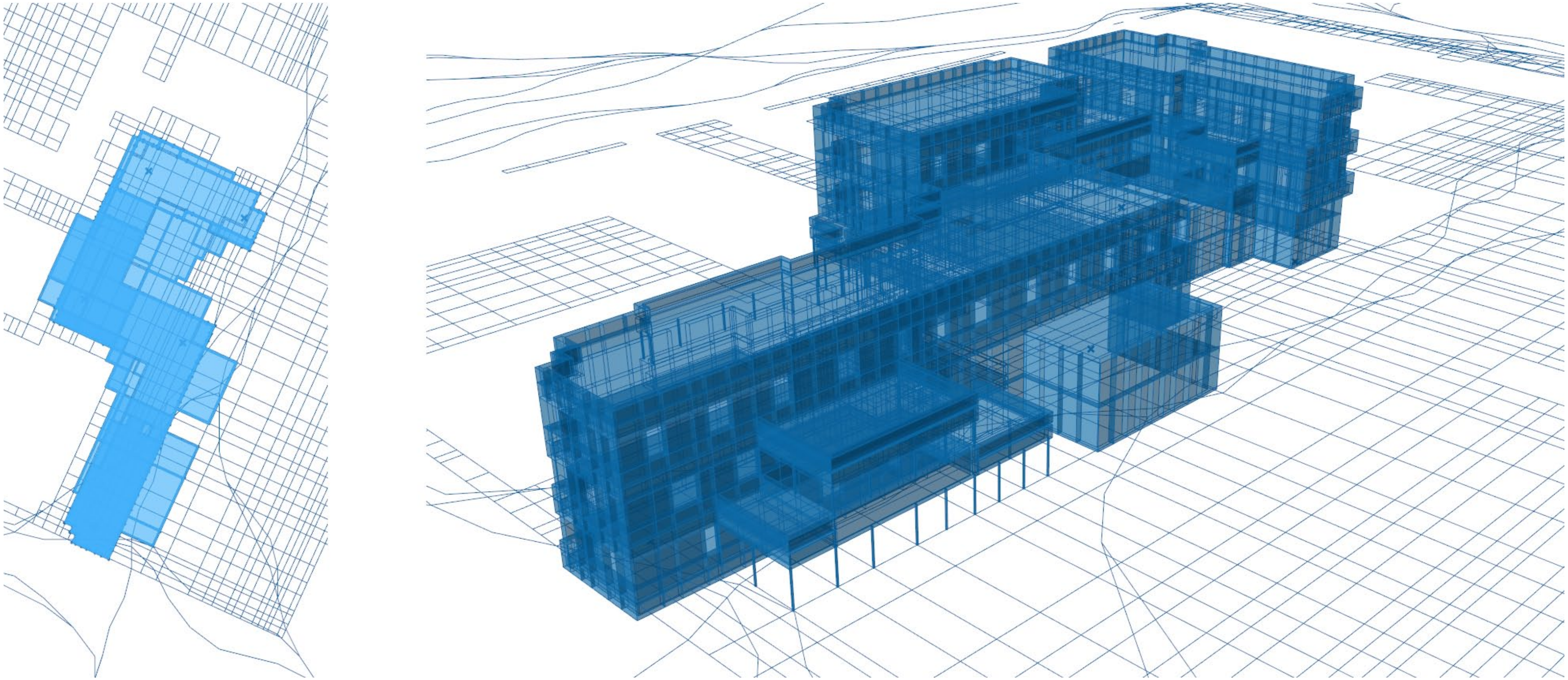
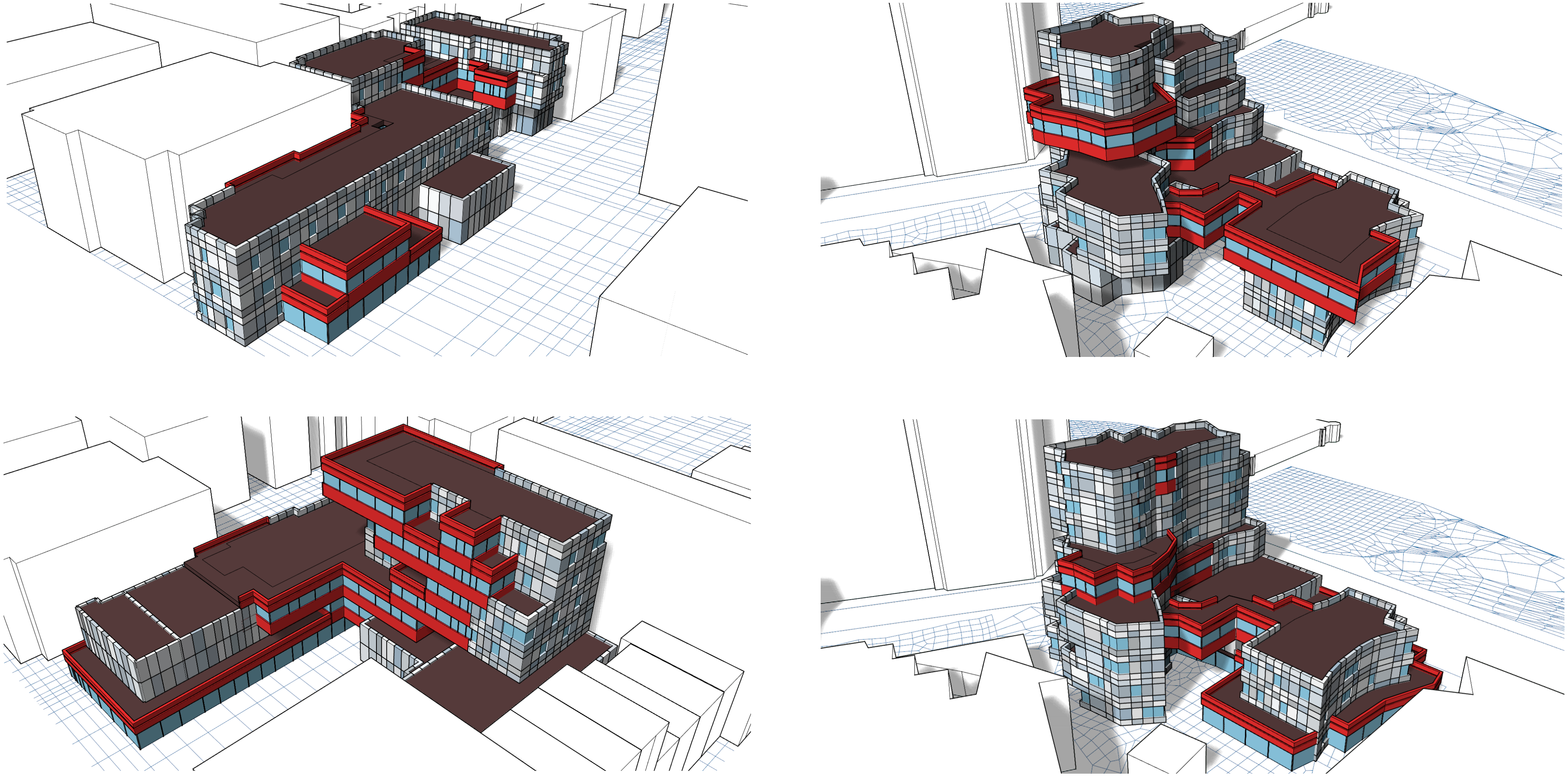


fig.71

DISPATCHING WALL DATA FOR MULTIPLE SURFACE TREATMENTS



end