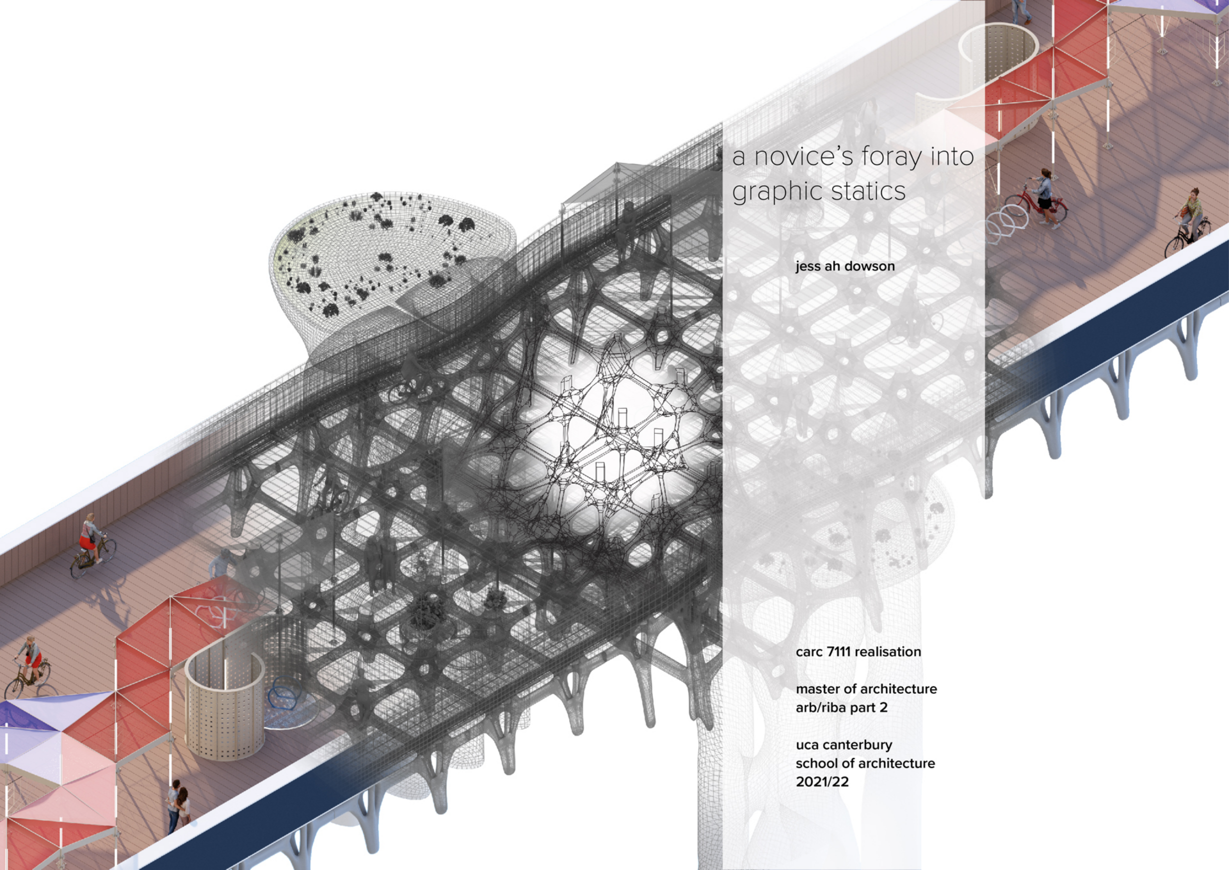


a novice’s foray into graphic statics

jess ah dowson

carc 7111 realisation

master of architecture
arb/riba part 2

uca canterbury
school of architecture
2021/22

contents

a novice's foray into graphic statics

jess ah dowson

carc 7111 realisation

master of architecture
arb/riba part 2

uca canterbury
school of architecture
2021/22

thesis overview

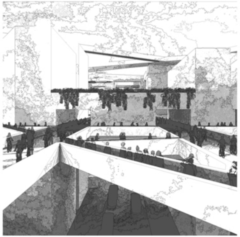
The research from the first semester of the thesis project pointed me in the direction of encouraging sustainable commutes and transportation across the Thames, with the railway stations of Erith, Dartford, Rainham and Dagenham East providing natural anchor points from which to generate a pedestrian & cyclists' network. The summary is below:

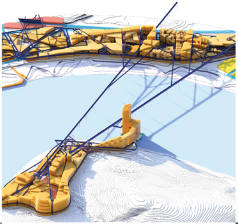
Working on the principle of seeking to provide a dedicated bicycle crossing to link up both sides of the river, with the intent to also reduce commute times, reliance on the QEII toll crossing and alleviate congestion caused by personal cars, the proposal is for a cycle superhighway crossing that would link Dartford with Rainham, in addition to Erith and Dagenham East. This would provide ample connections to the major industrial estates on both sides of the river located in this site, along with access to the District, Hammersmith & City Lines, and local branch lines for c2c, Southeastern and Thameslink towards Essex & Kent.

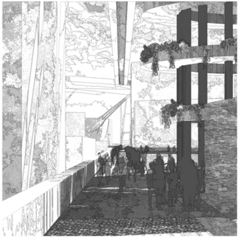
The second aspect would be to provide additional dedicated bicycle repair facilities, both manned and unmanned, along with workshops to allow for the education of any cyclist in the repair and maintenance of their bicycle. According to openstreetmap (OSM) data, there are only two bicycle shops within this 20km stretch, of which only one offers servicing facilities. Hence the need to improve this aspect of the infrastructure.

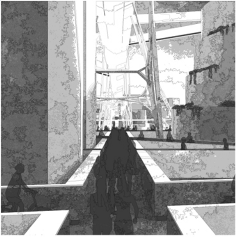
The third aspect would be to continue the spirit of the East London Green Grid project's desire to link up the Dartford, Crayford, Rainham + Wennington Marshes, whilst providing increased mobility beyond their reach. Thus providing relatively fast, direct access to much needed riverside greenspace from the urban centres of Dartford, Erith, Rainham & Dagenham.

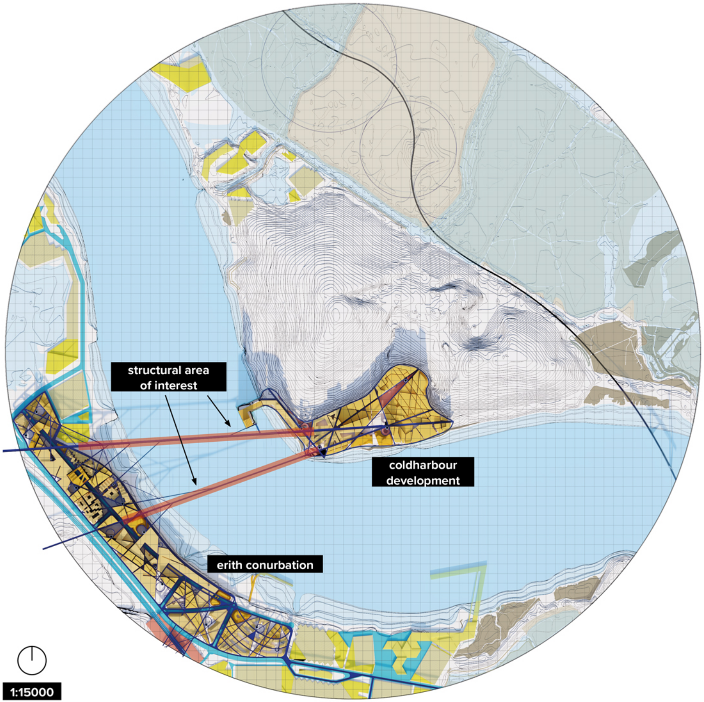
Critical roadblocks to any potential infrastructural proposal or development have been highlighted in red. Each of these criteria trigger an automatic consultation requirement with Natural England to understand and mitigate the impacts of any proposed development to nearby SSSIs. In this case, the Rainham Marshes is the closest, critical SSSI of interest.











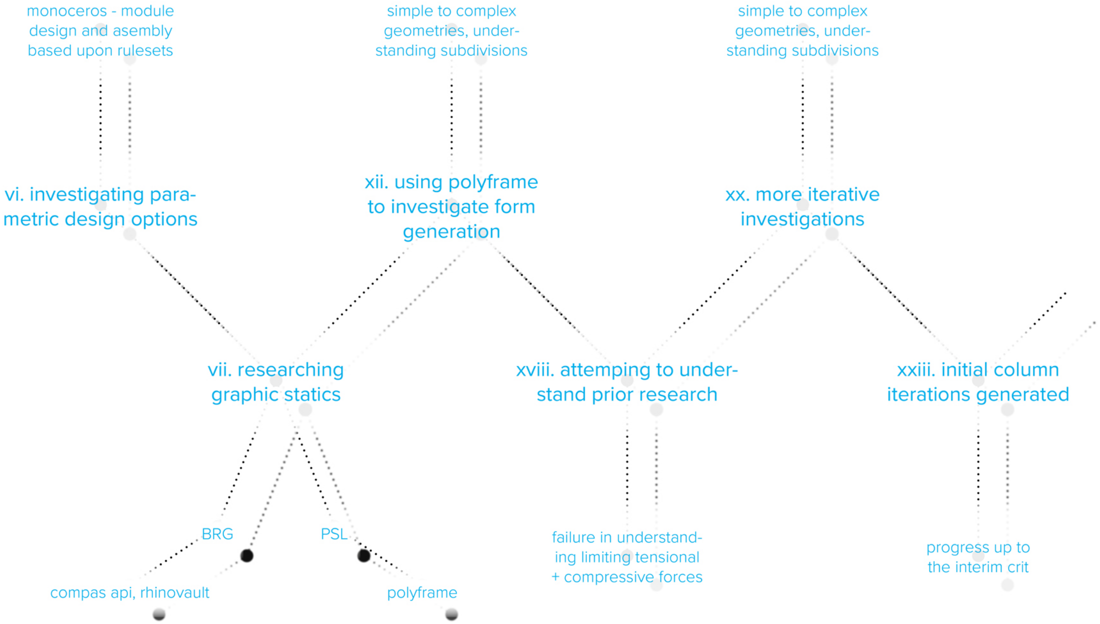
To that end, the proposal would have to be designed and implemented in such a fashion as to limit the amount of air pollution during construction phase, using materials that limited the amount of air pollution through dust generation or otherwise during usage and maintenance. This is in addition to limiting the footprint of the support structure along the proposed route as any route passes over sensitive natural environment.

Building upon the first semester's research, I anchored my second semester's work on a couple of assumptions: that the existing flood defences along the Thames may not be enough to mitigate the damage caused by a 1.0 to 2.0m rise in sea level over the next century, and a policy shift in the future would allow for the flooding of land as a means of natural flood defence leading to a significant change in the zoning development of the landscape.

To this end, I have decided to imagine a hypothetical future where the existing district of Erith has been densified, as well as bringing into existence a new brownfield development on the old Coldharbour Industrial Estate, at the foot of current landfill site. These two districts would be joined by a pedestrian and bicycle bridge, and the two districts would exist where pedestrian and bicycle traffic has been prioritised over vehicular - especially with the assumption that the majority of vehicular traffic will be automated in the future.

The aspect of the thesis project which I am focusing on within this Realisation module is the structure of the bridge itself, including deck span and structural columns/pylons as appropriate. This is where I hope the research into 3D Graphic Statics will provide both a suitable architectural and structural resolution to the thesis. I have provided concept images from the interim critical review along with an indicative masterplan, where I have highlighted the structural area of concern.

investigative timeline



monoceros - module design and asembly based upon rulesets

vi. investigating parametric design options

vii. researching graphic statics

BRG

compas api, rhinovault

PSL

polyframe

simple to complex geometries, understanding subdivisions

xii. using polyframe to investigate form generation

xviii. attemping to understand prior research

failure in understanding limiting tensional + compressive forces

simple to complex geometries, understanding subdivisions

xx. more iterative investigations

xxiii. initial column iterations generated

progress up to the interim crit

piping, dendro meshing, subd modelling + customised nodes/ cross-sections

xxvi. towards a physical output: meshing the geometry

xxvi. resin prints of column investigations

iterative development from derived understanding of first principals

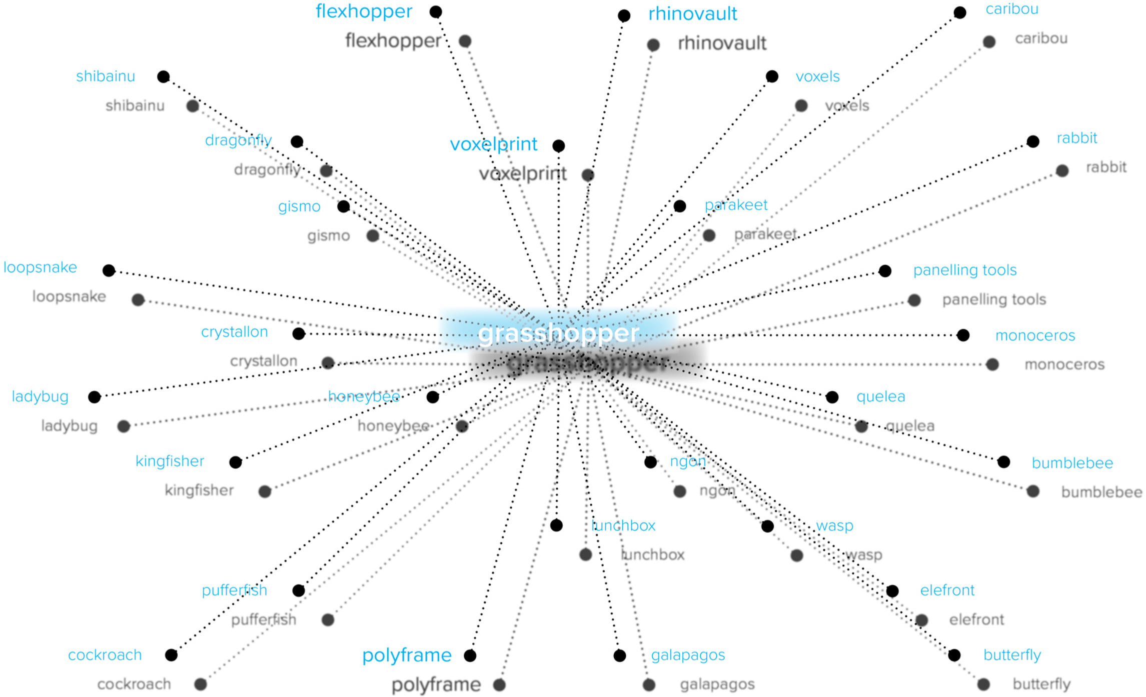
xxx. further bridge development: from columns to deck

xxxvii. resin prints of deck investigation

failure in investigating finite element analysis

xl. investigating concrete printing + theorised fabrication

xlii. concluding remarks


flexhopper
flexhopper
rhinovault
rhinovault
caribou
caribou
shibainu
shibainu
voxels
voxels
dragonfly
dragonfly
voxelprint
voxelprint
rabbit
rabbit
gismo
gismo
parakeet
parakeet
loopsnake
loopsnake
panelling tools
panelling tools
crystallon
crystallon
grasshopper
grasshopper
monoceros
monoceros
ladybug
ladybug
honeybee
honeybee
quelea
quelea
kingfisher
kingfisher
ngon
ngon
bumblebee
bumblebee
lunchbox
lunchbox
wasp
wasp
pufferfish
pufferfish
elefront
elefront
cockroach
cockroach
polyframe
polyframe
galapagos
galapagos
butterfly
butterfly

Overview

Generative Design is the process through which the use of designable algorithms aid in the rapid iteration and development of a singular, or multiple, outcome(s).

Parametric Design is the use of specific parameters, which when plugged into an appropriately designed algorithm provide a usable, designated output. Through changing these parameters, the ability to rapidly, and critically, automatically, adjust the designed outcome to respond to changes in brief or other design requirements allows for a simplified response to an otherwise costly endeavour.

These two processes – generative and parametric design, lead into and are a part of the same overall set of computational design tools. The goal is to use them to simplify the design process whilst still allowing for a significant amount of designability, or creativity.

The application of computational design to solving geometrical and/or structural problems within both architecture and structural engineering has been commonplace since the initial, brute force, application of this approach in calculating the concrete sails of the Sydney Opera House (White, s.d.). It is my intention to use computational design tools to aid in the theoretical development and implementation of my thesis project.

What are Computational Graphic Statics?

As per Juney Lee's research: "Graphic statics is a design and analysis method for two-dimensional (2D) discrete structures, that relies on geometrical rather than analytical or numerical representations of the relation between a structure's geometry and the equilibrium of its internal forces." (Lee, 2018:7)

The principal of two-dimensional and three-dimensional graphic statics relies on the relationship of two interlinked diagrams, one representing the geometric form of a loaded structure, and the second which represents the forces at equilibrium within the structure.

This particular analytical design approach, whilst falling out of favour due to the rise of computing in the 20th Century, has since found a new footing due to computational implementations allowing for immediate, reciprocal, dynamic feedback between these two diagrams, allowing for quicker iterative processes based upon both form and force (Lee, 2018:7). In his thesis, Liem provides a clear and concise breakdown of the procedural aspect of generating force polygons in graphic static analysis. (Liem, 2011:18-19)

Additional research by Masoud Akbarzadeh of the Polyhedral Structures Laboratory has expanded the use of graphic statics to encompass three-dimensional force and form diagrams based upon closed force polyhedrons. (Lee, 2018:7)

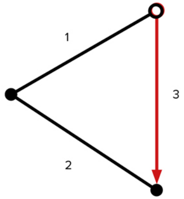
Aims & Limitations

The BRG @ ETH Zurich developed and taught a masters + doctoral level course on computational graphic statics in the autumn of 2020. As I was not a student at ETH Zurich, nor do I have a background in structural engineering (and my a-level mathematics is distinctly rusty), there is a legitimate question to answer as to whether I am competent to teach myself the necessary basics of computational graphic statics in the timeframe allotted in this module, as well as implementing it as a design tool for the purposes of architectural development.

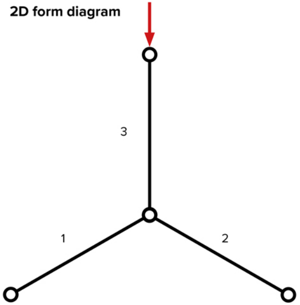
What do I hope to achieve?

I seek to find an architectural application of these computational design tools, with allowances made for my own unorthodox academic background, seeking to drive structural, formal and programmatic responses to my thesis project.

2D force diagram



2D form diagram



https://block.arch.ethz.ch/eq/drawing/view/1

block research group

The Block Research Group, based at ETH Zurich and headed by Prof. Dr. Philippe Block with Dr. Tom Van Mele, is also focused on advancing geometrical solutions to complex structural design. This research has routes in the analysis of masonary structures along with graphical analysis and design methods brought about by computational form finding for structural design (amongst others).

To this end, my primary interest in this group is their development of another plugin for Grasshopper called Rhino Vault. In it's second major iteration, it is now based upon the open-source, python based COMPAS computational framework also developed by the BRG. Similarly to PolyFrame, RhinoVAULT 2 is developed for funicular form-finding, but is limited to compression-only forces.

Examples of research and articles are shown on the opposite, (clockwise from top left):

Exploring Three-dimensional Equilibrium. Massachusetts Institute of Technology

Geometry-based Teaching of Structures Through Computational Graphic Statics

Redefining structural art: strategis, necessities and opportunities

Morph & Slerp: Shape description for 3D printing of concrete

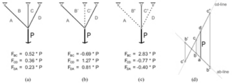
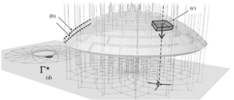
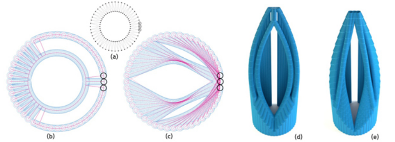


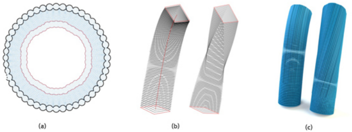
Figure 3.8 – For the well-known indeterminate problem of a weight on three bars, (a), (b) and (c) show different, possible states of self-stress of this system. Case (a) corresponds with the linear elastic solution. (d) shows the indeterminate force polygon for this problem (after Block and Ochsendorf [2005]).



Block, P. (2009)

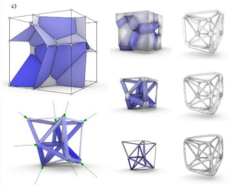


6: Displacement trajectories: (a) User-provided graph connecting centers of source RBF (black circles) and sink dots); (b) a family of segmented curve handles generated by walking between source (black circles) and target (pink dashed circles) marked on graph in (a) and computing the so-called Dijkstra paths; (c) smoothed version c of curves shown in (b). It can be noted that the zero-contours of the shape (blue) evolve along the control curv) and (c); (d,e) resulting 3D shapes corresponding to (b) and (c) respectively, with the smooth family(c) producing curvature and inclination in the print layers, hence more amenable to printing with wet concrete. Thus control ha to (c) is recommended.



7: Plane interpolation: (a) Intermediate profile curves (blue) obtained by interpolating between start (red) anc profiles using the Morph operation (Section 3.1); (b-left) Frenet-Serret moving frames extracted from a curve

Bhooshan, S. et al. (2020)



rating creases using RV2 and Rhino3GS; b) comparing d Rhino3GS; and c) materialisation of spatial structures nd 3D skeleton features

Lee, J. et al. (2021)

Learning from the past

Built for the Venice Architecture Biennale 2016, the Armadillo Vault stood as a statement of strength through geometry rather than through 'an awkward accumulation of matter'[4]. Comprising 399 CNC-cut limestone voussoirs and held together in compression without mortar, glue or reinforcement, the vault spanned 15m with a minimum thickness of just 5cm (Figure 1).

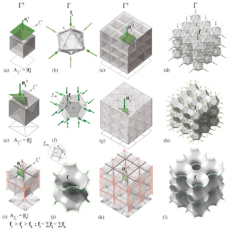
Inspired by Gothic stone vaults, some of which are proportionally as thin as an eggshell, the Armadillo Vault demonstrated the elegance of achieving strength through structural geometry and of effective use of material, made possible by the latest advances in computational design, optimisation and digital fabrication methods[24,25].

Because most modern engineers are stuck in 'Navier's straightjacket', as Heyman rightly puts it[26], such equilibrium structures can no longer be designed today, be it due to inappropriate analysis methods or the constraints of building codes[10]. These exceptional funicular structures are not driven by stress considerations, but demand a good structural form. We need to (re-)educate when used effectively, weak materials with low embodied carbon can be activated as structural materials.

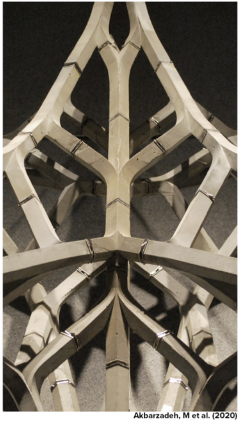
For the Sustainable Urban Dwelling Unit (SUDU) built in Addis Ababa, Ethiopia in 2010 (Figure 2), tile and pitched-brick vaulting techniques were employed for the floor and roof respectively[28]. As a result, formwork was not needed, only strings to describe the geometry. More importantly, the tiles had no bending capacity, but because they were placed to follow a tied arch, and stiffened using spandrel walls, their compression strength of only 2MPa was sufficient to guarantee structural safety under all loading cases. Using soil from the site reduced pollution due to transportation of materials, and only a minimal amount of cement (7%) was required to stabilise the hand-pressed, air-dried tiles.

For the ETH Zurich Pavilion built for the New York Ideas City Festival in 2015, local waste products, specifically Tetra Pak from beverage containers, were compressed to form the 9mm thin sheet material from which the lightweight voussoirs that made up the arches were assembled. Thanks to post-tensioning with truck belts and triangular arch-

Block, P. et al. (2020)

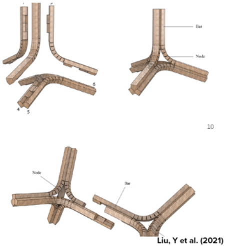


Akbari, M et al. (2022)



Akbarzadeh, M et al. (2020)

Akbarzadeh, M et al. (2017)



Liu, Y et al. (2021)

The Polyhedral Structures Laboratory†, based at the University of Pennsylvania's School of Design is led by Masoud Akbarzadeh. Its intent is to act as an interdisciplinary lab, linking architecture with the disciplines of structural engineering, computer science, mathematics and material science, leading to a far more competent approach to structural design within the field of architecture.

The primary aspect of this group's research I wish to investigate is the use of PolyFrame - a plugin for Grasshopper which provides a "computational framework for form finding" through the "construction of reciprocal polyhedral diagrams of 3D graphic statics for conceptual structural design purposes", through both compression and tension based force systems..

Examples of this lab's research outputs are shown opposite, which include (clockwise from top right):

Hedracrete: Prefab, Funicular, Spatial Concrete

Kerf Bending + Zipper in Spatial Timber Tectonics

Saltatur: Node-Based Assembly of Funicular Spatial Concrete

Strut Based Cellular to Shellular Funicular Materials

† https://psl.design.upenn.edu/

examples of two-dimensional graphic statics

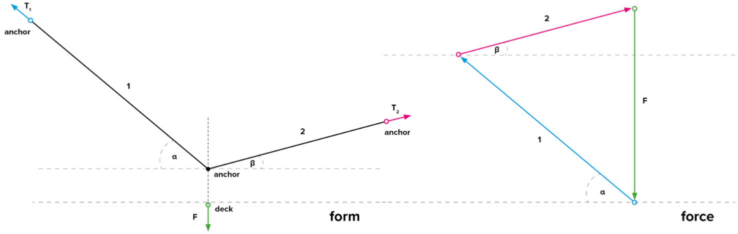
In this first, simple asymmetric tension system, the tension force vectors $\mathbf{T_1}$ & $\mathbf{T_2}$ can be calculated using the following equations:

$$\mathbf{T_1} = F / [\cos(\alpha) * \sin(\beta) / \cos(\beta) + \sin(\alpha)]$$

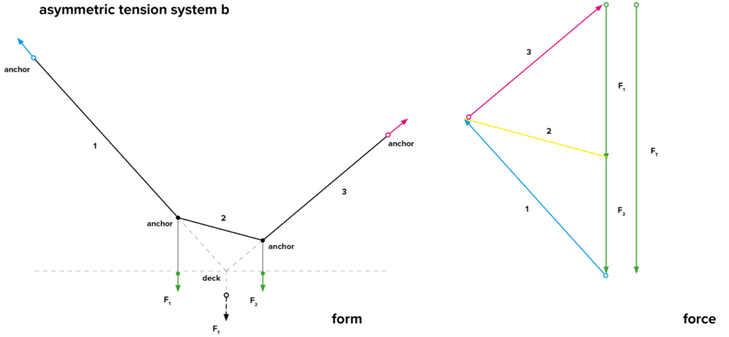
$$\mathbf{T_2} = F / [\cos(\beta) * \sin(\alpha) / \cos(\alpha) + \sin(\beta)]$$

where **F** is the weight of the deck acting through the anchor (given by **F = m * a**, where **a** is 9.81 ms^{-2}).

asymmetric tension system a

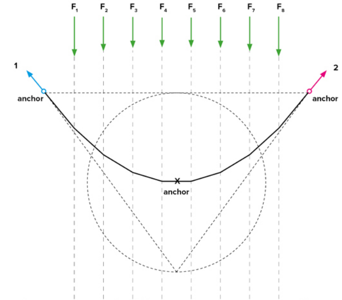


asymmetric tension system b

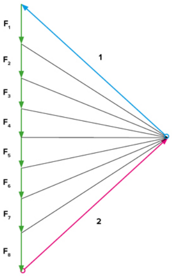


The following diagrams are simplified and drawn from those provided by the eQUILIBRIUM portal, developed by the Block Research Group (Block Research Group and Schwartz, 2022). Additional mathematical research provided by Omni Calculator (Omni*, 2022).

catenary approximation



form



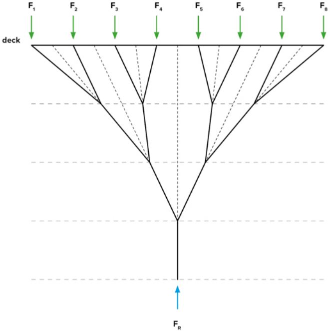
force

As per the guidance on the eQUILIBRIUM website:

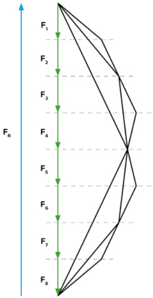
“The form a cable takes for a uniformly distributed load is equivalent to a parabola, for self-weight it is equivalent to a catenary. The parabola is often used as an approximation of a catenary as it can be constructed easily using geometric techniques.” (Block Research Group and Schwartz, 2022)

In this example, a parabola is used as an appoximation for a catenary system.

symmetric tree structure



form



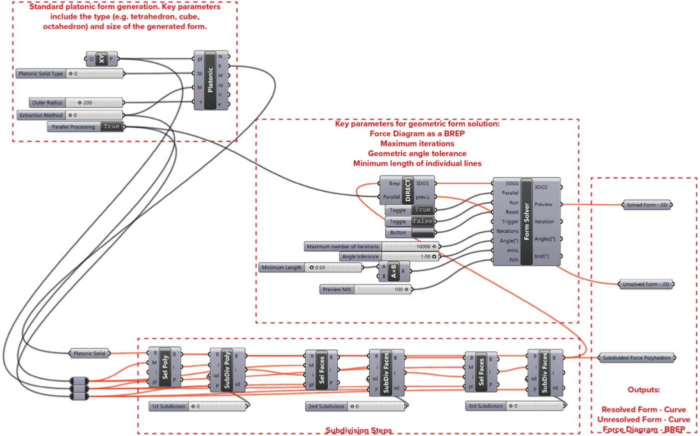
force

In this symmetric tree structure, forces $\mathbf{F_1}$ through $\mathbf{F_8}$ are all equal, and $\mathbf{F_R}$ is the reaction force

investigative method - polyframe

Building upon the information provided in a tutorial by Daniel Dolatabadi (Form Finding using '3D GRAPHIC STATICS' Plugin For 'Grasshopper', 2020), I connected this script to aid in exploring a series of pre-set polyhedrons provided by the PolyFrame plugin. The intent was to find, through trial and error, any geometric or aesthetic patterns that may emerge through the generation of a series of forms based upon three levels of subdivision of the polyhedron. Constants for this experiment are the minimum edge length of the generated form, the angle tolerance between edges, the maximum number of iterations, and the edge extraction method. Variables which changed were the level of subdivision (0-3) at each stage.

My original plan was to methodically move through each pre-set shape that the plugin offered, to build up as great an understanding as I could about the graphical interplay and inherent variety within graphic statics. This would have led to a base number of iterations of 2,240 (43 possible subdivision steps of 35 pre-set shapes), but with these initial explorations demonstrating the futility in a brute force approach, along with a lack of skill in programmatic automation within grasshopper that I currently have, I will take a more circumspect investigative route. To this end, I have only generated forms based upon 3 platonic solids (a tetrahedron, cube and octahedron) and 3 Archimedean solids (truncated tetrahedron, snub cube, truncated icosidodecahedron). These polyhedra were chosen for their variety and utility in analysing force diagrams along multiple axes of symmetry which would ultimately lead to a greater flexibility in understanding their application to generating a cross-section of a bridge span as well as additional structural elements that complete a bridge. All of these force diagrams generated are based upon the equilibrium found through compressive forces, not tensional forces. At this point, I am still uncertain as to how tensional representation is provided using either the grasshopper plugin, or the native scripts within Rhino by the PolyFrame algorithm suite.



Standard platonic form generation. Key parameters include the type (e.g. tetrahedron, cube, octahedron) and size of the generated form.

Platonic Solid Type 0
Outer Radius 200
Extraction Method 0
Parallel Processing True

Key parameters for geometric form solution:
Force Diagram as a BREP
Maximum iterations
Geometric angle tolerance
Minimum length of individual lines

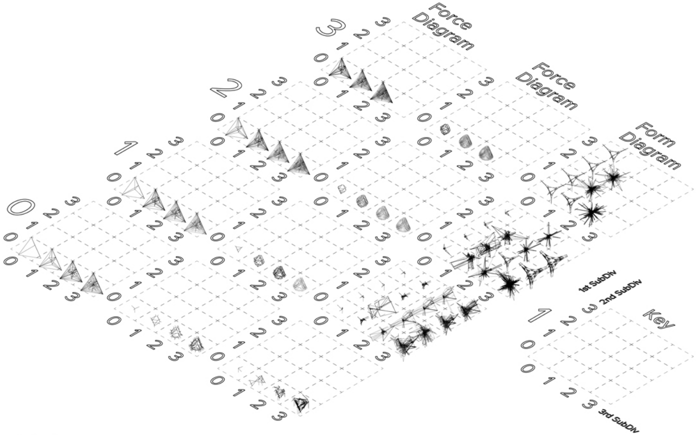
Toggle True
Toggle False
Maximum number of iterations 10000
Angle tolerance 1.00
Minimum Length 0.50
Preview Nth 100

Subdivision Steps

1st Subdivison 0
2nd Subdivison 0
3rd Subdivison 0

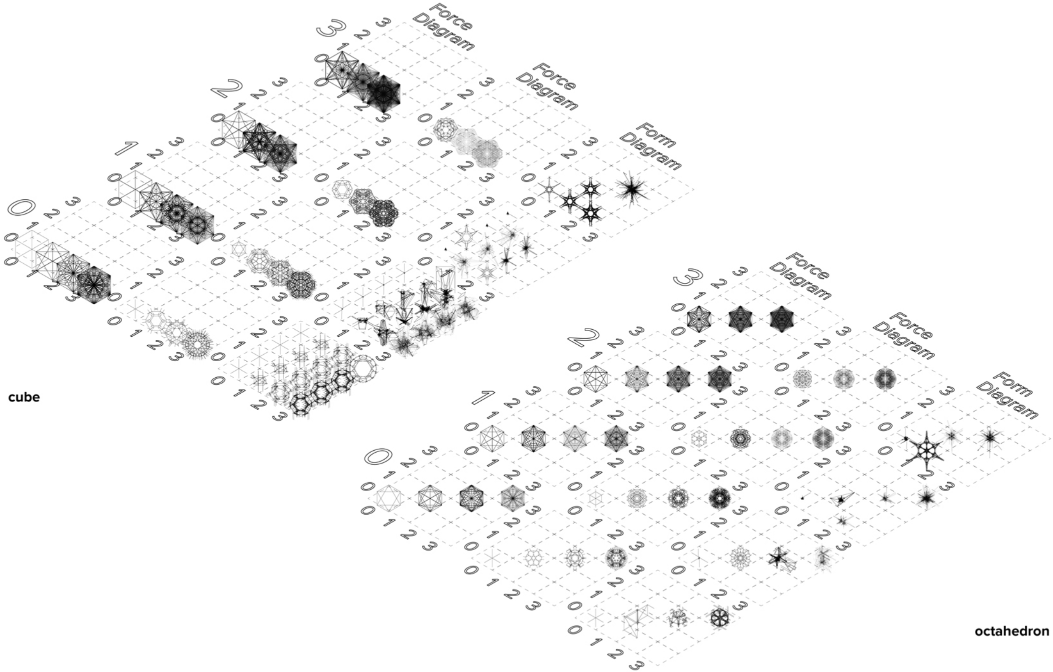
Outputs:

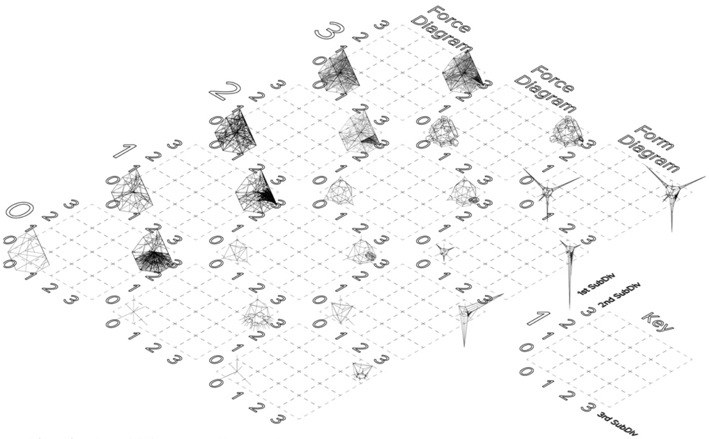
Resolved Form - Curve
Unresolved Form - Curve
Force Diagram - BREP



tetrahedron

Through this initial experimentation, several issues quickly became apparent – some which are obvious, some less so. An obvious issue from which the futility of a brute force approach was made apparent was that the greater the complexity in geometric form, the longer the computational time it requires to both be generated and ultimately resolved. This convinced me to limit my investigations to six polyhedral forms until I gain a greater understanding of what it is I am seeking from this particular method. The next issue discovered was that only two tiers of subdivision affected how the polyhedron developed, the third had no effect at all. Whilst on the following pages each iteration is placed in the appropriate grid denoting which tier and level of subdivision led to the generation of this form, it was always the case that the subdivision logic was as follows: “First Subdivision AND (Second OR Third Subdivision)” was true as opposed to what I originally expected, which was “First Subdivision AND Second AND Third Subdivision”. Despite analysing the grasshopper script, I have yet to understand what mistake I have made that has caused this difference in expected versus actual logic. The last issue occurred due to a lapse in what could be called experimental hygiene: each time I changed the input parameters – polyhedron type, subdivision level – the form solving algorithm required switching off and resetting. I did not always do this, as I only realised the necessity of this much later in my investigative process. Thus, some of the forms that failed to generate may have done so due to the classic “PEBKAC” (problem exists between keyboard and chair) as opposed to an inherent issue with the input parameters.


Force Diagram
Force Diagram
Form Diagram
0
1
2
3
cube
octahedron



archimedian type 00 / truncated tetrahedron

Successful iterations were decided upon via following criteria:

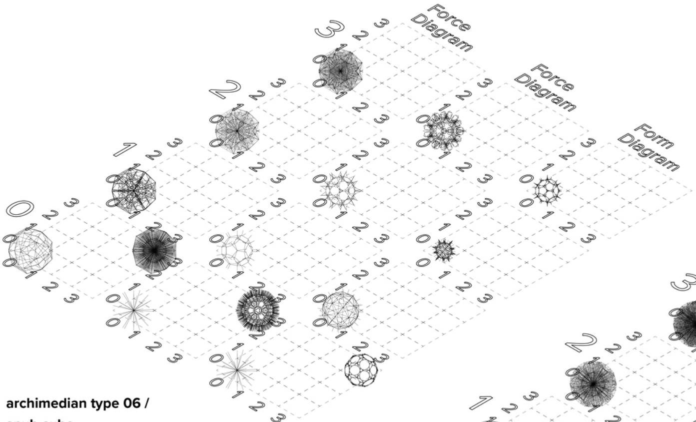
if after reaching iterative completion (the algorithm completes computation before hitting 10,000 iterations) and upon a visual inspection of the geometric form it is decided that the form in question is resolved, then it is deemed a successful iteration.

if the iteration reaches computational exhaustion (i.e. hits 10,000 iterations but is not computationally complete), the iteration is deemed to be unsuccessful.

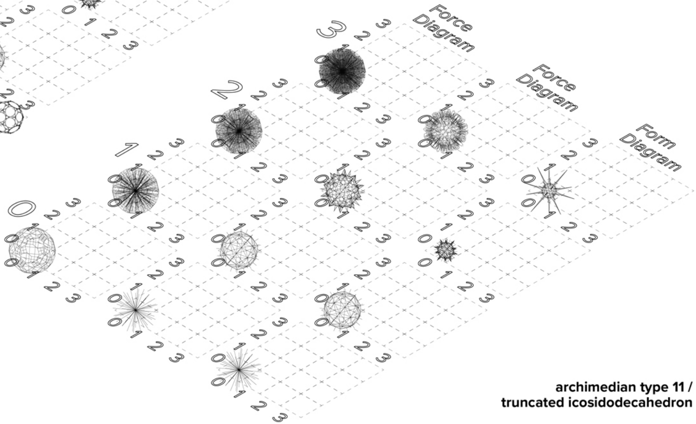
if the iteration fails a visual inspection, the iteration is deemed unsuccessful.

if after reaching iterative completion (the algorithm completes computation before hitting 10,000 iterations) and upon a visual inspection of the geometric form it is decided that the form in question is resolved, BUT an identical, successful form has already been generated within the same primary subdivision set, then the iteration is decided to have been unsuccessful.

Moving forward with experimentation and structural investigations, I've decided to focus upon the force polyhedra of Cubes and Tetrahedrons, along with three and four sided prisms, as I feel at a basic level these force polyhedra can be manipulated to provided structural elements forming a bridge.

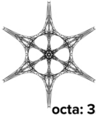


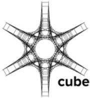
**archimedian type 06 /
snub cube**



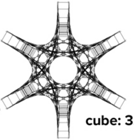
**archimedian type 11 /
truncated icosidodecahedron**

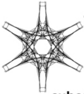
platonic iterations


octa: 3.0.1


cube: 3.0.1

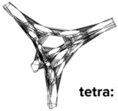

cube: 3.0.0

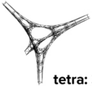

cube: 3.0.2


cube: 2.0.2


cube: 2.1.0


cube: 0.3.3

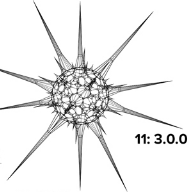

tetra: 2.1.3

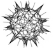

tetra: 3.0.1


tetra: 2.1.2

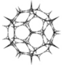

tetra: 3.0.0

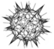
archimedean iterations


11: 3.0.0

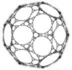

11: 2.0.0

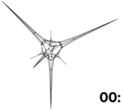

11: 1.0.0


06: 3.0.0


06: 2.0.0


06: 1.0.0


06: 0.3.3

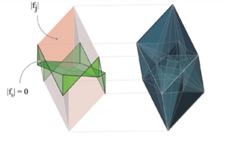

00: 2.0.0


00: 3.0.0

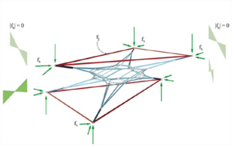
investigating saltatur

For this exercise, I utilised both the grasshopper scripts as well as the Rhino-based PolyFrame utilities to investigate and generate a geometric form approximating that of the Saltatur project (Akbarzadeh et al., 2020), to form an understanding of the workflow process and the representation of both physics and geometric resolution.

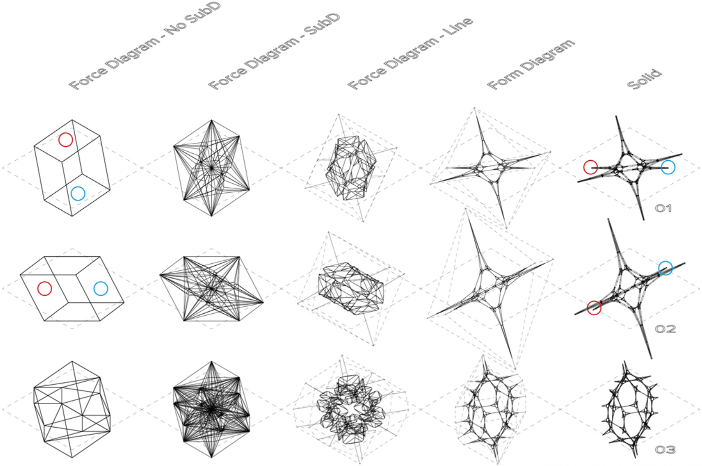
Starting with a basic form which I felt best approximated the diagrams shown to the top right, I generated an outcome that at first visual inspection is a rough approximation of the Saltatur project. However, multiple issues with this first generative attempt include: a wider body than the Saltatur outcome, extremities that do not completely line up in the same fashion, and missing struts. The tri-axis symmetry is however present. In all likelihood, I have misunderstood, or simply missed some steps during the generative phase - a real possibility as there are a few elements such as force adjustment and vertex repositioning that I have not yet exhaustively investigated. In a further attempt to refine the iteration to a closer approximation, I generated five additional forms, each based upon a force diagram which I drew as I attempted to understand the physics and polyhedral graphic statics translation. Unfortunately, each successive generation moved further away, rather than towards, the Saltatur output, which is why I believe the first iteration requires another look and some tweaking using the PFPerp, PFPlanarize + PFTransform commands.



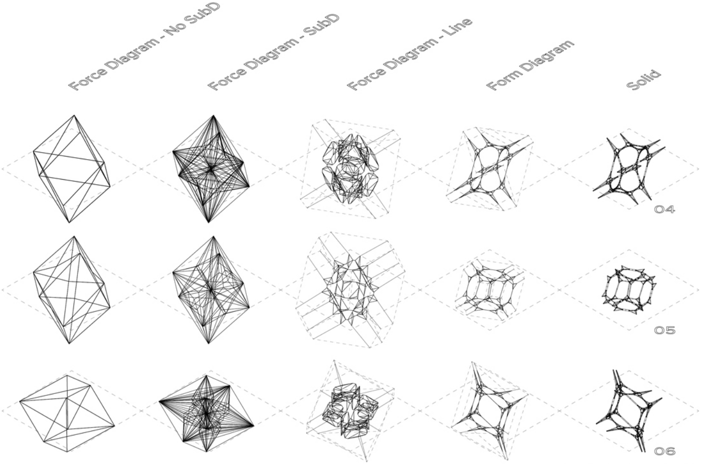
Akbarzadeh, M et al. (2020:3)



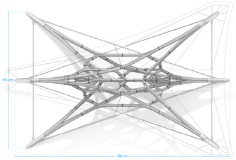
Akbarzadeh, M et al. (2020:3)



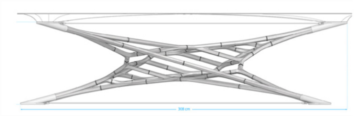
Iterations 01-03 of my investigation of the saltatur project by Akbarzadeh et al.



Iterations 04-06 of my investigation of the saltatur project by Akbarzadeh et al.



Akbarzadeh, M et al. (2020:3)



Akbarzadeh, M et al. (2020:3)

The intent behind the shift from iteration 01 to 02 was to compensate for the position of the extending struts, which in the Saltatur project are positioned directly over each other when viewed in elevation. The struts in question are circled in red in the diagrams opposite, and the corresponding faces on the force polyhedron are also indicated. Whilst this adjustment succeeded in bringing the indicated struts into the same plane, it caused the rest of the structure to warp in an unanticipated manner. The intent of iteration 03 was to model the faceted force polyhedron more closely as found on page xii, however my lack of understanding of the differing methods of visualising the planes of the force polyhedron led to unintended consequences - namely, an unrecognisable iteration when compared to the Saltatur project: a significant number of additional external struts appeared. Iteration 04 attempted a different approach, with the intent of providing additional internal complexity, but did in fact result in duplication of the external struts. I have since understood that this is because each external face corresponds to a single external force; thus, upon geometric resolution, each external force is met by a strut to compensate for compressive forces. Building upon the previous iterations, I reconfigured the force polyhedron once again to try and better replicate the Saltatur project. First visual inspection of iteration 05's subdivided force diagram, much like iteration 04's seems to correspond with the force polyhedral found on page xii. However, the resultant geometric form diagram is once again vastly different to the Saltatur project. It was at this point that I decided to break off this particular investigation and revisit my assumptions.

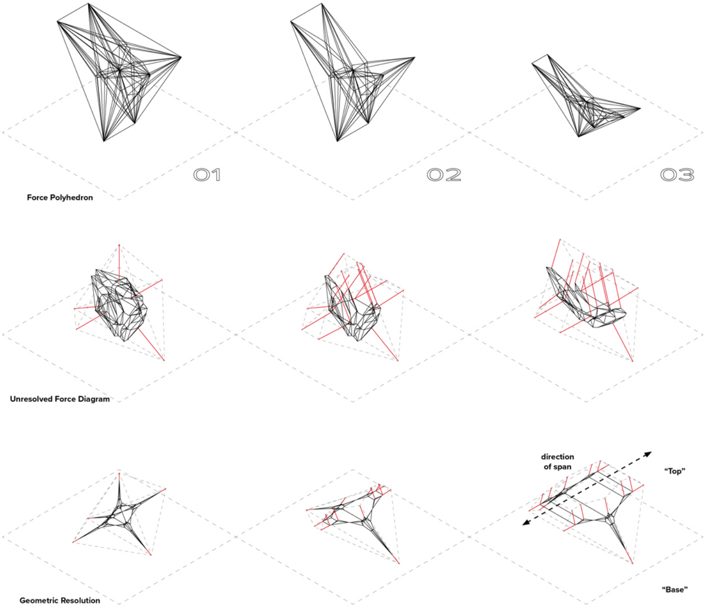
structural development - part one

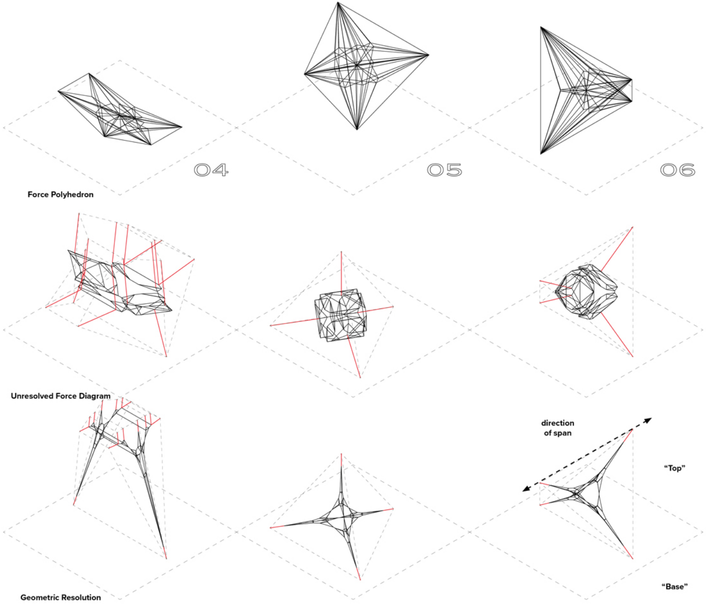
Using the understanding I have developed through experimentation of the process that in both attempting to reverse engineer the Saltatur result as well as the initial explorations, I began to explore using both a simple, undivided, three-sided prism and a simple, undivided, tetrahedron to develop some initial forms that can potentially be used as structural elements for the bridges.

From a simple three-sided prism, using the same subdivision method as in the initial investigations, I set the subdivisions to 3.0.0 (1st.2nd.3rd), and this was the setting I used throughout this particular iterative process, for both forms derived from the three-sided prism, as well as the tetrahedron. Rotating the prism so that it sat on one of the short edges, iteration 01 was resolved.

From iteration 01, I removed the top one-third of the prism, including internal subdivision to that third, such that the polyhedron now had four external faces. The logic behind this move was dictated by the desire to represent how I felt the forces would be distributed from an independent bridge span resting upon the supportive members.

Iterations 03 + 04 were simple transformation commands, elongating the force polyhedra such that the forces acting perpendicular to these planes would tend towards the z-axis, and thus reduce the footprint of the structural members.





Iteration 05 began with the same principle of Iteration 01, in that I rotated the force polyhedron such that the greater number of structural supports would be tending towards the ground plane. However, I decided to adjust this such that an equal number of supports would meet the ground and the bridge span, thus moving to Iteration 06.

From Iteration 06, the same principal of subtraction applied in moving from 01 to 02, in that I removed half of the tetrahedral form, including its subdivisions, such that I increased the number of external facing planes by two, and thus the number of external forces acting on the polyhedron by two, reconfigured it to a form that I felt better suited the aforementioned independent span resting upon the structural member.

Iterations 08 + 09 were also simple transformations designed to reduce the footprint of the structural members.

Following on from the generation of these simple iterative forms, I needed to look into a way to provide indicative thickness for the purposes of digital representation, 3D printing as well as future design resolution. There are five methods that I have so far tested, four of which provided results, and the fifth, whilst potentially far more useful for ultimate design resolution has thus far proven uncooperative.

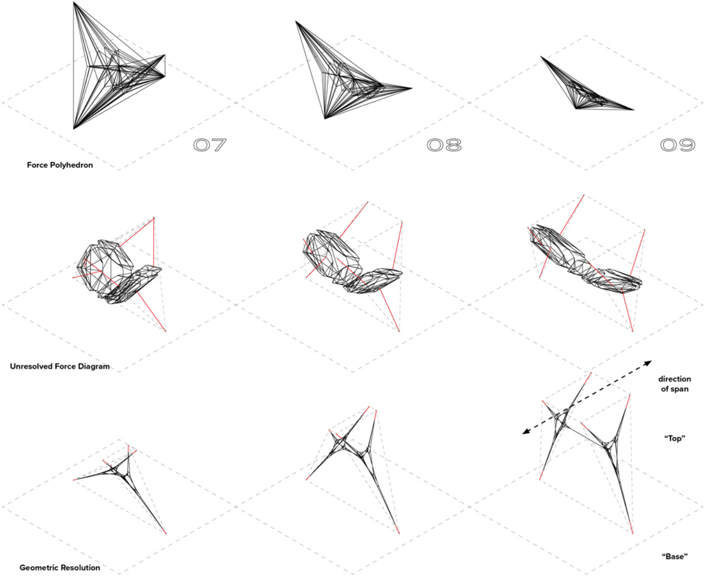
The four currently successful methods are as following:

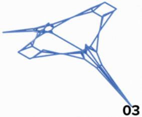
the use of PolyFrame's built-in piping algorithm that provides thickness variance dependent upon the compressive forces involved

a rudimentary cross-sectional generation algorithm as part of the set of tools provided by PolyFrame in Grasshopper. This provides options for manual thickness and rotational variance, amongst other constraints.

a voxel-based meshing algorithm called Dendro which allows for manual thickness and smoothness variance dependent on the size of both voxel and piping radius.

a multi-piping algorithm provided by Grasshopper as standard that outputs a sub-divided surface, which allows for manual variance in thickness, node offset and various fitting options. It is this forth algorithm that I used to generate the meshes for resin printing sample iterations at 1-1000 for a visual inspection and rudimentary physical investigation.

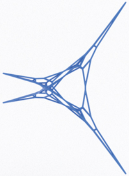




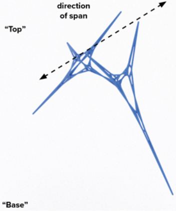
03

04

four forms generated through different methods, rendered with translucency



06

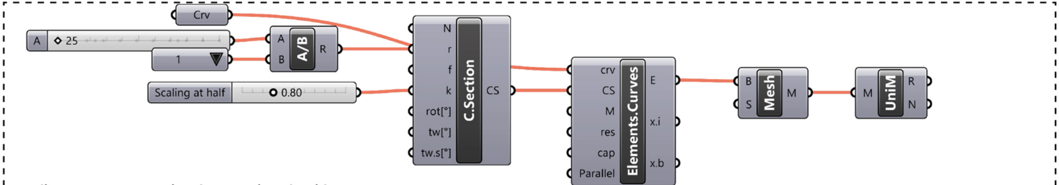


08

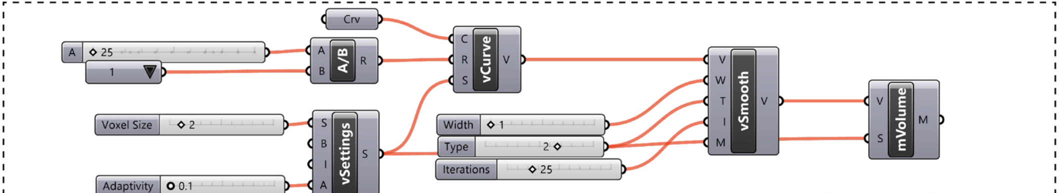
09

The fifth, currently uncooperative cross-sectional generation method, requires a form generated within the Grasshopper segment of PolyFrame’s tools as opposed to the Rhino segment; and thus far trying to reverse engineer the outputs derived from the Rhino segment have proven to be futile. Should I succeed with this method, the advantages from this particular set of algorithms allows for a customisable cross-sectional area, as opposed to a pre-generated form based upon a simple polygon, along with customisable nodal elements, all the while keeping the physics calculations intact. Customisability is ultimately the advantage to this particular method of generation.

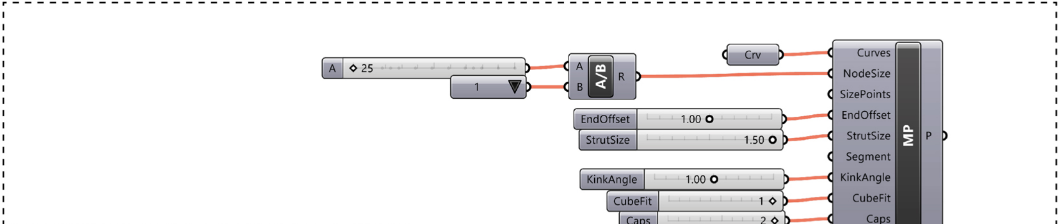
Ultimately, the use of both the Dendro plugin as well as the sub-division multi-piping command provided a suitable form for exporting to .stl for resin printing. Photographs of the resulting resin printed maquettes can be seen on the following pages. These prints were scaled roughly to between 1:1000 and 1:500.



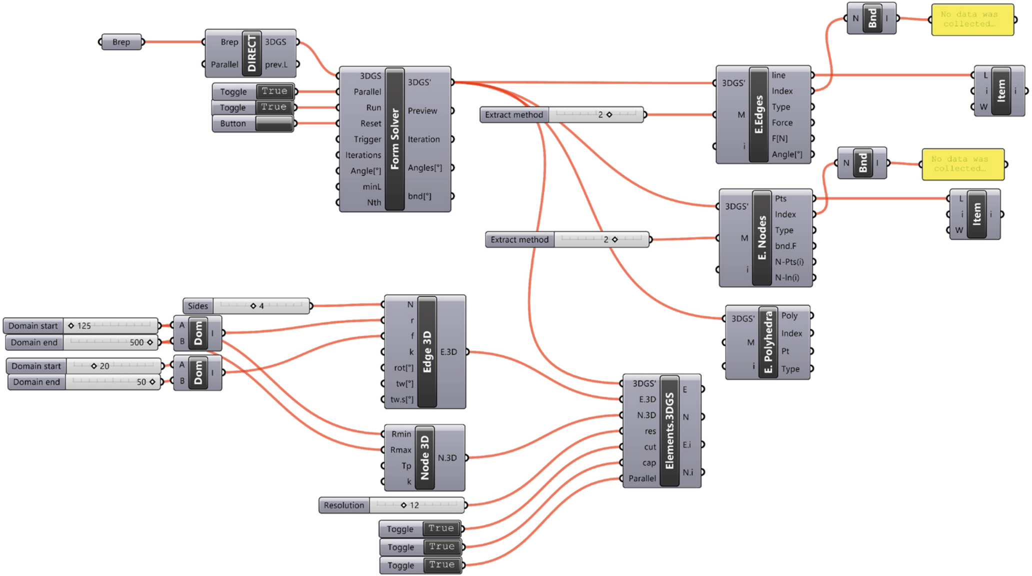
a rudimentary cross-sectional generation algorithm



a voxel-based meshing algorithm called Dendro

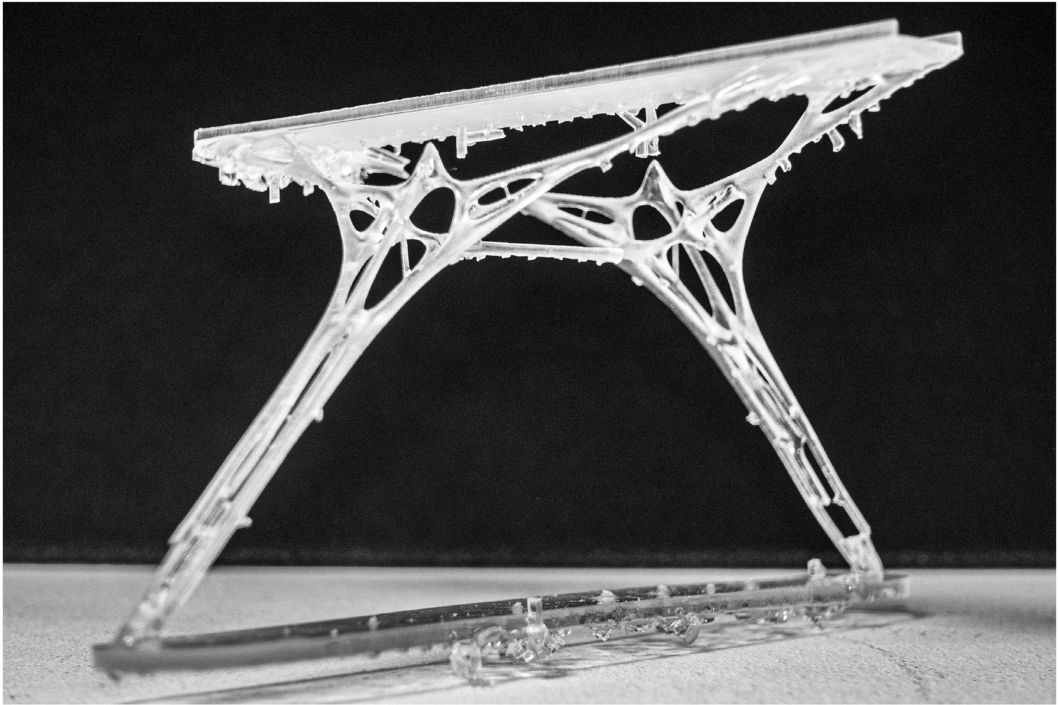


a multi-piping algorithm provided by Grasshopper as standard

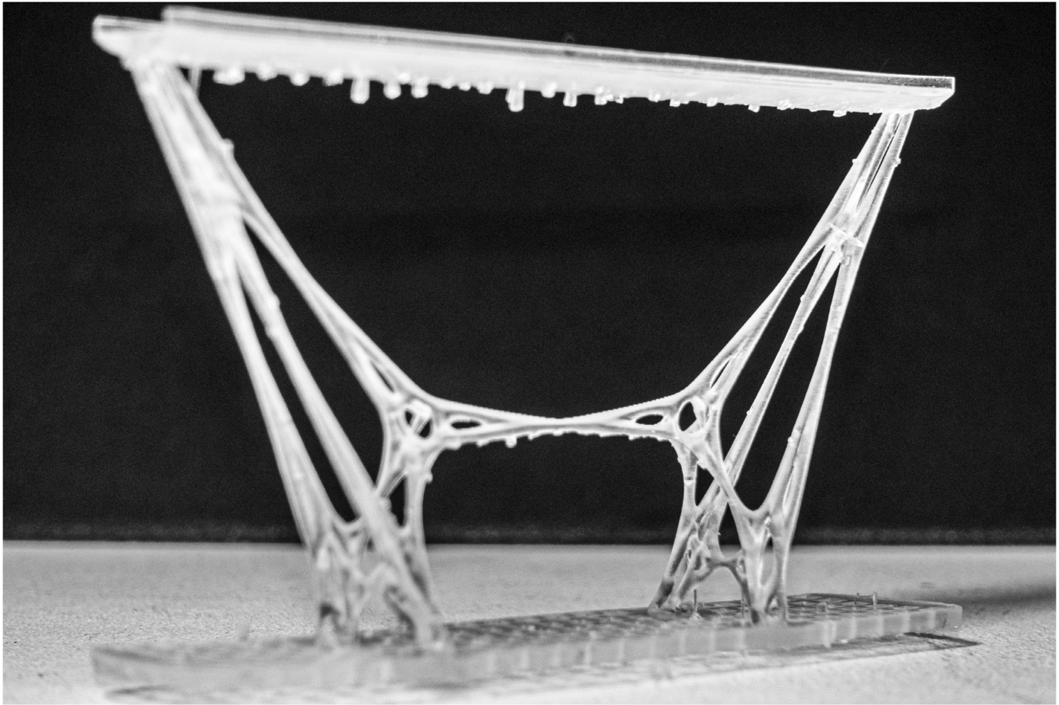


the fifth, currently uncooperative cross-sectional generation method

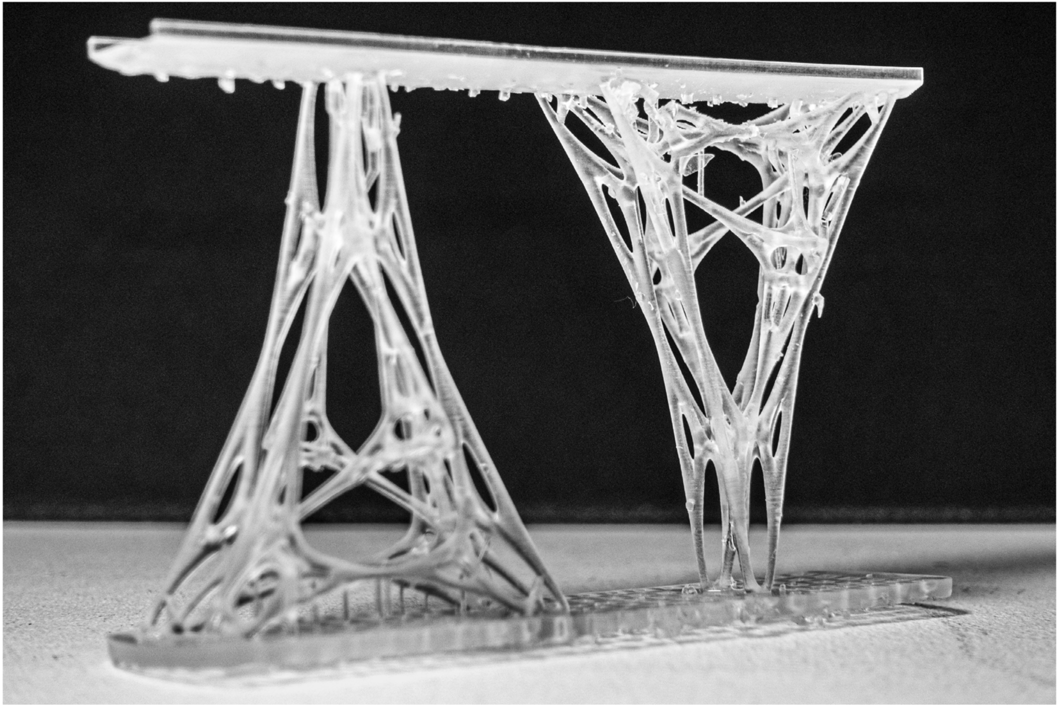
maquettes of initial forms



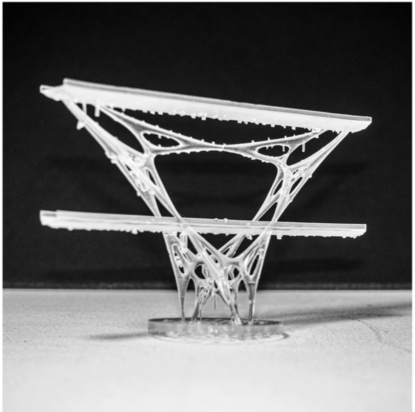
Resin print of iteration 03 with indicative bridge span (Dowson, 2022)



Resin print of iteration 08 inverted with indicative bridge span (Dowson, 2022)



Resin print of manually modified iteration 09 with indicative bridge span (Dowson, 2022)



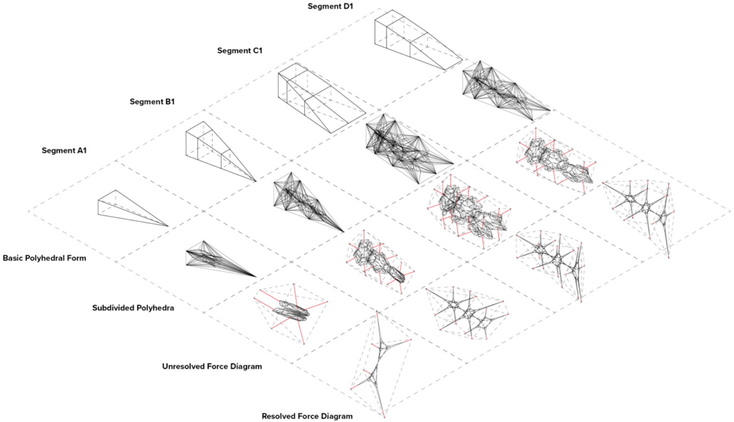
Resin print of iteration 06 with indicative double bridge span (Dowson, 2022)

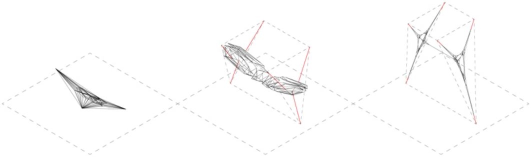
Resin print of manually modified iteration 09 with indicative double bridge span (Dowson, 2022)

structural development - part two

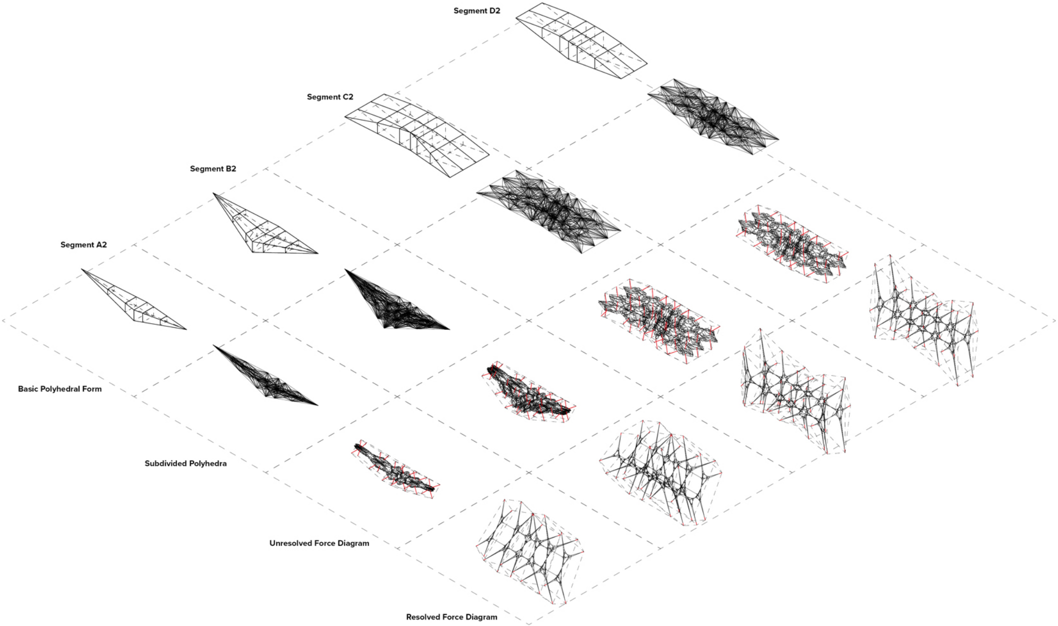
Taking one of the earlier generated forms, specifically, iteration 09, the intent was to use this form as an indicative pair of structural columns that would cradle a deck span running between them. This developed into an exercise of exoskeletal investigations which ultimately would no longer bare any immediate structural association with iteration 09, and thus require an alternative approach to marrying structural supports to the deck itself, which will be tackled later on. In addition to iteration 09, the polyhedral development at this stage was inspired by diagrams A & B from Figure 6.2 of two different configurations of polyhedral cells on p112 and Figure 6.5 on p116 from '3D Graphical Statics Using Reciprocal Polyhedral Diagrams' (Akbarzadeh, 2016:112,116). In all likelihood, at this stage I will move away from generating forms using the PolyFrame system, and manually design a (series) of structural supports that would morph into a column system capable of supporting two layers of these decks – one at 56m above ordnance datum, and one at 28m above ordnance datum, in accordance with the deck heights proposed in the thesis project.

The other reason I will be moving away from using the PolyFrame system at this stage is that I feel that I lack the confidence of understanding at fundamental level of the generative method behind this system. For instance, looking back at the initial generative experimentations and the multiple failures, reading Akbarzadeh et al. and their description of the construction of a force polyhedron from first principles did suggest to me that one of the reasons behind these failures could have been that those specific subdivided forms failed to find an equilibrium of the forces involved (Akbarzadeh et al., 2016). I mention this now, as during the development of this exoskeletal deck support structure, I attempted to marry it to a column generated in a similar fashion. However, this separate column element kept throwing an error specifying that certain faces or edges required further manual subdivision – thus suggesting to me that a global equilibrium had yet to be established.

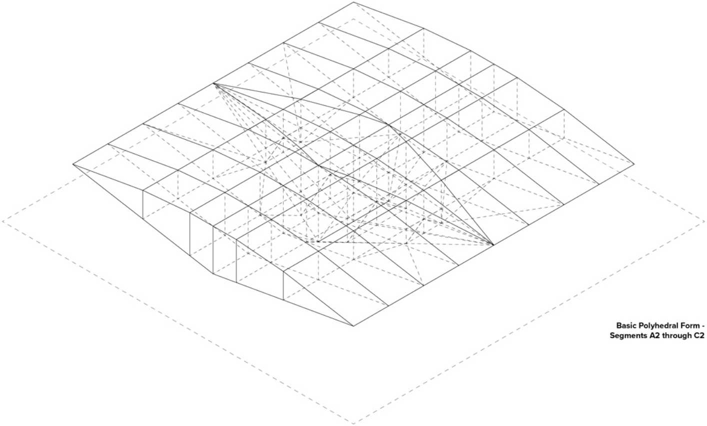




Iteration 09, brought forward


Segment D2
Segment C2
Segment B2
Segment A2
Basic Polyhedral Form
Subdivided Polyhedra
Unresolved Force Diagram
Resolved Force Diagram

The next failure in my comprehension of form generation was in how (and more importantly, why) to limit the external forces, their direction and the magnitude of them, along with the reasoning behind such measures. This is already configured by the force polyhedra I have drawn, especially how external forces act on the system (remembering that each planar face represents a force acting perpendicular through it), however, there is a measure of control offered by adjusting the magnitude of the forces involved within the system or acting upon the system – which can affect the geometry of the system in question, either forcing it to scale or reconfigure as appropriate. From my limited understanding, this element of theory is covered by Akbarzadeh when he discusses the differences between determinate and indeterminate systems along with compression-only and general polyhedral frames (Akbarzadeh, 2016:100-105). To this end, I believe what I have generated in this process is a compression-only representation (following that PolyFrame continues the convention that blue lines represent compression forces, red tension and green external as described on p99 (Akbarzadeh, 2016:99)). Whether this particular polyhedral cell formation can exist in a tension/compression or tension-only configuration with external forces of equal magnitude acting upon the cells is not a question I can answer – other than to assume that as this is the form that was algorithmically generated by the PolyFrame system, the answer is that this form only exists as a compression-only system in this instance. Is this distinction important here?



Basic Polyhedral Form - Segments A2 through C2

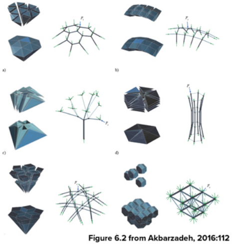


Figure 6.2 from Akbarzadeh, 2016:112

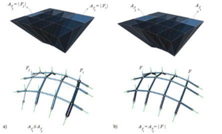
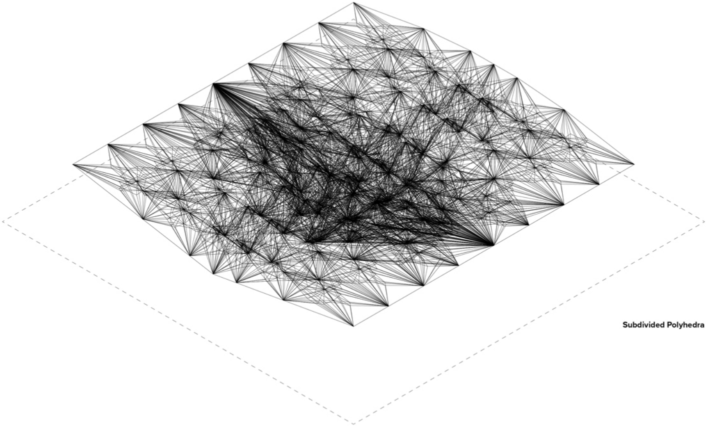


Figure 6.5: a) A given polyhedral frame and its reciprocal force diagram with different areas of the faces corresponding to the applied loads and b) the same polyhedral frame with optimized force diagram with equal areas per face corresponding to the applied loads.

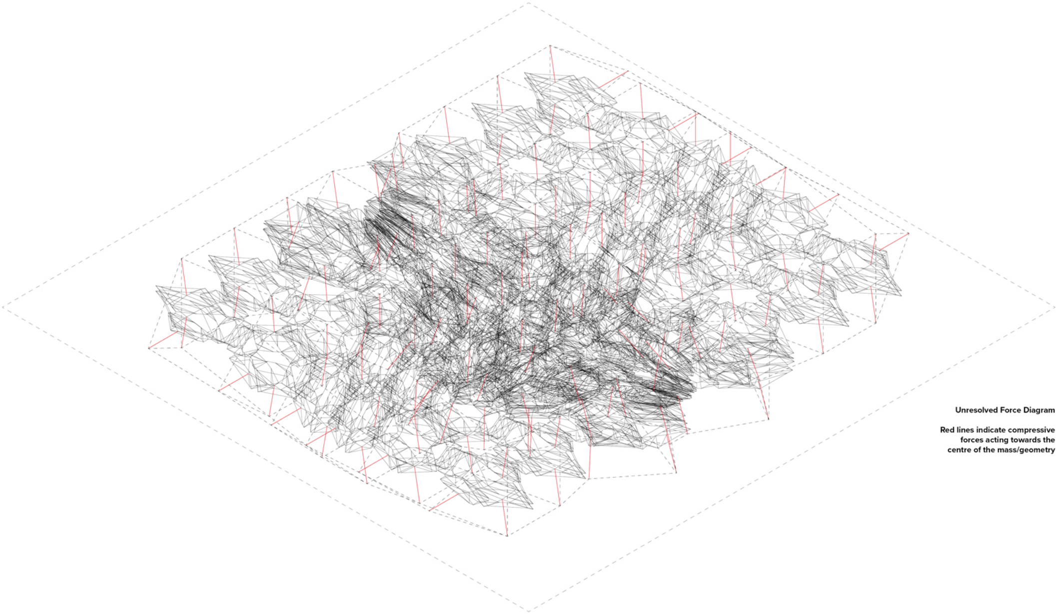
Figure 6.5 from Akbarzadeh, 2016:116

At a basic level of structural understanding, this bridge, like any structure, will be subjecting to dead loads, and live loads. A vertical dead load in this example would be one of the interventions I place upon the bridge – from something as simple as a seat to a commercial module resting upon the deck. A live vertical load would be a pedestrian, cyclist, or other user of the bridge transiting from one point to another. Then there are oblique live loads such as wind shear, along with any ground movement that would affect the load distribution. How I would envision expanding this particular structural development method would be by plotting all vertical dead loads along the entire span of the bridge, then using this load information to determine the magnitude of compressive forces along the span of the bridge, which would adjust the resultant generated geometry. From an aesthetic level, this would be a distinct visual indicator from the ground of the various programmes at play across the span of the bridge, along with providing an asymmetrical and yet still structurally sound aesthetic. Of course, the limitations of this are not just in today's technology or my understanding of this particular method of design production, but it would severely limit any evolution of the programme of inhabitation of the bridge over time, as certain areas would be forever limited in their dead load capacity.

Having broached the topic of aesthetics, this brings us to a question of representation of structural form, and the sensible manifestation of the digital into the physical realm. Two immediate fabrication representations come to mind, partly in response to the digital representation used thus far – one being concrete, and the other being steel hollow section, both of which would require a novel approach to fabrication. For now, I am going to focus on how this may be fabricated using concrete, with a view to using additive manufacturing techniques (a.k.a. 3D printing). I fully expect this technique to change the aesthetics of the form developed thus far. However, what piqued my curiosity about applying additive manufacturing techniques to bridge construction, or bridge segment construction, were three projects: one was MX3D's steel printed canal bridge in Amsterdam, installed in 2019

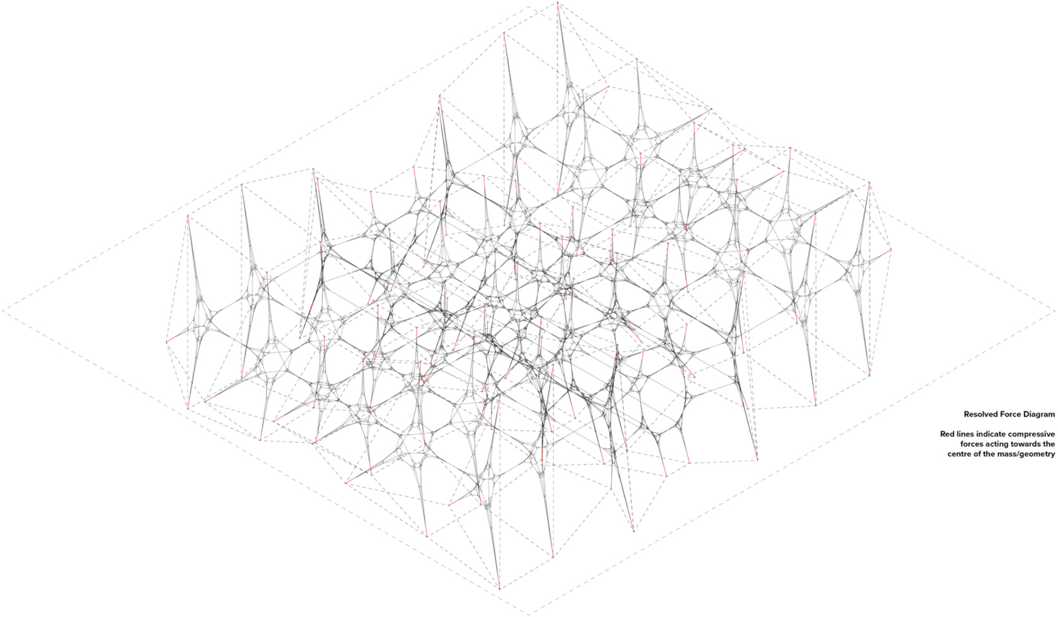


Subdivided Polyhedra



Unresolved Force Diagram

Red lines indicate compressive forces acting towards the centre of the mass/geometry



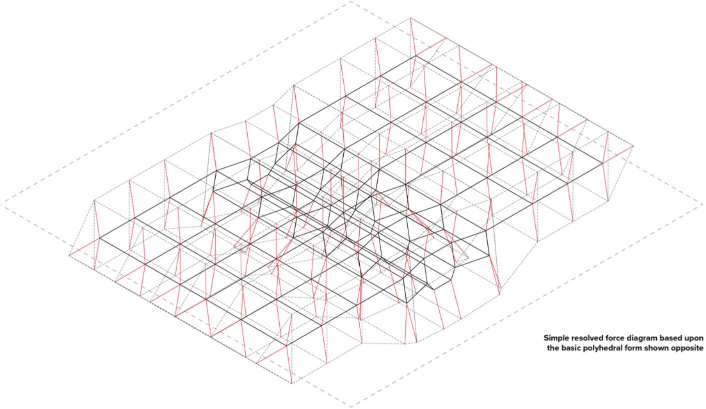
Resolved Force Diagram

Red lines indicate compressive forces acting towards the centre of the mass/geometry

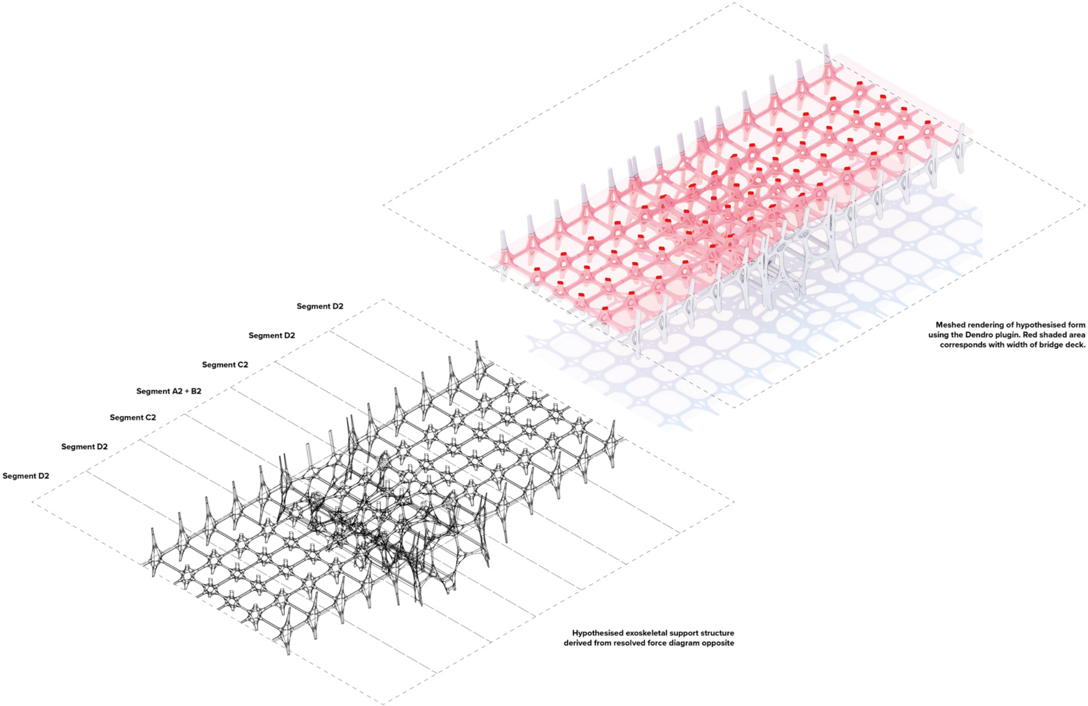
(officially opened in 2021 (Parkes, 2021)), the second was Striatus, developed by the Block Research Group in conjunction with ZHA & in3D (https://www.striatusbridge.com/), whilst third was OptiBridge, a printed concrete footbridge spanning 5 metres (OptiBridge: a topology optimized 3D-printed concrete bridge, s.d.). It is the technology developed by the team at Ghent – namely VoxelPrint (Vantyghem et al., 2021) & CobraPrint (Ooms et al., 2021), that I will be investigating in order to create a suitable theoretical model for 3D printing. A differentiating aspect between the Striatus process and the OptiBridge process was the difference in internal wall build. The Striatus process followed a fabrication method similar to PLA printing, in that it builds a wall out of hollow cells, however the OptiBridge process built solid internal walls out of a single layer of deposited concrete. This is reflected in the voxel forms developed through the VoxelPrint plugin, where my assumption is that a single voxel represents a single 'drop' of concrete filament.

A previous intent was to use these aforementioned plugins to provide an output file capable of being analysed using the finite element analysis programme Abaqus, an industry standard software developed by Dassault Systems. However, having spent a significant amount of time studying the software's basic functions, and associated tutorials, I am still at a loss as to how to properly utilise the platform to perform suitable simulated analysis of the model, especially under compression and tension.

On the following spread are photographs of a section of the bridge span resin printed at 1:125, demonstrating how the underside of the bridge deck would interact with the theorised column supports - which in this scenario would also be fabricated with 3D printing in mind, allowing the two forms to organically merge.



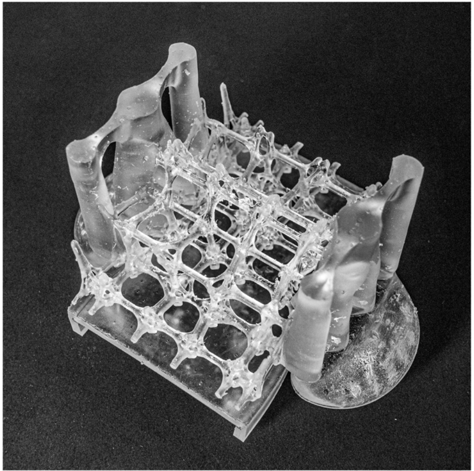
Simple resolved force diagram based upon the basic polyhedral form shown opposite



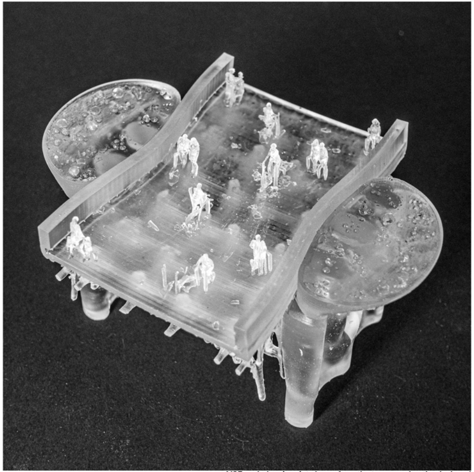
Meshed rendering of hypothesised form using the Dendro plugin. Red shaded area corresponds with width of bridge deck.

Hypothesised exoskeletal support structure derived from resolved force diagram opposite

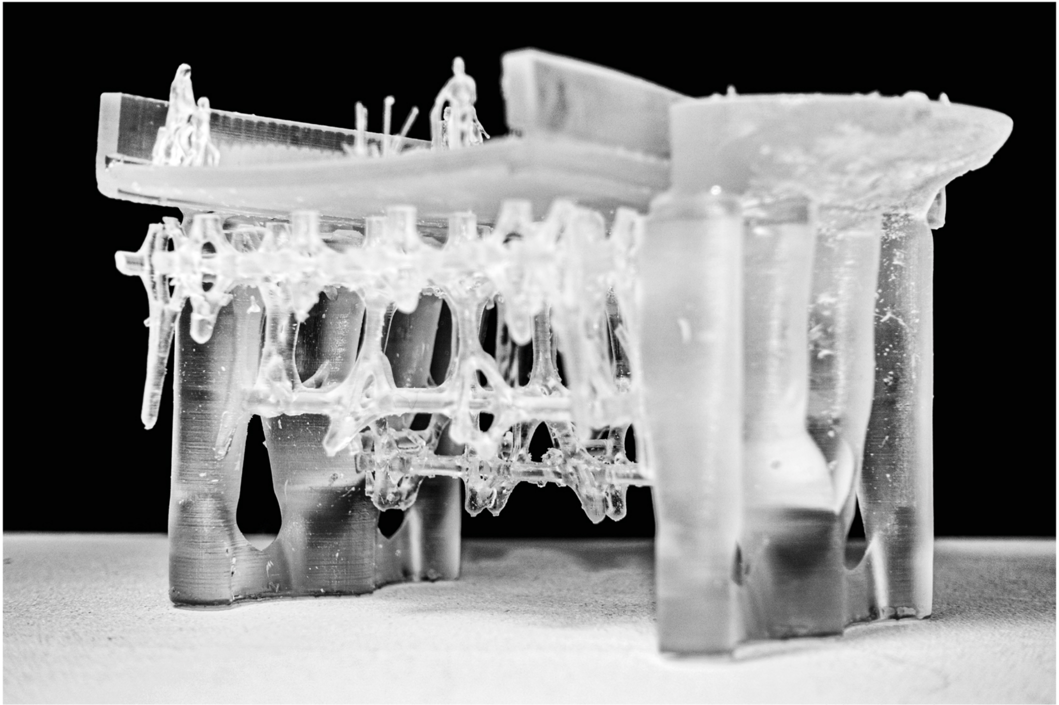
maquette of deck development



1:!25 scaled resin print showing the underside of theorised deck revealing the geometric form generated through PolyFrame (Dowson, 2022)



1:!25 scaled resin print shown from the top, revealng the deck and planters integrated into the column system (Dowson, 2022)



1:!25 scaled resin print shown in an angled view revealing both the geometric underside as well as the shape of the supporting columns (Dowson, 2022)

With any additive manufacturing method, if the design has any overhangs, bridging or severe deviations from the construction plane, the more complex the formwork or support structure will need to be. The Concre3DLab at the University of Ghent have already experimented with using a 'thixotropic support fluid' to increase the constructability of various designs, allowing severe overhangs and bridging by using a viscous support fluid that itself can be customised depending on the nature of the design being printed. All of this reduces the environmental impact of the construction method itself be reducing the amount of cement required and allowing kinder mixtures without the use of accelerants (3D concrete printing with a thixotropic support fluid, 2021). The Striatus project developed and demonstrates a new concrete printing technology – the ability to print non-uniform, non-parallel layers in one continuous print operation, which coupled with the design development of the bridge using the COMPAS AEC platform (built by the BRG), allowing for funicular form finding, shows another workflow from digital development to physical realisation using concrete (ETHZurich et al., 2022).

What I am seeking to demonstrate in this report is the beginnings of a working method that either an architecture student or architectural practitioner, that lacks a dedicated engineering background or training, may be able to accomplish and contribute to these nascent and rapidly developing avenues of inquiry. Other than these changed premises, the other adjustment is the incorporation of 3D graphic statics as a starting point for design development rather than 2D graphic statics, moving towards 3D printing a contiguous element as opposed to casting individual members in a more traditional method. In addition, the theorised scope is far larger, as I am proposing using concrete printing to cast segments of a far larger bridge span, whilst the reason this inquiry of mine remains wholly theoretical is due to a lack of access to concrete printing equipment.

As a starting point, to move from the digital representation developed thus far into a constructable outcome, I took the mesh output, merged all coplanar faces to reduce complexity, and then used the quad remesh command to further reduce complexity


Striatus Bridge Fabrication (in3D, 2019)


Striatus Bridge Assembly (Dell'Endice, 2019)


Striatus Bridge Fabrication (in3D, 2019)

OptiBridge Development (Vantyghem, 2022)

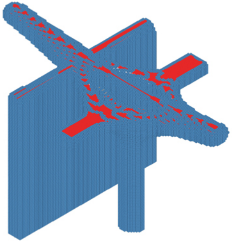
OptiBridge Deployment (Vantyghem, 2022)

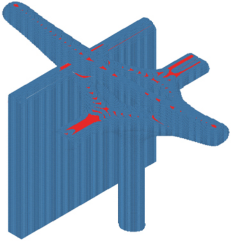
and convert the mesh into a SubD format, and then converted into a NURBS BREP. This process may seem convoluted, but it appeared to provide the best outcome for reducing complexity whilst ensuring the aesthetic was not unnecessarily compromised. From here, I set the print bead height to 10mm, and width to 50mm, based on the technical information provided by the Striatus bridge development, as a baseline for setting print parameters in CobraPrint (which in turn would generate the analysis file for Abaqus). Upon experimentation, it appears that CobraPrint itself is designed with a maximum print module height of 4000mm, and thus I have cut down the module for testing to 4000mm x 5000mm. Due to this limitation, the direction of print is sub-optimal, as it is running across the width rather than with the span, but as I do not know whether this limitation is due to the coding on CobraPrint's part, or an inherent limitation that the University of Ghent found with their own research, I am uncertain at this stage. Unhelpfully, the two research papers dealing with both CobraPrint and VoxelPrint are locked behind an Elsevier paywall which our university does not have access to. Due to the aforementioned limitation, and the severe computational bottleneck I encountered when running the plugin, I abandoned experimentation with CobraPrint in favour of VoxelPrint.

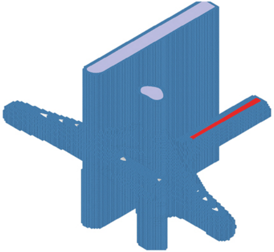
Building upon the Striatus' project parameters and the brief experimentation with CobraPrint, I started with the following parameters: voxel dimensions of 25mm3, bead width of 50mm and print layer height of 25mm (both height and width need to be multiples of the voxel specified). These parameters generated a baseline representation for 3D printing. However, the resolution was not fine enough, such that at certain points it would likely cause a breakdown in the printing process, and certainly cause structural concerns regardless (these are highlighted in red in the figures opposite). I increased the resolution of the voxel form by reducing the voxel dimensions to 12mm3, adjusted the print height to 24mm and print width to 60mm, however, this still caused structural gaps to appear in the voxel form. Attempting to rectify this issue, I rotated the print direction and set the parameters for a voxel of 25mm3, bead width of 100mm, layer height of 50mm, and then a fourth set with a voxel of 12mm3, bead width of 96mm and layer height of 12mm. The first of these did show a decrease in unwanted openings, however there were still a significant number, whereas the second showed a completely sealed unit, theoretically allowing for moving on to testing both in simulation and at scale. As with the CobraPrint plugin, the voxelisation process reiterated a key recurrence of an issue that has been dominant throughout this investigative series – that of computational bottlenecks.

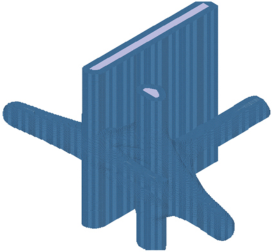
Computational bottlenecks have been hit at almost every major stage of this investigative process, from the initial experimentations with PolyFrame to meshing those iterations, and through developing my own force polyhedra to theorising the printable forms of segments of these concrete decks. These limitations have been frustrating, and I feel that they have frequently hampered my investigative speed and direction – for instance during the initial experimentations I stated that I originally wished to methodically iterate through 2,240 polyhedral forms, yet was limited by both the computational time and my lack of knowledge in automating procedures in Grasshopper. The second major hurdle was during the meshing stage where level of detail (LOD) was at first difficult to judge depending on the meshing method used – sometimes resulting in generation times in excess of 10 minutes producing meshes with polycounts upwards of 1x107, which would have to be reduced. Lastly, during the development of the deck supports based upon a customised polyhedral form, at the final iteration stage (where I tasked the computer to form the entirety of the geometry linking segments A2 through D2 together), this generated form would be produced successfully, but the file would fail to save, forcing me to export the form as a geometry-only file in order to continue. This perhaps demonstrates the current limitations of the PolyFrame plugin, at least in the hands of a novice investigator.

This brings me to my last point: the level of and progress in iterating this design to a logical architectural solution. As I look back through this document, and the processes detailed within, especially with a mind to starting to investigate the necessity


First VoxelPrint iteration w/ 25mm voxel


Second VoxelPrint iteration w/ 12mm voxel


Third, rotated, VoxelPrint iteration with 25mm voxel


Fourth, rotated, VoxelPrint iteration with 12mm voxel


MULTI & SINGLE ELEMENT
Expansion Joints - **Type 6**

ROLLER SHUTTER
Expansion Joints - **Special**

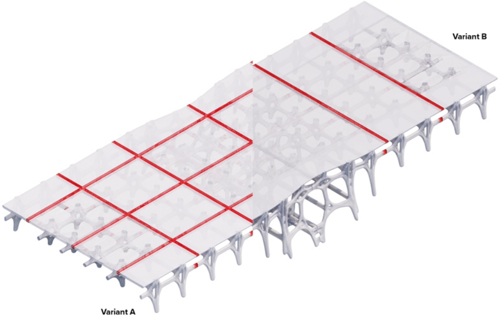
FINGER TYPE
Expansion Joints - **Type 7**

TF
Expansion Joints - **Type 5**

T-MAT TYPE
Expansion Joints - **Type 5**

SEAL TYPE
Expansion Joints - **Type 4**

Examples of existing movement joints for bridge construction provided by the EKSPAN group (EKSPAN, 2016)



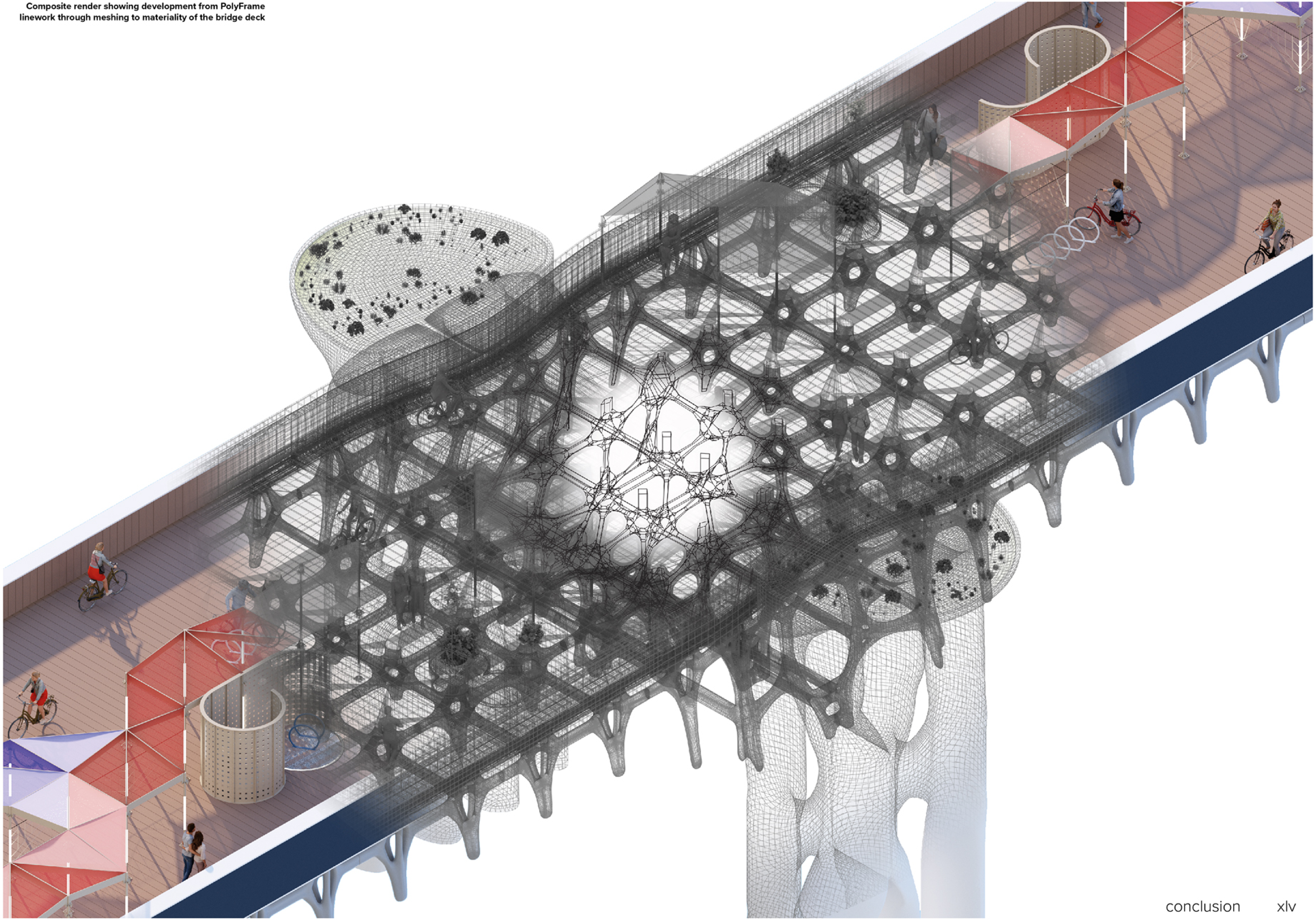
for movement and expansion joints between deck segments, I feel that this form would be better suited to that of a suspended cradle holding up the bridge deck, as opposed to a 'exoskeletal' form (as developed) that runs the length of the deck, acting as a spine held between columns at approximately 52500mm between column centres along the length of the deck. At this stage in the process, this is however a moot point, and thus I will turn my attention to a brief investigation of movement joints for this exoskeletal form that has been developed.

Large expanses of structure require allowances for physics, especially structure that is liable the effects of the stresses and strains of expansion and contraction due to temperature, along with being subjected to vibrations, both natural and artificial, during its lifetime. These come in the form of movement joints, which in traditional bridge construction often separates deck spans between column supports and/or between the ground plinth the bridge deck, depending on the overall bridge span, structure and engineering requirements. The proposed construction method here is very much in the experimental stage – as proved by the Striatus and OptiBridge projects previously mentioned, neither of which to my eye showed obvious movement joints incorporated into their structure – I suspect this is due to their relatively short span and temporary nature. A proposal of the kind I am imagining would require movement joints placed at appropriate points along the span, and I feel that this would already be facilitated by the nature of the printing method, as the ends of the printing segments could be plugged by a bespoke movement joint specifically designed for a printed bridge of this magnitude. What shape this would take would depend on whether printing technology advances such to allow for segments as voluminous as those proposed in Variant B, or whether only smaller segments as in Variant A would be possible. For the sake of argument, I have highlighted in red where potential areas requiring movement joints between the deck segments are located. Looking through existing examples of various movement joints, specifically those provided by EKSPAN (EKSPAN, 2016), elastomeric seals are recommended for pedestrian bridges.

conclusion

What I have achieved through this method of design development and technological investigation is a basic understanding of how graphic statics has been applied to three-dimensional computational form generation. From this basic comprehension, I managed to generate indicative column supports as well as the support structure for a deck, leading to resin and PLA prints at scales of 1:1000, 1:500, 1:125 and 1:50. The limitations in my knowledge stem from my lack of understanding of how to manipulate the PolyFrame simulation to provide tensional or compressional forms as opposed to simply compressional – or whether it is in fact providing those and the stipulation of tension/compression is down to the user defined external mathematics surrounding the generation of the particular force polyhedron/polyhedra. This in turn has limited my confidence in developing any particular form further, in addition to dissecting existing projects (for instance, the Saltatur research project) which coupled with being stymied by the limits of my own computational hardware has curtailed the level of design resolution I originally wished to achieve. However, despite these setbacks and limitations, what this investigation has provided me is an insight into physics-led or physics-conscious design methodology with immediate reciprocal feedback allowing for a relatively more rapid design response in the future (once my further desire for understanding of the background is satisfied). Whilst I have decided to focus on how these forms may be achieved through printed layer deposition methods, existing research has shown that this particular structural design method has a broad basis for material resolution in timber, steel, other forms of concrete, and no doubt there are more options that could be developed upon further research and experimentation.

Composite render showing development from PolyFrame linework through meshing to materiality of the bridge deck



- *3D concrete printing with a thixotropic support fluid* (2021) Directed by Concre3DLab Ghent At: https://www.youtube.com/watch?v=_P9dTiYntEc (Accessed 24/04/2022).
- *3D Graphic Statics: Polyhedron-Based and Vector-Based Approaches* (2020) Directed by IWSS2020 At: https://www.youtube.com/watch?v=Z6b_bx-B37w (Accessed 27/02/2022).
- Akbari, M. *et al.* (2019) 'From Polyhedral to Anticlastic Funicular Spatial Structures' p.10.
- Akbari, M. *et al.* (2020) 'Geometry-based structural form-finding to design architected cellular solids' In: *Symposium on Computational Fabrication*. SCF '20: Symposium on Computational Fabrication. Virtual Event USA: ACM. pp.1–11. At: https://dl.acm.org/doi/10.1145/3424630.3425419 (Accessed 18/04/2022).
- Akbari, M. *et al.* (2022) 'Strut-Based Cellular to Shellular Funicular Materials' In: *Advanced Functional Materials* At: https://onlinelibrary.wiley.com/doi/10.1002/adfm.202109725 (Accessed 21/02/2022).
- Akbarzadeh, M. (2016) *3D Graphical Statics Using Reciprocal Polyhedral Diagrams*. [application/pdf] ETH Zurich. At: http://hdl.handle.net/20.500.11850/183500 (Accessed 18/04/2022).
- Akbarzadeh, M. *et al.* (2016) 'Three-dimensional graphic statics: Initial explorations with polyhedral form and force diagrams' In: *International Journal of Space Structures* 31 (2–4) pp.217–226.
- Akbarzadeh, M. *et al.* (2017a) 'Hedracrete: Prefab, Funicular, Spatial Concrete' In: *ACADIA* (Disciplines + Disruption) pp.76–81.
- Akbarzadeh, M. *et al.* (2017b) 'Prefab, Concrete Polyhedral Frame: Materializing 3D Graphic Statics' p.11.
- Akbarzadeh, M. *et al.* (2020) 'Saltatur: Node-Based Assembly of Funicular Spatial Concrete' In: *ACADIA* (Distributed Proximities) p.7.
- Bhooshan, S. *et al.* (2020) 'Morph & Slerp: Shape description for 3D printing of concrete' In: *Symposium on Computational Fabrication*. SCF '20: Symposium on Computational Fabrication. Virtual Event USA: ACM. pp.1–10. At: https://dl.acm.org/doi/10.1145/3424630.3425413 (Accessed 21/02/2022).
- Block, P. (2009) *Exploring Three-dimensional Equilibrium*. Massachusetts Institute of Technology. At: https://www.block.arch.ethz.ch/brg/publication (Accessed 21/02/2022).
- Block, P. *et al.* (2020) 'Redefining structural art: strategis, necessities and opportunities' In: *The Structural Engineer* 01/2020 p.7.
- Block Research Group and Schwartz (2022) *eQUILIBRIUM*. At: https://block.arch.ethz.ch/eq/drawing (Accessed 06/03/2022).
- Bolhassani, M. *et al.* (2018) 'On Structural Behavior of a Funicular Concrete Polyhedral Frame Designed by 3D Graphic Statics' In: *Structures* 14 pp.56–68.
- EKSPAN (2016) *Expansion Joints & Seals*. Directed by EKSPAN (s.l.): EKSPAN LTD. At: https://www.ekspan.com/media/2837/expansion-joints-seals-brochure-issue-03.pdf (Accessed 08/05/2022).

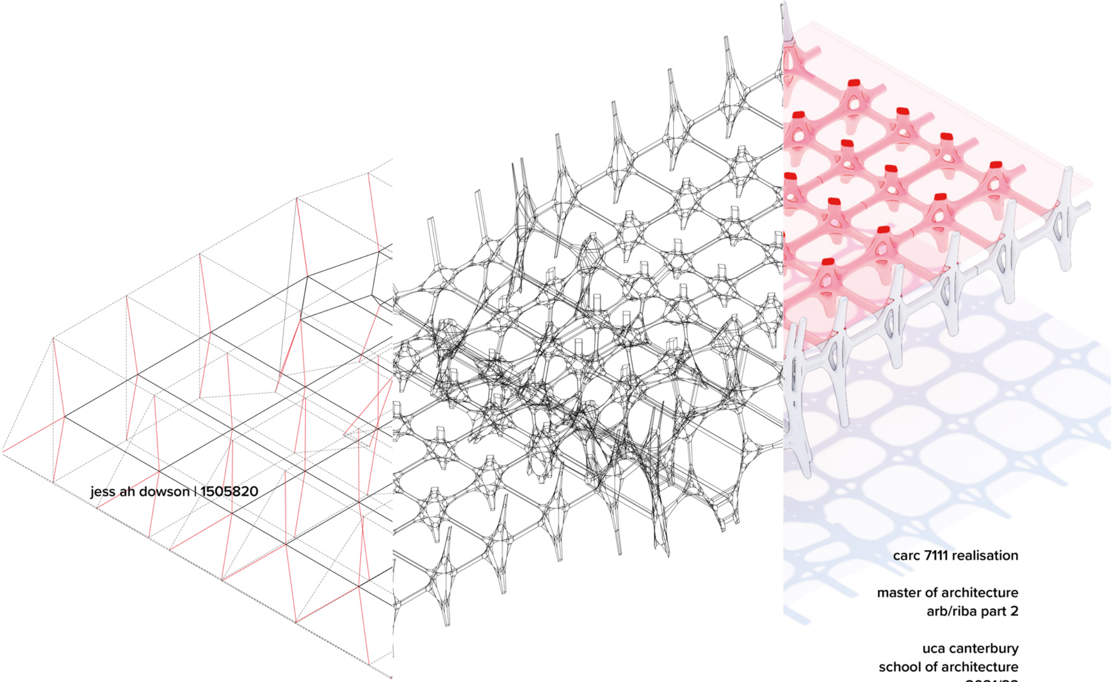
- EPFL (s.d.) *applications of graphic statics*. At: https://www.epfl.ch/labs/sxl/index-html/research/graphic-statics-applications/ (Accessed 27/02/2022).
- ETHZurich *et al.* (2022) *Striatus 3D concrete printed masonry bridge*. At: https://www.striatusbridge.com/ (Accessed 24/04/2022).
- *Form Finding using '3D GRAPHIC STATICS' Plugin For 'Grasshopper'* (2020) Directed by Dolatabadi, D. At: https://www.youtube.com/watch?v=lWiayzD8fp8 (Accessed 27/02/2022).
- Hablicsek, M. *et al.* (2019) 'Algebraic 3D graphic statics: Reciprocal constructions' In: *Computer-Aided Design* 108 pp.30–41.
- Lee, J. (2018) *Computational Design Framework for 3D Graphic Statics*. [application/pdf] ETH Zurich. At: http://hdl.handle.net/20.500.11850/331210 (Accessed 22/02/2022).
- Lee, J. *et al.* (2021) 'Geometry-based Teaching of Structures Through Computational Graphic Statics' In: *IASS* p.8.
- Liem, Y. (2011) *Graphic Statics in Funicular Design*. TU Delft. At: https://repository.tudelft.nl/islandora/object/uuid:1391866c-18f9-48ba-8c69-87ebd58e8516/datastream/OBJ/download (Accessed 06/03/2022).
- Liu, Y. *et al.* (2021) 'Kerf Bending + Zipper in Spatial Timber Tectonics' In: *ACADIA* p.9.
- Matthews, S. (2014) *Design of durable concrete structures*. Bracknell: IHS BRE Press.
- Nejur, A. and Akbarzadeh, M. (2021) 'PolyFrame, Efficient Computation for 3D Graphic Statics' In: *Computer-Aided Design* 134 pp.1–20.
- Omni* (2022) *Tension Calculator*. At: https://www.omnicalculator.com/physics/tension (Accessed 06/03/2022).
- Ooms, T. *et al.* (2021) 'A parametric modelling strategy for the numerical simulation of 3D concrete printing with complex geometries' In: *Additive Manufacturing* 38 p.101743.
- *OptiBridge: a topology optimized 3D-printed concrete bridge* (s.d.) [Collection] At: https://www.ugent.be/ea/structural-engineering/en/research/projects/all-projects/optibridge2.htm (Accessed 22/04/2022).
- Parkes, J. (2021) *Long-awaited 3D-printed stainless steel bridge opens in Amsterdam*. At: https://www.dezeen.com/2021/07/19/mx3d-3d-printed-bridge-stainless-steel-amsterdam/ (Accessed 22/04/2022).
- Vantyghem, G. *et al.* (2021) 'VoxelPrint: A Grasshopper plug-in for voxel-based numerical simulation of concrete printing' In: *Automation in Construction* 122 p.103469.
- White, Z. T. (s.d.) *V&A · Computers and the Sydney Opera House*. At: https://www.vam.ac.uk/articles/computers-and-the-sydney-opera-house (Accessed 21/02/2022).
- Zheng, H. *et al.* (2020) 'Machine learning assisted evaluations in structural design and construction' In: *Automation in Construction* 119 pp.1–17.

jess ah dowson | **1505820**

carc **7111** realisation

master of architecture
arb/riba part 2

uca canterbury
school of architecture
2021/22



jess ah dowson | 1505820

carc 7111 realisation

master of architecture
arb/riba part 2

uca canterbury
school of architecture
2021/22