UNDERSTANDING FLOOR PLAN OPTIMISATION: VISUALISING METRICS FOR MULTI-LEVEL RESIDENTIAL LAYOUTS

A. SAGUINSIN,
UNSW, Sydney, Australia
adrielsaguinsin7@gmail.com

Abstract. Spatial planning in the layout process of multi-level residences has underutilised automated means of generation and optimisation. An algorithm that produces multiple outputs for use in the decision making process streamlines the workflow and enables the extraction of metrics such as area, circulation ratios and structural wall lengths. This allows for the in-depth analysis of apartment layouts within the floor plate of a building on multiple levels. This research looks at the distribution of a variety of apartment types within multiple levels of the same building, to generate an efficient design without sacrificing requirements set by the design brief. Within the parametric modelling environment of Grasshopper, the algorithm allows for modification after the automated optimisation which enables flexibility through minimal manual control. The resulting algorithm optimises a multi-level residential layout based on existing data detailing positions of apartment types within a floor plate. In focusing on a multi-level residential building, flexible layouts can be defined that could change the way high-rise apartments are set up within the initial planning phase. In automating this stage of the spatial planning process, design exploration can be extended beyond manual modifications allowing for focus in other details of high-rise apartments. Combining the workflows of architects and engineers during the layout planning stage means that it structures collaboration into a smoother process. What would otherwise be a manual process extending over multiple disciplines can now become a single automated process to streamline the workflow, encourage collaboration and clearly visualise metrics for presentation to clients. Extending this research could explore other metrics that could further improve the optimisation of multi-level residential floor plans.

Keywords. Spatial planning; optimisation; floor plan; metrics; residential space.

1. Introduction

An important aspect of designing within the built environment is presenting the design to the client as well as the entire design team in a way that is clear and understandable. The presentation of information must be relevant and informative to listeners, especially in the spatial planning stage as that sets the crucial foundation for the development of an entire structure. The visualisation of metrics in conjunction with floor plan outputs displays relevant information for spatial planning and informs decision-making when all possibilities are presented. Optimising for metrics provides a unique form of generation that takes into account structural, design and project information to streamline inter-disciplinary collaboration. This rapid generation of iterations allows users to efficiently present a number of layouts with varying metric priorities and comparatively analyse each one to suit the requirements they set. As said in Hillier's 'Space is the Machine' (1998); "Architectural and urban design, both in their formal and spatial aspects, are seen as fundamentally configurational in that the way the parts are put together to form the whole is more important than any of the parts taken in isolation." which argues for the concept that parametric elements of architecture form a much more significant whole.

This research aims to critically analyse the spatial allocation and layout of high density residential floor plans on multiple levels. It will optimise floor plans by exploring relationships between spaces on the same plane as well as meeting standard requirements between levels in a multi-level residence. Residential layout generation like the work of Merrell's focused on a singular household and only output a visual model of the structure. Without information like metrics, the output of his work is only presentable on a visual level and goes no further than concept. Using visual scripting tools, this research will realise an algorithm that takes into account the requirements of Australian standards as well as the demands of the clients and the design team. Through the evaluation of the relationship between spaces, this research will begin to understand how spaces within floors work together and how each floor plate in a multi-level residence relates to each other. In a similar fashion to Das' space plan generation for commercial spaces, this research looks thoroughly into residential structures. The approach is entirely automated which enables an output to inform decision making. The rapid generation of Das' work expands upon the potential for iterative generation that presents an information dashboard which communicates information regarding the generated layouts. The exploration of the spatial relationships can reveal design solutions that would also optimise the design process, therefore making it more efficient. The resolution is to create a software tool that can generate a floor plan on multiple levels, output metrics and optimise each level while maintaining a relationship as set by the high-level requirements. Within the given time frame, the tool will be developed to be able to modify a generated floor plan and visualise metrics that can allow the design team to assess the spatial relationships and overall layout of each floor plate. This tool improves the design process involving architectural floor plans by automating the iterative process and visualising optimum design solutions based on varying metric priorities set by the user.

2. Research Aims and Objectives

In identifying the problem for which this research project aims to solve, clear objectives must be established as a guide for milestones to complete. These objectives also act as markers that can prove whether or not the research project is successful.

Automated space planning has long been researched with a range of methods. As mentioned in previous works, there are many different ways in which a floor plan for certain types of spaces can be generated. Depending on the space, the method for generation must suit the need of the spatial planning for that particular space type. But the industry has remained traditional in the sense that automation is still seen as a secondary means, especially within a multidisciplinary design team. A non-linear workflow in the spatial planning process results in a tedious back and forth between team members neglecting the efficiency of automation. In aiming to solve the need for a streamlined interdisciplinary workflow within the spatial planning stage of a building structures project, automation as a method was established. The objectives of this research project are to automate layout generation for multi-level residences using a visual scripting system within a parametric modelling environment. This automation will then enable the rapid generation of layouts, for the specific space type, which develops multiple iterations of a solution that meets the criteria set by the user. Since previous work in layout generation have not taken into consideration metrics that provide relevant information to both the multidisciplinary design team as well as the client, optimising for those metrics becomes a unique method that can be explored in developing a tool to generate and optimise floor plans. Using an evolutionary solver, these metrics that define the technical aspects of floor plans as well as the building as a whole, the entire algorithm can be purpose built to optimise specifically for those metrics. This information is then visualised in a dashboard and various views on a three-dimensional model, to present to the user. This visualisation in combination with rapid generation allows users to use the tool in informing their decision making. Since the tool iterates through multiple possible solutions during calculation, live visualisation that updates in conjunction with the process of the algorithm is necessary in providing a robust visualisation within an automated process.

3. Research Questions

Current means of layout generation are constrained by manually set requirements that only output the floor plan itself on a visual level. But from these outputs, metrics can be extracted that can assist in the decision-making process much more so than a drawing of a plan or three-dimensional render. In order to guide the process in obtaining an outcome suitable for inter-disciplinary collaboration in layout generation, details such as metrics must be considered and their relationship with the structure explored. Therefore the research aims to answer the following questions:

How can metric constraints based on existing multi-level residence spatial layouts be used to optimise an algorithmically generated floor plan in order to streamline the design process?

To what extent can this optimisation be used to minimise structural resources and cost?

In what ways can spatial planning optimisation improve upon existing layouts of residential spaces?

4. Methodology

The main methodology that this research will undertake is that of action research. Action research is a practice-based method which involves the experimentation with potential solutions to a design problem in collaboration with those involved in the progression of the topic of knowledge overall. The process of action research is a cyclical one in which the researcher plans, experiments, observes and then reflects at each stage of their process. This way, multiple iterations are explored and thus design solutions are reached effectively through a pseudo means of trial and error. In relation to this research project, an action research methodology will be used to iterate between stages in the script of a Grasshopper environment. The planning through the analysis of existing floor plans as a data set will determine the action of the algorithm with a set of constraints and guidelines. Through this process, optimisation occurs which can then be improved in future prototypes. This cycle repeats in such a way that each prototype is a development over the previous one, therefore these experimentations bring potential design solutions closer to fruition. The experimentation could also lead to newfound knowledge that could only be uncovered through the execution of a unique method, contributing to the collaboration with the topic as a social science.

Existing floor plans provide valuable data directly related to the relationship between spaces. The architectural considerations that support these floor plans are embedded in how the spaces are positioned in relation to each other. By taking these spaces and extrapolating how they are related to each other, a set of rules can be applied to an algorithm. With this algorithm, optimisation then starts to take place. The analysis of spaces visualises metrics for each differing layout which can then be used to inform in the decision-making process. This in turn optimises the spaces based on the extrapolated data in an automated process. With this method, this research can introduce a new process in the workflow of spatial planning. The efficiency in which spaces are placed within a bounding box could also reap financial and structural benefits, therefore increasing the need for some sort of computation integration into the generation and modification of floor plans.

5. Background Research

5.1. LITERATURE REVIEW

The spatial planning process involves methods of coordinating the distribution of people in active spaces at varying scales and can be that of a tedious one which does not fully take into account the requirements set by the demands of the design team and the client. Automating this process will reformulate the design process to be a much more efficient one and therefore clearly visualise required changes and details of significant importance. But in order to optimise a layout autonomously, a data set comprised of existing floor plans must be used as a basis to work from. This data set can be analysed further beyond the logistical reasoning behind its spatial allocation and can be evaluated for its social logic while considering high level requirements of a multi-level residence. Previous works that have explored the automation of spatial planning has only taken into account the needs of the client, which include room types and the number of rooms constrained within a boundary. The automated generation is also limited to a specific form, with little tolerance for the type of dwelling. Others have explored the highly regular layouts of commercial spaces which do not critically analyse the relationships between the spaces within a structure, but rather lay out the requirements almost randomly which still leaves a majority of the decision-making process to the user.

Most data-driven automated spatial allocation programs have free reign in some sort of capacity. Merrell, Schkufza and Koltun’s (2010) work in computer-generated residential building layouts was constrained only by the high level requirements of the client. Their architectural program was trained on a Bayesian network based on adjacencies between spaces as well as quantifiable metrics that dictate the form of the residence. However, the overall shape of its exterior as determined by the position of aligned spaces was left without boundaries. This resulted in an optimised layout that only took into account common spatial adjacencies and client demands. The work also focused on the computer-graphics side of things, putting priority on the visual appearance of a final three dimensional rendering rather than set limitations dictated by standard. Their work looked over was the myriad of real world factors that are taken into account by architects when design a layout for a residence (Merrell et al. 2010, p. 9). His idealised program optimises the layout of a specific building type, a standalone residence, and neglects specificities for various projects in its free reign. Although fit for a project with almost no limitations, the program ultimately leaves the majority of modifications to the designers and only generates a high number of options.

Anderson has taken a more commercial approach to automated spatial planning a developed an algorithm that procedurally generates desk layouts for a private office space. Although this work does not deal with the traditional scope of generating rooms within a boundary, its spatial planning is of key note as it is highly focused on constraints. Within a set space, their algorithm lays out desks with certain limitations regarding clearance rules as well as edge boundaries. This is important as it is a seminal foundation in optimising a space within the strict requirements. The algorithm is benchmarked against the layouts design by human architects and was proved to be better but was stricter in its layout (Anderson, Bailey, Heumann, and Davis 2018). This meant that there was little tolerance within the desk array that would normally be relaxed under the hand of a human architect, which results in standardised arrays with no room for negligible outliers. Although highly focused on efficiency, Anderson's work rarely shifts into a relaxed mindset when dealing with its constraints. Definite rules in the algorithm may only be guidelines in some cases regarding certain office spaces. The need for tolerance, even amongst standards, is important so that a layout feels natural. Although streamlining the spatial planning process results in a better workflow, the automation has to allow for tolerances in its generations.

An entirely automated approach to decision making in the architectural planning process seems to be the way forward, but the work of Das has left the majority of the decision making to the design team. While their system generates layouts autonomously, its focus on quantity and design scores leaves the crucial decisions to the client and design team (Das, Day, Hauck, Haymaker, and Davis 2016). Their work on rapid generation was built as an assistance tool rather than a primary one, which means that instead of focusing on any actual optimisation in the spatial planning process, their algorithm generates variations ad nauseam then outputs a design score based on pre-programmed metrics. While this approach adheres closely to constraints and high level requirements, the space plan generation side only gives options. But its strengths lie in the system's ability to analyse a space and critically evaluate it. This added metric output algorithm built into the system becomes much more impactful for certain design scenarios where the layout is already fixed. The spatial analysis side of Das' system, even if its focus is on a commercial structure, is helpful in deconstructing a layout and then optimising it. Data gathered from the system's analysis is useful in determining exactly how the spaces within fit together.

Previous works have been focused more so on commercial spaces rather than residential ones. But even systems capable of generating residential layouts stray from factors that are considered major in real-world architectural practice. This research will explore further into the sociological reasons behind room adjacencies to optimise and generate multi-level residential layouts. It will also look into adhering closely to constraints with slight tolerances, rather than setting definite rules for an algorithm. The mostly automated approach enabled by this system means that the spatial planning stage in the design process will be streamlined and the entire workflow becomes more efficient. Optimising a layout based on factors beyond simple adjacencies could change the way architectural space is generated.

5.2. BUILDING METRICS

Metrics is a broad word which encompasses a range of meaning dependant on the industry. In technical terms it is a system or standard of measurement. Perhaps the most known definition of 'metrics' is in business terms, a set of figures or statistics that measure results. This is directly related to its use in the built environment industry as metrics is a standard term that describes the information defining a structure. It is the data that makes up the technical or non-tangible aspects of a building which could be defined as a set of numbers representing a part.

Involving the use of these metrics in a system of automation begins with understanding that there are a wide range of metrics that can be extracted from a building which leads on to the simplification of the research. Within the scope of this project, the very basic or essential metrics regarding the definition of layouts within a building became the focus for visualisation. Since there are so many metrics to be defined throughout the development of this spatial planning automation tool, their organisation is important in understanding not only the level of detail they require, but also their place in the final display output.

5.2.1. Project Metrics

Project metrics can be classified as the information that a client would most likely focus on. These are the metrics that are almost completely dependent on the criteria of the client and are the direct result of either the initial design brief or factors involving information sourced from outside of the technical definition of the structure in a project. Numbers defining cost and value for example are a direct result of the calculation between the aspects of a building and an exterior factor that determines the expense in relation to the specific aspect of a building. A metric like the gross floor area (GFA) and gross leasable area (GLA) is connected to the overall site of the building and how the area within that boundary is utilised. These metrics are directly connected to affecting both the cost and value of the building overall. The GLA is also in direct relation with the apartment type distribution for the layout. A client's requirements for the types of apartments sitting within each floor plate of the building affects which areas are leasable and are again tied to the cost and value of the building overall.

5.2.2. Structural Metrics

In collaboration with Arup on this research project, the focus shifted heavily on the structural metrics of a building and how they affect the layout on each floor. A firm based around engineering disciplines find the value of structural information consequential, therefore making it a high priority in the visualisation of metrics for automated spatial planning.

Structural metrics define the relations between elements or parts of a building overall. It is the technical information describing the physical configuration of a structure and is defined by a set of numbers and figures. These range from basic information like building height and the areas of not only the entire building site but each of the included apartments within each floor plate. These two metrics are also in direct correlation with the wall span and floor depth of each apartment. The calculation of wall spans consists of the length of a single side of an apartment that spans the entire area. This determines how far apart each of the structural walls are for each space divided within the floor plate and extends vertically upwards to form structure for the building as a whole. The wall spans also determine the floor depth of the building which is the thickness of the concrete slab needed to support each floor. The variation in the depth of this slab affects the building height and can be calculated by:

$$f = w / 33 \quad (1)$$

Where f is the floor depth and w is the wall span.

6. Case Study

In creating an algorithm that generates a floor plan for multi-level residences and outputs metrics, the basis of the entire script lies on being able to generate a feasible layout and extract information from it. The automation of the spatial planning process opens itself up to a myriad of possibilities in terms of how it generates a floor plan, based on the requirements and constraints set by the user. Constraining the layout to a set of specific rules means that it would generate very differently if given a completely different set of rules. After specifying exactly how the algorithm is set up to generate a layout and output metrics, iterations of the algorithm were developed in order to explore the possibilities of its varying outputs.

6.1. DESIGN ITERATION 1

The original concept for this research project involved a smaller scale, working with rooms and spaces within a single apartment to optimise for certain metrics. This idea optimised spaces for structural efficiency but later revealed how limited the metric visualisation would be on such a small scale. Developing the algorithm parametrically within Grasshopper meant that the entire project could be scaled up to deal with multiple apartment types on a single floor plate. Optimisation on this scale would work exactly the same but had more opportunity for the extraction of metrics, especially in relation to the building as a whole. In this way, more information can be outputted and multiple iterations of an entire floor plate could prove more useful in the architectural and engineering industries.

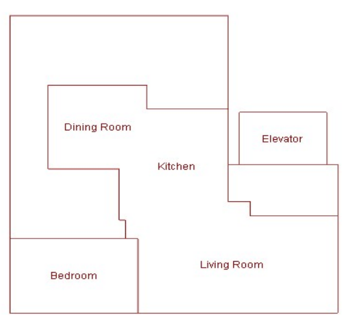

Figure 1: Preliminary Concept.

6.1.1. Floor Plan Optimisation on a Building Scale

An obvious approach to generating a floor plan in a parametric environment, using a visual scripting software like Grasshopper, is to generate rectangles and reposition them using constrained minimisation within the evolutionary solver. Galapagos can then be connected to the parameters of each generated rectangle and modify them by minimising the distance between them and the difference in area as set by Australian standards.

The benefit of a parametric environment is that everything can always be modified later. With this in mind a rectangular site was created to act as a boundary for constraining the generated rectangles representing apartment types. This border would be the floor plate and the rectangles within can re-arrange themselves to meet basic constraints. To start off with, the algorithm calculated the area of each rectangle, constrained them to a minimum area depending on the apartment type and then re-oriented them to create a cluster of boxes that fit within the floor plate. These constraints are the most basic requirements for generating a layout automatically and in theory ensured that each generated rectangle represented an apartment type accurately and clearly. In addition to position and size constraints, the algorithm was also developed to constantly test for collisions between rectangles. This ensured that the apartments never overlaid each other and therefore a feasible layout can be generated.

The immediate observation from this was the amount of dead space that the algorithm generated as a result of loopholes within the constraints. Although the positioning and sizing had almost a free reign within the boundary of the floor plate, collision testing resulted in the rectangles being position rather far apart from each other. Although this could easily be solved by generating more apartments to fill the entire floor plate, the limited sizing parameters did not allow the rectangles to piece together into a legitimate layout.

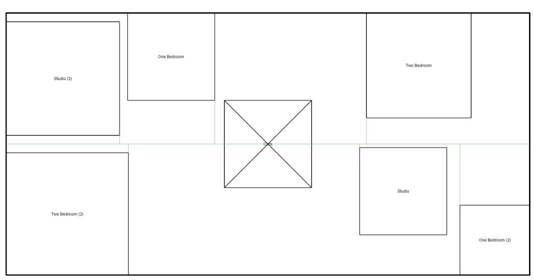

Figure 2: Building-Scale Layout Generation.

6.2. DESIGN ITERATION 2

Limited by the rectangle parameters from the first iteration, the second iteration of the algorithm sought to tighten the floor plan by expanding upon the sizing parameters as well as tightening the boundary of the floor plate. This improves upon the generation process by producing a more feasible layout that is constrained more strictly but flexible in its parameters.

What this creates are apartments that are much more fluid in its shape which results in a layout that is pieced together more closely like a puzzle. Adding in separate parameters for each axis of the shape as well as rotational components leads the solver to a solution that pieces together the apartments rather than repositions them on the edge of the floor plate boundary. From this point the tolerance for a cohesive layout is quite minimal and therefore the metrics can start to be layered on. Information like the areas, apartment type distribution and number of rooms are straightforward and can be presented on the layout. This way, the live updating layout can clearly show exactly where each apartment is on the floor plate, its size as well as how it relates to the overall floor plan.

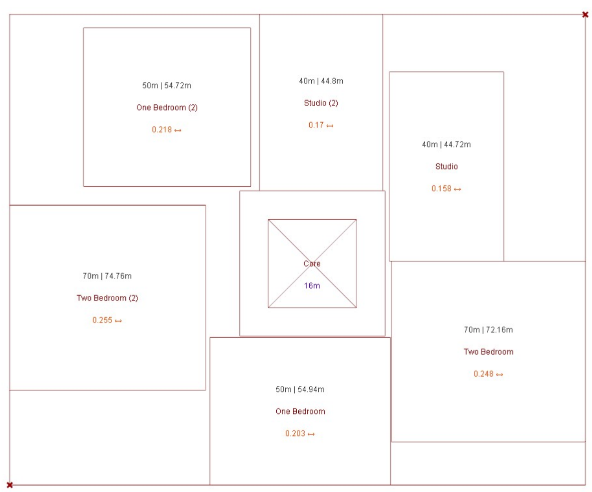

Figure 3: Design Iteration 2.

6.2.1. Metric Priority

An observation from the testing of this iteration found that the evolutionary solver could not quite get to a final solution. This was because each of the metrics layered on, which although basic, had equal weighting. This means that the solver was optimising for each of them equally and never arrived at an optimised solution.

In order to solve this problem, metric priorities were added by introducing penalties and incentives to the script. This involves multiplying each of the calculated metrics with a weight number, set by the user, which would then list those metrics in a certain priority order. Each of the resulting calculations were then fed into the fitness number for the evolutionary solver and formed an orderly list of metrics with differing weight values as the target. The evolutionary solver is then able to optimise for each of the metrics in a list order which avoided clashing within the script and enabled proper solutions to be discovered.

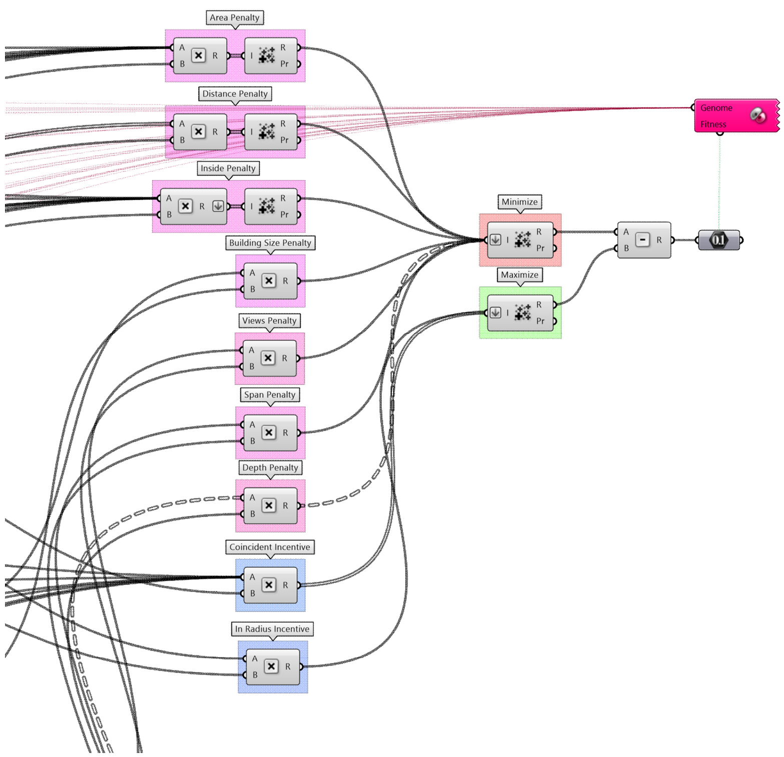

Figure 4: Metric Priority (Penalties/Incentives).

6.3. DESIGN ITERATION 3

Having spent a considerable amount of time listing off metrics to include within the information outputted from the algorithm, it could not be truly accurate without a sufficient method for generating layouts. Although dead space was minimised in previous iterations, it was still not enough to generate a working layout that could legitimately move forward in the design process. The method of generating rectangles greatly simplified the extraction of metrics but also severely limited the way in which the algorithm could generate layouts. Even with a manual change of parameters, results showed that the layouts generated remained fairly similar and could therefore not sufficiently inform in decision-making.

With such a method proving to be quite limiting, even with an overabundance of parameters and constraints, it was best to move onto a completely different way of generating separate spaces within a boundary. Having established a core in the building within a parametric environment, means that everything else can be modelled around it. This results in simple curve extensions that connect from the core to the boundary of the floor plate. Similar to Jared Tarbell's (2003) substrate algorithm, this method uses lines to create boxes which will then represent the apartments on a floor plate. Using lines instead of rectangles multiplies the possible shapes that a single room could be and is therefore a more sufficient method for dividing up a floor plate.

At this point, dead space within the layout has been completely eliminated and the algorithm creates space efficient layouts that can be layered onto with metrics. The division of the floor plate allows for each apartment to be treated separately as a surface and extracted for visualisation. This division was made efficient because of the equally divided area of the entire floor plate. Lines extending from the core utilised the whole area within the boundary which resulted in an even division between the spaces created between each generated line.

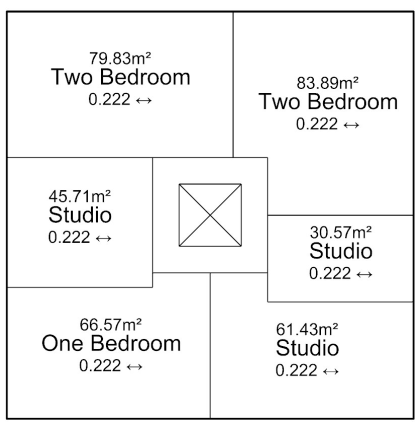

Figure 5: Design Iteration 3.

6.3.1. Metrics Dashboard

During the scripting of the layout generation, the components within the visual scripting software were able to output information about the calculation going on inside of each part of the script. These outputs can form the basis for the data that is used to calculate each of the metrics described previously and visualised on screen along with the model resulting from the generation. These outputs are important as they are what the evolutionary solver is optimising for and therefore determine the ways in which the algorithm gets to a solution.

Using the knowledge gained from background research on the calculation of each metric, the component outputs can be taken advantage of by connecting to a part of the script dedicated to equations to calculated all of the listed metrics. This then in turn connects to a three-dimensional text tag system within the script that visualises these results in an organised dashboard. What this allows is a completely parametric set up for the metrics that intertwine with user input enabled by sliders in the beginning of the script. So as the evolutionary solver rapidly iterates through varying solutions for the given criteria, or the user manually uses the sliders to modify the layouts, the changes are reflected instantaneously in both the three-dimensional model and plan view of the building as well as the dashboard containing the numbers and figures describing the metrics.

Apartment Type	Studio	One Bedroom	Two Bedroom
Amount | Distribution	10 | 16.7%	30 | 50%	20 | 33.3%
Area | Efficiency	30.9m^2 | -22.9%	32.5m^2 | 59.7% 40.9m^2 | -35.1% 79.8m^2 | +-18.3%	83.9m^2 | +19.8% 100.1m^2 | +43%
Ventilation Score	Average Ventilation	Average Ventilation Good Ventilation Good Ventilation	Good Ventilation Good Ventilation
Aspect Ratio	1.67	1.58 1.39 1.39	1.63 1.79
Views	No View	No View	View
Estimate Value	~ $552618	~ $589370 ~ $741948 ~ $1449678	~ $1567196 ~ $1870362

Height	GLA	GFA	Floor Depth	Potential Value	Estimated Cost
32.29m	3680m^2	4000m^2	0.229 ↔	~ $6771171.5	~ $1064426

Figure 6: Visualised Dashboard.

6.4. FINAL ITERATION

Developing upon the working iteration, finalisations were done in the visualisation of the tool. All the technical aspects of the tool proved to be working successfully and showed that it was able to generate a layout, optimise that layout based on set criteria and display metrics as its output. But in order for the tool to be clear about its about, many design choices were made to improve upon the look of the output. Since the tool outputs three main views; the three-dimensional model, the plan view and the dashboard, it was important that they all matched up in terms of design. It was also essential that the information presented as the output was clear, readable and actually able to update as the evolutionary solver worked through iterations.

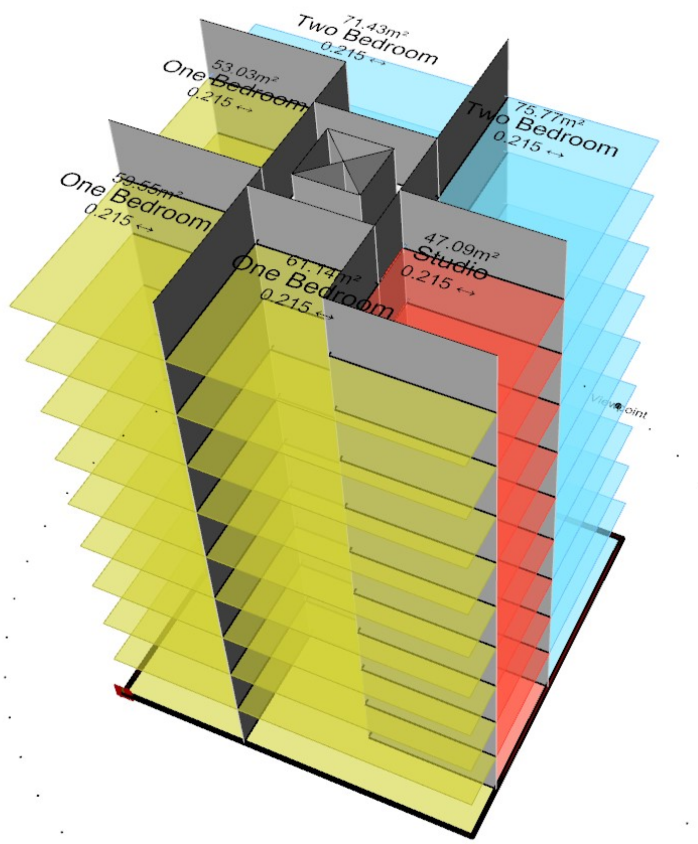

Figure 7: Finalised Model Visualisation.

7. Significance of Research

This paper concludes that the automatic optimisation of floor plans, specifically for multi-level residences, in combination with an extraction and visualisation of metrics streamlines the workflow of multiple disciplines and presents a clear dashboard of information relevant in the decision-making process with the client. Automation in spatial planning using an evolutionary solver means that multiple iterations can be generated with a varying set of metric priorities. Analysis of these iterations leads to an efficient workflow in making informed decisions and allows users to manually set the constraints of the algorithm to generate desired layouts.

8. Evaluation of research project

Spatial planning requires clear and direct communication with a multi-disciplinary design team as well as the client to achieve a desired solution. In creating an algorithm that can output a range of layouts with differing metric priorities, many options can be generated and analysed to inform decision making. Through the process of building up the algorithm, individual metrics were layered on and gradually added detail to a basic layout. In considering metrics like structural wall spans, floor depth, views and cost; they act as constraints and parameters that guide the algorithm into generating a desired floor plan output. Now with these metrics layered on, floor plans start to take shape by solving for particular metrics and finding the solution that best fits within the set constraints. This is different in the way previous works like Merrell, Schkufza and Koltun’s constrain their layout generations as this research works with the information that becomes part of the output of the algorithm. In this way, an extensive output becomes much more valuable in informing the decision-making process and contributes to the more technical side of spatial planning. The visualising of metrics as part of the output clearly displays vital information for both the client and the interdisciplinary design team.

Adding in a measure of priority with penalties and incentives, allows the algorithm to be easily modified to output layouts meeting the requirements of a design brief. Multipliers recalculate the importance of each individual layer of metrics so that the algorithm solves for the desired ones as needed. In this way, a visualisation of the details in each outputted layout occurs which contributes greatly to making an informed decision. Through this means of prioritisation, the designer and the client can also look at a wide range of layouts and outline the benefits and drawbacks of each. This streamlines the workflow of multiple disciplines and allows communication between client and designer(s) to be clearer. The visualisation of metrics lays out crucial information needed in the decision making process and can therefore be expanded upon to include further details that make up a multi-level residential floor plan.

This research, however, is limited by the time frame which resulted in a faster means of working which neglects detailed experimentation of all methods of generation and optimisation. The use of Galapagos as an evolutionary solver for example was not the only option in the means of optimisation. With a wide range of evolutionary solvers available, the working time frame was simply not long enough to be able to identify, learn and test how these other evolutionary solvers work as well as how they affect the optimisation of the same layout. Another limitation included the simplification of the control site for generation and optimisation. Working with a rectangular site highly simplified the means of generating spaces within a boundary and, although parametric, the tool's resulting capabilities were limited to only a certain shape of building. In addition to this, the methods for dividing the floor plate could have also been explored in depth, with this research only settling on the first solution that produced a competent layout. The means for generation are bountiful which means that further research and experimentation within the aspect of generation may have resulted in a more efficient, parametric and robust way of generating apartments within a floor plate. The visualisation of metrics was done quite clearly however, the level of detail and complexity required for a number of them was not up to standard in the real world. With a limited knowledge in engineering, accurate and detailed calculations for some of the structural metrics were not able to be done which resulted in a highly simplified display of structural information that lacked accuracy.

In terms of further development, all aspects of this research could be expanded to provide more detail. The benefit of working within a parametric environment is that every element is easily modifiable which makes expansion straightforward. Although the tool in its current stage generates and optimises a layout on a building scale, working with apartments as the spaces, it could also be layered on with an optimisation of the spaces within each apartment. In this way, generation and optimisation can occur on three levels; a building scale, an apartment scale and a room scale. Parametricism could also be improved to not be limited to a certain building type which was a direct result of the given time frame. Flexibility within the tool would prove beneficial for rapidly iterating between layouts and visualising the differences through the set of metrics displayed.

This research proved quite successful, however, in developing a tool that was able to automate spatial planning and take into consideration metrics as part of its visualisation. The resulting streamlined interdisciplinary workflow allows both the project team and the client to collaborate on the evaluation of layouts and use the tool to inform decision making.

9. Conclusion

This research shows that automated spatial planning can streamline the interdisciplinary workflow within a project team. Automation enables rapid generation which allows multiple disciplines to understand and make informed decisions based on the varying options laid out before them. The visualisation of metrics relating to the generated layouts displays live changes as iteration occurs which provides useful information in making the workflow for spatial planning more efficient. The combination of all these elements within a parametric environment enables a comparative analysis between iterations which in turn informs decision making and employs an automated tool for spatial planning resulting in an efficient system of generation and optimisation.

Acknowledgements

I would firstly like to thank my tutors Nicole Gardner, M. Hank Hauesler and Alessandra Fabbri for their time, experience and valuable feedback in working with me through this project.

I would also like to thank Arup, who collaborated with me as an industry partner, but in particular David Madden for their guidance, experience and assistance in developing this project.

References

Anderson, C; Bailey, C; Heumann, A; Davis, D; 2018, ‘Augmented space planning: Using procedural generation to automate desk layouts’, International Journal of Architectural Computing, Vol. 16(2), pp. 164-177, accessed 3 August 2018 from Sage Journals Online.

City of Sydney 2017, accessed 20 August 2018, <https://www.cityofsydney.nsw.gov.au/development/development-applications>

Das, S; Day, C; Hauck, A; Haymaker, J; Davis, D; 2016, ‘Space Plan Generator’, Acadia, accessed 3 August 2018 from CumInCAD.

Ghazi Al Ali Architect 2011, *SEPP 65 Apartment Design Guide (ADG) Compliance Table*, Homebush.

Hillier, B and Hanson, J.: 1984, The social logic of space, Cambridge: Cambridge University Press.

Hillier B, 1998, Space is the Machine: A Configurational Theory of Architecture, Cambridge University Press.

Jared Tarbell 2003, Complexification, accessed 4 August 2018, <http://www.complexification.net/gallery/machines/substrate/>

Merrell, P; Schkufza, E; Koltun, V; 2010, ‘Computer-Generated Residential Building Layouts’, Stanford University, accessed 3 August 2018 from Stanford University Database.

Richard Schaffranek 2012, *Floorplan Generator*, online video, accessed 24 July 2018, <https://vimeo.com/37186522>

Richard Schaffranek 2015, *Floor Plan Generation*, online video, accessed 24 July 2018, <https://vimeo.com/136231619 >

Rider Levett Bucknall 2018, RLB Intelligence, accessed 27 September 2018, <http://www.rlbintelligence.com/>

Simon, J, 2018, accessed 7 August 2018, <http://joelsimon.net/evo_floorplans.html>