DESIGN TO OPTIMIZED

Ensuring Multi Residential Buildings meet Solar Access Compliance through Genetic Algorithms

A. SAGUINSIN,
University of New South Wales, Sydney, Australia
jeremysaguinsin@gmail.com

Abstract. Solar access is a key factor when creating residential buildings, with regulations requiring at least 2 hours of sunlight for 70% of the apartments. This regulation is not only in favour of energy saving intentions, but for the psychological wellness of the residents. Though the current process of attaining the information is quite tedious, requiring almost an hour's worth of work to be done effectively. Furthermore, the fixes required to achieve an optimal building plan is more of a gruelling task, described to be a matter of a controlled trial and error. The research investigates the required regulations for a building in relation with solar access to analyse the feasibility of a multi-residential building, as well as generating or re-imagining given basic building plans. By using computational methods such as Grasshopper and its various plugins the current manual processes can be automated, establishing an efficient workflow within the design process.

Keywords. Automated analysis, Solar analysis, Automation, Optimization, Multi-residential buildings, Workflow optimization, Genetic Algorithm

1. Introduction:

Within the past and the coming years, the has been an ever-growing need of residential structures is occurring, with the demand within 2009 to 2015 accounting for “one-third of all residential building approvals” (Shoory 2016 p 19). Due to this high demand, an adequate sum of the AEC (Architectural Engineering Construction) industries have focused their work on developing these buildings, resulting in the high demand for validating its compliance. As of now the current methods of analyzing and optimizing a building form has perpetually been a tedious task of evaluating and re-evaluating, which are done manually within some companies, resulting in limited accuracy of readings and diminished available time. However, with the introduction of computational methods and tools, most of previous problems stated can be alleviated, through the automation of processing and improving analysis accuracy.

The project tackles the situation around solar access within multi-residential buildings, according to the NSW Design Guide it states that 70% of the apartments within a building situated in Sydney’s Metropolitan Area, must have a minimum of 2 hours’ worth of sunlight in both the living room and private open spaces. This regulation is not only in favour of energy saving intentions, but for the psychological wellness of the residents, to “feel healthy, people need appropriate visual contact with the external world” (Altomonte 2008 p.4).

The current method across the AEC industry, of checking the compliance of a building is conducted towards the building façade, with the amount of light entering a building not being considered. This project will aim to implement a workflow that will improve the accuracy of analysis through the consideration of direct sunlight inside a building, and the use of genetic algorithms to optimize buildings for solar access. This will be done through the computational tool Rhino and Grasshopper, with an overarching goal of automating the process.

2. Research Aims

Given that the project is set within a ten-week time frame, realistic goals must be set. These goals centre around the notion of developing a workflow for analysing and optimizing a building form for solar access.

- Develop a Grasshopper script that can convert Revit buildings into identifiable apartment rooms that grasshopper is able to read.

- To develop a Grasshopper script that can accurately and reliably analyse a multi-residential building for its solar access.

- Successfully create an apartment generator script that will try to generate a building with similar Gross Floor Area or Volume, whilst optimizing it for solar access.

3. Research Questions

As stated previously this research aims to improve process of optimizing buildings for solar access through the utilization of genetic algorithms. Though it will not solely centre around the area of optimization but also explores ways on improving digital compliance checks for solar access. Thus, two questions were developed to provide goals within completing the project.

- How do we improve the current analysis methods of solar access analysis through computational means?

- In what ways will genetic algorithm improve the workflow for optimizing building forms for solar access?

.

4. Methodology

The process outlined can be referred to the iterative process of Action Research (AR), with an aim of defining the problems, then proposing and testing appropriate solutions, which leads to an overarching goal of contributing legible information and knowledge within the situated field. The AR process can be broken down in to four main headings; Diagnosing, Action planning, Action taking and Evaluating. Within Diagnosis a primary problem is clearly outlined, in this case the problem outlined is the inefficient ways of conducting analysis and optimization for improved solar access. Thus, what is needed is the implementation of an improved workflow through the introduction of computational methods. This leads us into the process of both action planning and taking, involving the implementation and experimentation of different proposed methods. The methods proposed for analysis are through Grasshopper and its plugins Human and Ladybug. Human will be used to automate the building detection within a Rhino file, assessing which layers within the file contains the needed geometry. Grasshopper will take the collected building geometry and begin rationalizing the form to a 2D simplified state, where each apartment can be labelled. Finally, the solar analysis will be handled through Ladybug, by providing legible and appropriate meshes, location and start date and end date of winter solstice an accurate analysis of the building form can be generated. What will be tested is the different ways a building can be analysed for their solar access, which will evidently lead to a final solution. As for the optimization process, a tool will be developed which will try to generate different building masses with similar Gross Floor Area and determine which proposed building will yield the best results. The optimization tool will be using a further simplified model as to increase processing speed, the model will again be analysed through Ladybug and optimized using Galapagos or Octopus and its Evolutionary Solvers. Evolutionary Solver (ES) utilizes the groundwork of Darwin's Theory of Evolution, survival of the fittest. With reference to Grasshopper, the ES will generate different building designs through a Grasshopper script and test them based on a dynamic fitness value generated from both Ladybug and Grasshopper i.e. amount of rooms within a building that meets regulations and total GFA of each building. Finally, by evaluating the validity of the different proposed analysis and optimization methods, then choosing the methods that yield the best results a preliminary workflow can be created, that will hopefully reduce the total time taken.

5. Background Research

Automation is an essential aspect within the computational design paradigm, whether it be in construction, design, optimization or analysis. Analysis is one of the most needed and desired to be automated within design, due to the current process being time consuming and inaccurate. The overlying aim to automate the analysis process is to create an efficient "feedback cycle", where the designer is continuously informed throughout the design process, allowing them to make changes during the design process (Abdelmohsen, Eastman, Lee 2011 p 403). In this research the aim is to more specifically investigate how to automate the procedure of direct sunlight incident on a multi-residential building, to check if a building meets the requirements of NSW Apartment Design guide. Though if the building does not meet these requirements, optimization is required.

Why is daylight access so important for building regulations? Daylight access is defined as the amount of available light incident on buildings private and open spaces causing a certain amount of illumination (De Kay 1992 p 131 – 138). This analysis will determine the height restrictions of buildings and zoning lows for areas. With the continuous progression of buildings and streets becoming denser in urban areas, regulations are becoming stricter as setbacks are now being pushed. Though this is not the sole reason for restrictions, a study by Sergio Almonte explores the effects of natural light within a building on the human psyche. The feeling of comfort and well-being has been scientifically proven to coincide with a room's lighting condition and the "very perception of the environment that surrounds us" (Altomonte 2008 p 4). As such the positive feeling is not only contributed by the very presence of light but our visual perspective of the outside world. A balance between building regulations and human psyche is required to design an optimum building.

The detail of a model can influence the accuracy of sun analysis, a less detailed model provides a less accurate sun analysis. This can be referenced to the current method used to undertake this process, using simple massing for each apartment room and checking the light incident on the outer facades. Though reliable to an extent, its accuracy is diminished since external items that could affect the sunlight will not be considered. These analysis tools usually require some sort of computational experience as the complexity of the model will require some sort of adjusting to the program to become accurate. A paper on sun light on complex geometry outlines that not only should light incident on a building be considered but also the angle of incidence (da Veiga, La Roche 2002 p 105-109). The software used is a web-based analysis tool that considers location of the object and the time of day, thus being able to calculate which areas are taking in sunlight. Unfortunately, the software used is very much outdated due to its limited analysis readings, and requiring computational knowledge to be functional, though this does show the possibility of solar access being using on much more complex geometry.

Ladybug is a plugin for Rhino/Grasshopper that is currently being used today to assess the environmental impacts towards a building. By using computational methods, one can assess the validity of a model within its given location, the plugin can provide analysis on building energy usage, internal temperature, lighting analysis (Roudsari, Pak, Smith, Gill 2013 p 3128-3135) etc. Though for this case, the focus of the plugin will revolve around sun analysis. The plugin is commonly used for daylight and basic energy efficiency analysis for simple building models. In addition to this, due to the fast capabilities that ladybug provides optimization techniques such as evolutionary solvers become possible. An approach for this could be to generate multiple iterations of massing assessing which are most applicable to building regulations and how much the generated massing strayed off from the limiting factors. De Luca has done a similar process though his process involved random patterns which were assessed within an urban environment. A problem with this method is that too many variables are changing, such as such as height, volume, shape and orientation. In addition, the generated building massing's are not considering the validity of living spaces as well as its effect on surrounding buildings (De Luca 2017 p 439). Though is generating new designs reliable, granted it takes many iterations to find optimal designs and its validity will always vary, but what if we just change the existing massing to a much more organic looking geometry. Research was done using such a method, by increasing the number of visible surfaces seen by the sun thus creating a much more complex shape (Giani, Belfiore, Lobaccaro, Masera 2013). This showed significant changes, improving the overall solar access of the buildings, furthermore, causing little to no changes towards the surrounding buildings. This method falls within the same problem of De Luca's method, livable space. Due to the complex shapes this method could create there could be a significant drop of livable spaces available.

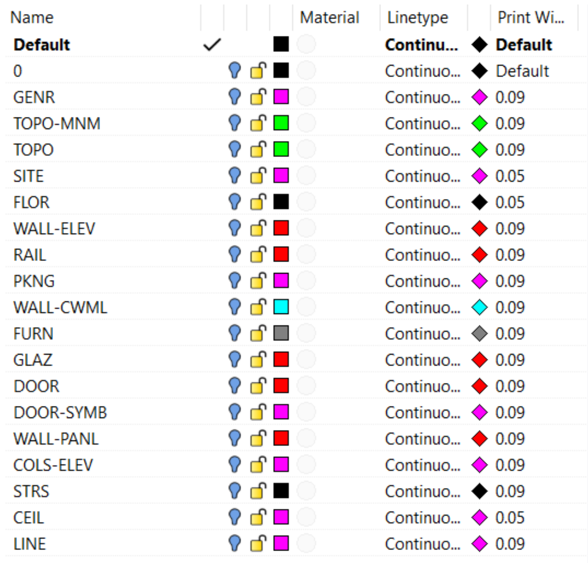
There is a significance of using both analysis and optimization methods to improve buildings, though there is always factor that is left astray, the amount of livable spaces within the generated or changed massing. Granted they improve the overall solar access of buildings, but where within those buildings is the sun getting access too. In addition, the output information for these analysis methods are presented as pictures on a screen, visually they convey a general scope of problem areas, though how much of that can be beneficial. Data output and analysis accuracy/validation of buildings are areas that could be improved upon. The building analysis and optimization methods presented provide a good starting line on what is currently usable within industry.

6. Case Study

The research explores methods of analysing and optimizing a multi-residential building for solar access through the means of computational tools such as Grasshopper. The process will involve converting a pre-existing Revit model into a legible rhino geometry to be further analysed through Ladybug. The results taken from Ladybug will be assessed through the NSW Design Guide 2015 for its solar access. The building will then be optimized using either Galapagos or Octopus with its respective Genetic Algorithm (GA). By using these computational tools, a workflow from concept to optimized can be developed.

6.1.1 ANALYSIS – Importing

The first iteration of this project is finding methods of converting Revit geometry into legible Rhino/Grasshopper geometry so that it may be analysed for solar access. By exporting the two example Revit models that come with the CAD program, the file format chosen for exporting from Revit is a .dwg file. The reason for exporting in such a format is that it retains the most amount of applicable information compared to the other export types, in the form of grouping like items such as doors its own respective layer "DOOR" (Figure 1) the layer names are seen to be the same throughout all imported Revit models tested.



Name	Material	Linetype	Print Wi...
Default		**Continu...**	**Default**
0		Continuo...	Default
GENR		Continuo...	0.09
TOPO-MNM		Continuo...	0.09
TOPO		Continuo...	0.09
SITE		Continuo...	0.05
FLOR		Continuo...	0.05
WALL-ELEV		Continuo...	0.09
RAIL		Continuo...	0.09
PKNG		Continuo...	0.09
WALL-CWML		Continuo...	0.09
FURN		Continuo...	0.09
GLAZ		Continuo...	0.09
DOOR		Continuo...	0.09
DOOR-SYMB		Continuo...	0.09
WALL-PANL		Continuo...	0.09
COLS-ELEV		Continuo...	0.09
STRS		Continuo...	0.09
CEIL		Continuo...	0.05
LINE		Continuo...	0.09

Figure 1: Imported Layers from Revit .dwg file to Rhino

When imported into Rhino the building will be in the form of meshes, and will also retain Revit Family information, resulting in each building within the model to be locked in a group called a “block”. There is a simple but manual fix to this, by exploding the model 4 to 5 times each mesh face will become its own individual item. The reason behind this is so that the meshes can go into each respective layer so that it may be later identified through grasshopper. Once the model importing problem had been figured out, a Revit test model provided by PTW was used to aid in the development in the analysis and optimization scripts on. The model consists of three disconnected multi-residential buildings, which require an analysis and comparison with their original analysis.

6.1.2 ANALYSIS – Identification

The model itself retains a lot of 3D information, what is needed is to use all that information and create a simplified model in the form of a basic floorplan and individual 2D surfaces, which will be later identified as each apartment room. The process of converting the model into a 2D floor plan requires the Grasshopper plugin Human, with this we can identify different building elements through the imported layers. The first element to be considered are the floors of the building this provides the base plate on which the reconstructed floor plan can lay upon. A process of converting the meshes into a solid was done through basic grasshopper means, with the topmost surface of each solid being chosen as the main 2D surfaces.

The same process was conducted on other items such as walls, doors, curtain walls etc. Instead of selecting the topmost surfaces, the bottom surfaces were selected, the result is a basic floor plan of the buildings (Figure 2).

Figure 2: Resulting Floor plan after Grasshopper Conversion

Though many problems were faced when achieving this result, many items such as doors, walls and curtain walls were missing when the first attempt was done. This was due to some items starting position relative to the Z axis were at different varying levels thus not being registered as an applicable surface Another method was required to alleviate this problem, through choosing both the top and bottom most surfaces of the solid, then finally moving all of them so that they may have the same z-axis value respective to its floor.

6.1.3 ANALYSIS – Rationalizing

The floor plan of the building will be used with floors so that the apartment rooms can be created. Though within the model there are various tiny gaps that hinder the process of apartment identification, though it may appear invisible to us, the Rhino is able to notice these slight imperfections. These gaps can be seen between walls and doors of the floor plan (Figure 3) and were solved using different methods.

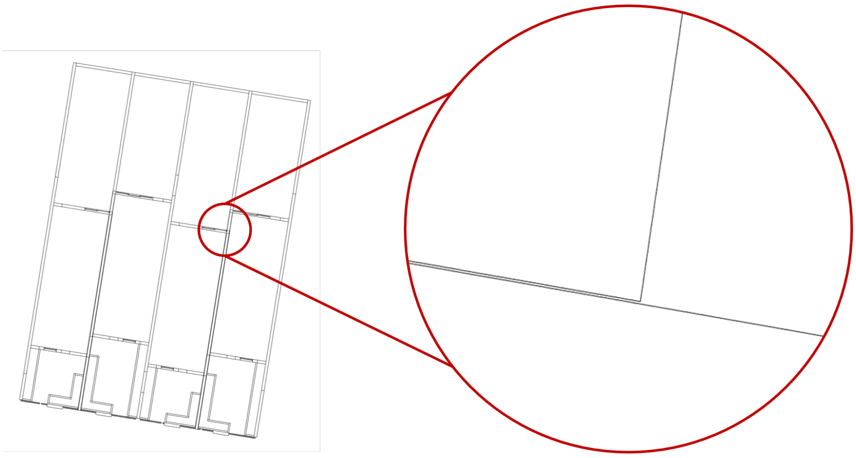


Figure 3: Small Gaps within the floor plan that is scattered throughout the model

The gaps in between the wall were solved by scaling the surfaces by a factor of 1.15. The doors on the other hand were solved by offsetting the surface edges by roughly 400mm, as scaling them would result in unwanted elongation of the doors.

Once these problems are solved, we can begin identifying the apartment rooms of the building. Using the edges of the walls, doors, etc, of the building in conjunction with the floors, we can split the floor surface so that what is left are a plethora of different surfaces with varying lengths and areas, and within the mess are the apartment rooms (Figure 4). All that is needed is to filter out the unneeded surfaces, and what will be left is the individual apartment rooms.

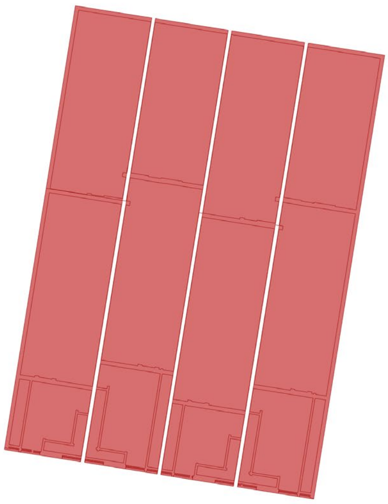


Figure 4: Apartment splitting results before filtering

For this rationalization process, to accurately choose what is considered an apartment by reading through the Apartment Design Guide 2015, three methods of identifying and filtering were implemented. The first being area, most apartments have an area range of 35m^2 (Studio) being our minimum and 95m^2 (Three Bedroom) being our maximum. Surfaces outside this range would be filtered out of the list. The second filtering method would involve the perimeter of the surfaces, since hallways will most likely fall into within area range. This filtering process was done by getting the total average perimeter of all the surfaces and filtering out items greater than the average. The final filtering method is choosing surfaces that contain a door within the surface. By using the centre point of the doors and checking if the point of the door resides within the surface or not. Using these 3 methods of rationalizing most of the apartment rooms (Figure 5) within the building can be used for analysis.

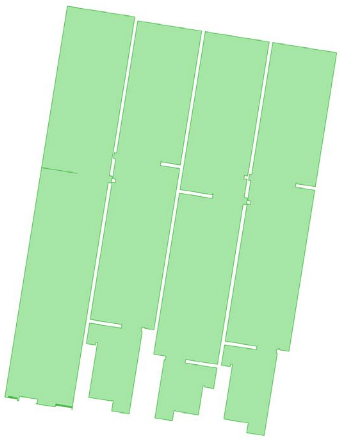


Figure 5: Result of filtering out unwanted Surfaces

6.1.4 ANALYSIS – Analysis Methods

Throughout this process 3 different analysis methods were created, though accuracy of the differing analysis methods were not tested due to unknown validity of the original analysis data. As a base line, the method of analysis will be compared to the current manual process being conducted at some companies. The manual method involved a using series of pictures taken at the suns position during winter solstice (9am – 3pm) at 15-minute intervals. The apartments seen within each subsequent photo are documented based total amount of times seen. This method is tedious and inaccurate, minute details are not considered, and the time intervals are much too large. With the introduction of computational tools such as Ladybug, minute details through a context mesh and time intervals can be considered. Using the same building massing, but instead using Ladybug for the analysis, the results are much different compared to the original, rooms once thought to be compliant were off the 2-hour mark by roughly 5 – 8 minutes. Though this method still does not consider sun light entering the private open spaces and living rooms of the apartment.

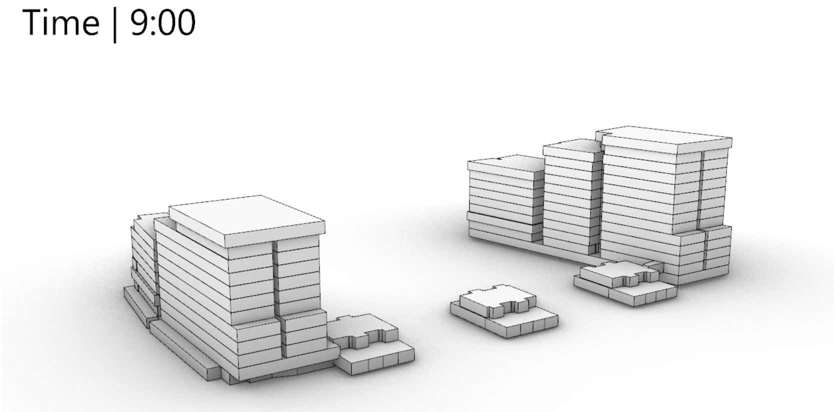


Figure 6: Analysis based on building masses

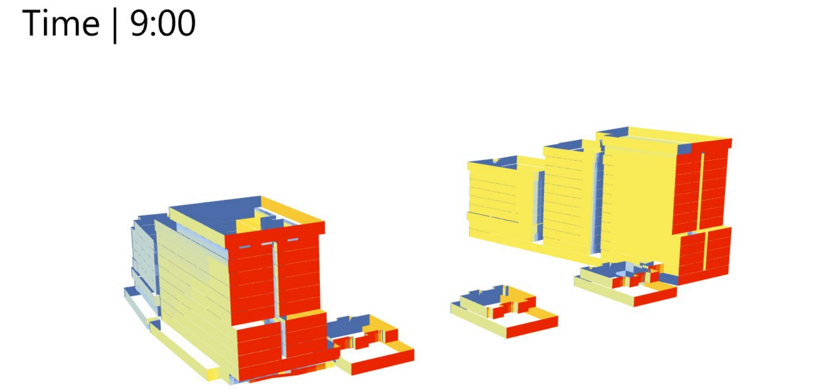


Figure 7: Analysis on the Massing Façade using Ladybug

The first method of analysis created was assessing the sun light towards the floor of the apartments, which have been raised by roughly 1.1 meters. In addition, the context building involved a completely detailed model. Though this method considered the inside of the apartments it is unknown which section is the living room and private open spaces, furthermore there is a lot of open space above the floors that have yet to be utilized. As such the apartment is only considered compliant if any area in the analysis mesh receives 2+ hours' worth of sunlight.

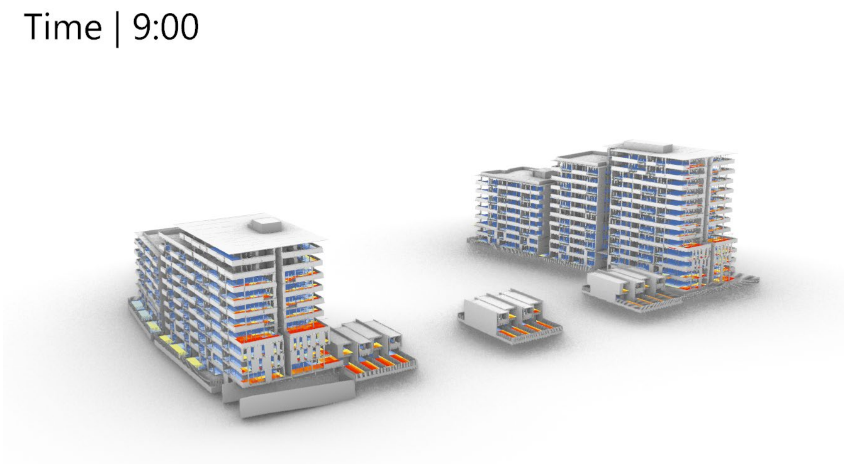


Figure 8: Analysis on the Apartment floor using Ladybug

The second method developed involves using the balcony edges as our analysis mesh, which is extruded up, then moved inwards by roughly 2 meters into the apartment, so that analysis concerning the entire inside of the apartment can be considered. Though this system is not perfect, as the distance from the edge of the balcony to what is thought to be the living room is different, leaving them too far in the building or too far out.

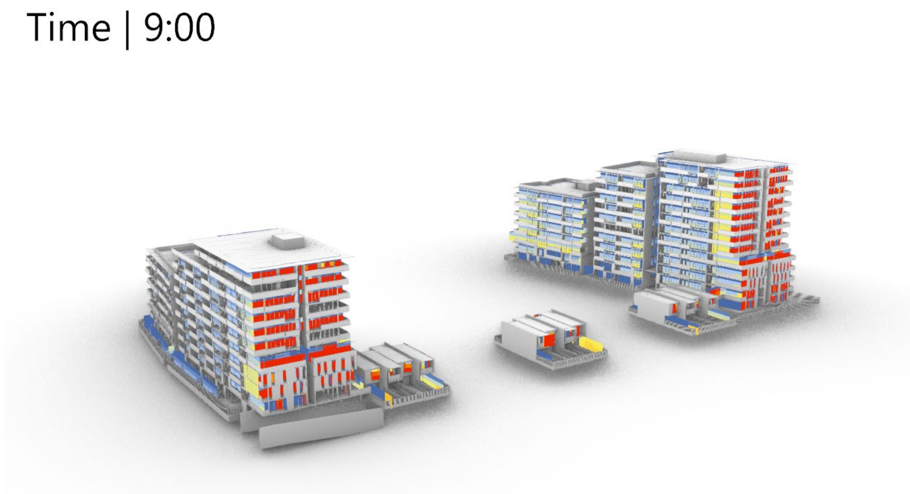


Figure 9: Analysis inside the Apartments using Ladybug

The third and final method involved combining the two previous methods together. The purpose of this was to consider that the results from both the floor and inside wall analysis were over 2 hours then the apartment will be compliant, this was the expected result. Problems that were previously stated such as unknown location of living rooms, were attempted to be solved, but currently remain. Though they may cause some slight inaccuracies in some areas, the results are shown to be somewhat similar compared to the previous methods stated.

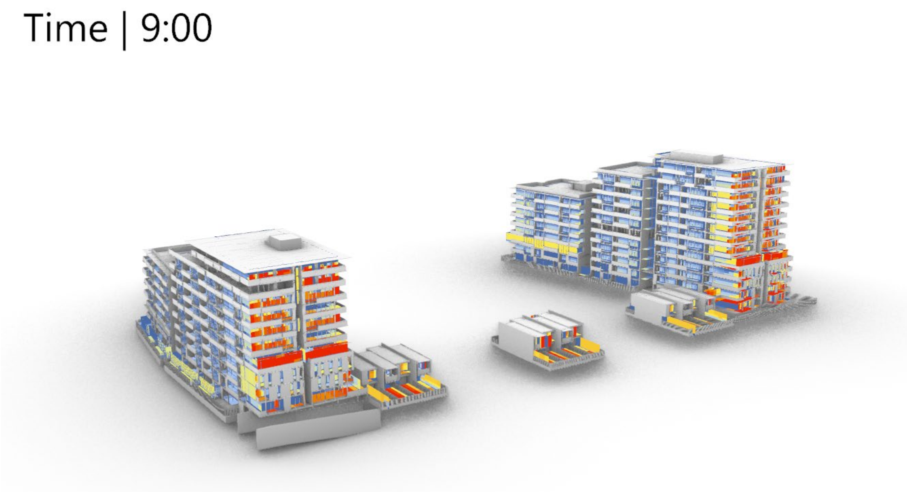


Figure 10: Using the combination of Apartment Floor and Inside Walls as method of analysis

By assessing the validity of the methods presented, what is thought to be ideal in terms of accuracy is the combination of the Inside Wall and Floor analysis. Using this method, the total compliance of the buildings is 57.67%, deeming it not compliant with building regulation, resulting in the need for optimization. In addition, the method chosen will be used as the mode of analysis within the analysis section of the optimization script.

6.1.5 ANALYSIS – Output

At its current state there is no validity or usefulness of the analysis information, as it all resides within a non-informative visual state. What is needed is to output this information into 2 human friendly outputs, these were decided to be in the form of a floorplan highlighting which apartments are compliant. This process was done using Ladybug's provided Photo Export component along with a C# script that enables it to automatically iterate over the entire slider. Finally, the second output will be in an excel spreadsheet presenting the apartment number, max total hours of sunlight and its compliance in the form of True or False. Using the plugin TT Toolbox, within it contains a simple excel export component that would quite easily create an excel spreadsheet, provided with valid information and information structure i.e. appropriate Grasshopper list management.

Building	Level	Room Number	Total Hours	Compliant
A	0	1	2.066667	TRUE
A	0	2	2.066667	TRUE
A	0	3	2.066667	TRUE
A	0	4	1.533333	FALSE
A	0	5	1.533333	FALSE
A	0	6	1.866667	FALSE
A	0	7	1.533333	FALSE
A	0	8	3.8	TRUE
A	0	9	3.933333	TRUE
A	0	10	4	TRUE
A	0	11	4	TRUE
A	0	12	3.866667	TRUE
A	0	13	3.933333	TRUE

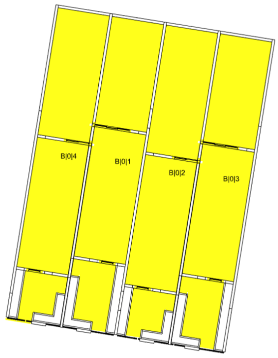


Figure 11: Output from grasshopper Excel Spreadsheet (Left) outlining compliance of apartment and Floor Plan (Right) highlighting which rooms are complaint

6.2.1 OPTIMIZATION – Apartment Generation

Two methods were presented when optimizing a building for solar access, the first being to modify the pre-existing model by changing window positions, room layout etc. Though may yield to be the most useful, given the time frame and the complexity of the model, developing this method would be too intensive. Respectively the second suggested method would be applied, which is to completely generate a new building mass based on the Gross Floor Area (GFA) of the original building, this original site area and solar access.

The method of achieving the task is using Genetic Algorithms (GA). GA is an evolutionary based solver that attempts to optimize its input by changing its genes (in this case sliders), which will subsequently change the fitness value. Its process involves creating an initial population with random genes, then deleting 50% of its population based on how close it is to the ideal fitness value. With the remaining 50% of the population, it will begin repopulating to its initial population size, this process is to provide genetic diversity. Then finally the repopulated population will be mutated, this is to further add genetic diversity. All this is provided through the plugins Grasshopper plugins Galapagos and Octopus.

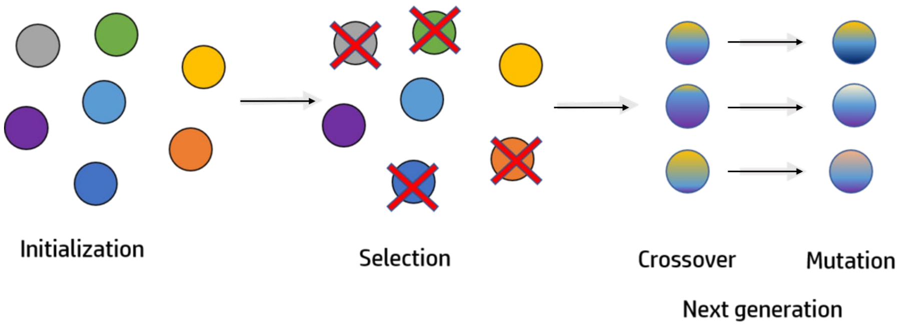


Figure 12: Process of Genetic Algorithm

Iteration 1:
The first iteration involved using Galapagos, and since that Galapagos is only able to intake one fitness value, the Apartment Generation and Optimization process needs to be separated. The apartment generation script would simply change the height and size of each building was created. As for the analysis, direct sunlight was measured on the façade of the building and based on those reading will change the orientation of the buildings, based on a universal angle. This proved to be a very reliable process, relative to how simple it is.

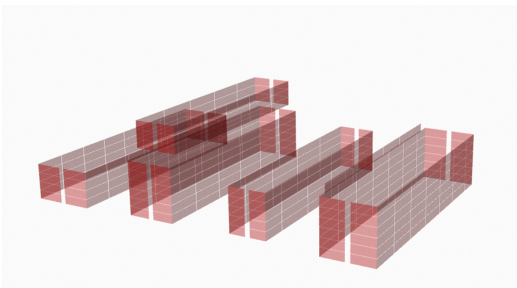


Figure 13: First Iteration of Apartment Generation using Galapagos

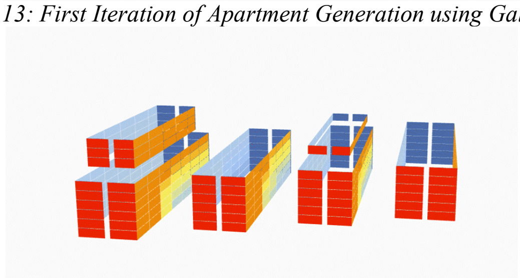


Figure 14: First Iteration of Apartment Optimization using Galapagos

Iteration 2:
Due to the limitations of Galapagos, this iteration opted to use Octopus, which can take in multiple fitness value, along with that more options are available for change, such as mutation rate. Many changes were made to

improve the overall apartment generation and optimization process. The changes made involves the implementation of 2-bedroom apartments rooms, changeable building positions, analysis is now based inside the apartments and general improvement in processing speed.

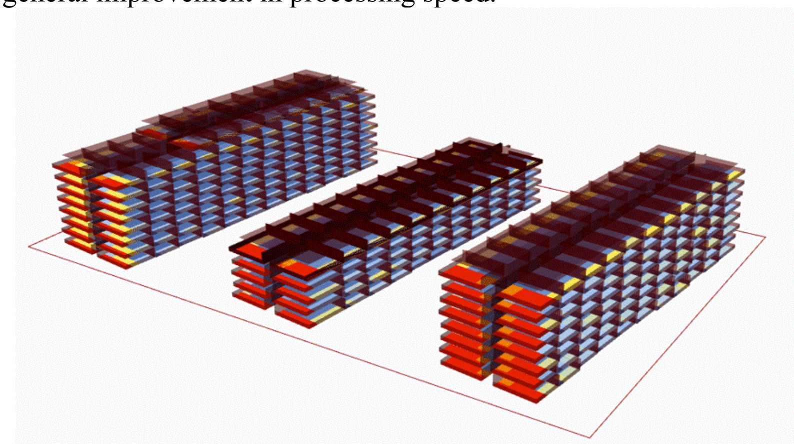


Figure 15: Second Iteration Apartment Generation and Optimization using Octopus

Iteration 3:

By assessing the results taken from Octopus and feedback from the tutors, further changes were applied to the Grasshopper script. What was implemented were to have a more sophisticated positional movement, instead of having a series of fixed positions, the buildings would move around its given yet restrictive area. The other implementation was to give the apartment rooms the ability to move into or away from the building, allowing more possible variant of an optimized building.

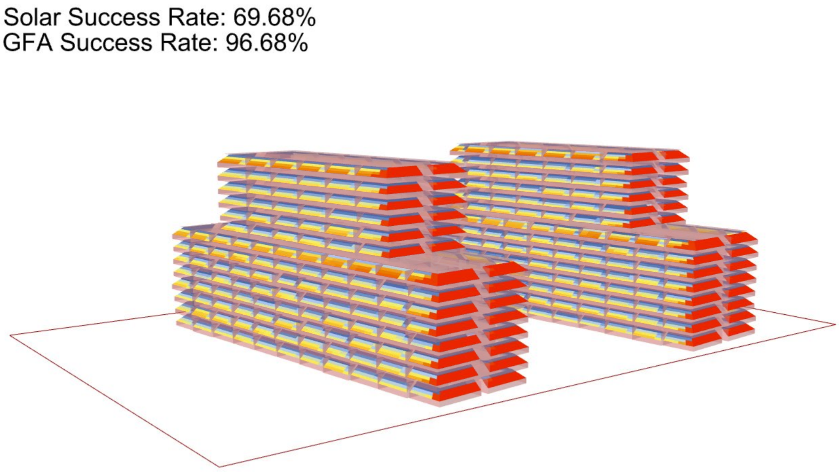


Figure 16: Third Iteration Improved Apartment Generation and Optimization, Displaying the success rate of percentage of compliant rooms in total and how close it is (In terms of GFA) compared to the original.

By using the method used in iteration 3 the final optimized result compared to the original building, yields a significant improvement in solar access. The original building had an average of 57.67% compliant rooms based on the implemented analysis method, the optimized version came from running Octopus for an hour, had an average outcome of 92.91% of compliant rooms, which is an increase of almost 40%.

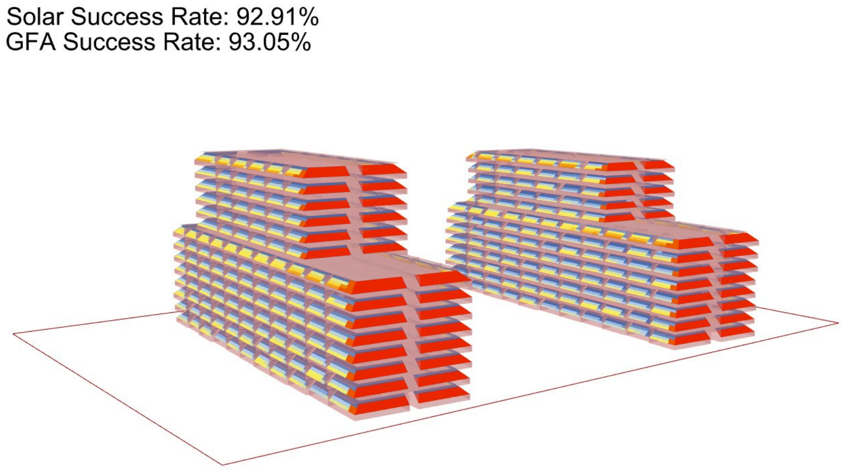


Figure 17: Final Building Result optimized for solar access, after running Octopus for 1 hour

7. Discussion (evaluation and significance)

With the introduction of computational methods towards the process of analysis and optimization, an improved workflow can be established. The goal previously stated for this project was to find solutions to improving solar access within multi residential buildings. This aim was tackled by introducing different methods of analysing a building, in the hope of improving the accuracy and reliability of the results. Furthermore, developing a simple script that would generate new building forms like the original, with its aim to optimize its solar access using genetic algorithms.

The analysis section of the project tackled the conversion of Revit to Rhino models, so that it may be accurately analysed through Grasshopper and its plugin Ladybug. This goal of converting the Revit model was successfully achieved towards one model, though with if the script were developed in the future, it will most likely be able to achieve a similar goal, compared to the one presented in this project. As for the analysis of solar access 4 method were explored, analysis towards the façade, apartment floors, inside walls and a combination of floors and walls. Through review from peers and tutors regarding the different methods of analysis, what was deemed the most accurate is the combination of floors and walls, as it was able to validate the sunlight incident both the private open spaces and what is assumed to be the living room. Limitations are present within this script, first is the assumption of the living room location within the apartment, which has been set 2 meters from the balcony edge. What would be needed to alleviate this problem is to introduce an improved apartment identification system, outlining the balcony and living room. The second limitation is the relation between CPU intensity and analysis accuracy. The accuracy of the results is determined by how fine the analysis mesh is, resulting in greater CPU usage, thus longer waiting time for the analysis to process.

Within optimization, a basic process of improving a building for solar access was developed. Through the project three different iterations of apartment generation and optimization was created. Using Genetic Algorithms, new building is generated with two goals in mind, to have a similar Gross Floor Area as the original model, and to improve the solar access within the generated buildings. Currently the optimization script can only generate one apartment type, leaving room for other building types with improved final solutions. Furthermore, within its current state other building regulations are not considered, such as building height, privacy, apartment mix, etc. These regulations can be added to further improve the validity of the building. Different approaches to optimizing the building can be implemented, instead of generating new buildings, adjusting the existing one can prove to be much more useful.

Overall the aims that were outlined at the beginning of this project were achieved to a fault. There are some problems that can affect the accuracy of the results and the outcome from the optimization process. With the implementation of such methods into the AEC industry, it will improve the process of optimizing and analyzing a building for other regulations not explored in this project, such as privacy.

8. Conclusion

Within this project, the aim was to automate and improve the analysis and optimization process currently being performed in the AEC industry through the introduction of computational methods. With the exploration of current practices currently being performed in the AEC industry as well as the NSW building regulations, a workflow automating the analysis and optimization process of one regulation was created. The specific area the paper investigates, is the improvement in solar access within multi residential buildings using genetic algorithms/evolutionary solvers. What was developed were two separate scripts that were relatively successful, the first one implementing an improved method of analysing a building for solar access, the second generating an optimized building form for solar access, with similar GFA to the original building. By using this process, the tasks take roughly 1 hour altogether to complete, 5 minutes for the analysis and a recommended 1 hour for the optimization process. This is a significant reduction compared to manual method, practiced in some architecture studios.

If the project is further developed implementations towards the optimization process will be where it is most needed, as to further improve its viability in a workplace. Implementation could include, other building regulations, different apartment types, and finally curtain walls and windows being added into the list of changeable variables. This is to simply add complexity as more possible outcomes can be accounted for, resulting in better optimized buildings.

With the introduction of computational methods to the AEC industry, much more accurate designs can be achieved, improving the lifestyle residents/users have within the of the building.

Acknowledgements

I would like to thank the tutors Nicole Gardner, Hank Haeusler, Yannis Zavoleas and Cristina Ramos at UNSW CODE for helping and guiding me throughout the process of this project. I would also like to thank my supervisors Brandon Heng and Michael Yip as well as the people at PTW Architects for aiding me with their support and input in this project, providing 3D models and general guidance.

References

Abdelmohsen S, Lee J, Eastman C. 2011 "*Automated Cost Analysis of Concept Design BIM Models"* College of Architecture, Georgia Institute of Technology

Altomonte S 2008 *"Daylight for Energy Saving and Psycho-Physiological Well-Being in Sustainable Built Environments"* University of Nottignham

De Luca F. 2017 *"From Envelope to Layout – Buildings Massing and Layout Generation for Solar Access in Urban Environments"* Sapienza University of Rome

De Luca F, H Voll 2017 *"Solar Collection Multi-isosurface Method Computational Design Advanced Method for the Prediction of Direct Solar Access in Urban Environments"* Springer Nature Singapore p 170 – 181

DeKay, M. 1992 *"A Comparative Review of Daylight Planning Tools and a Rule-of-Thumb for Street Width to Building Height Ratio"* Proceedings of Passive Solar Conference 1992

da Veiga J, La Roche P 2002 *"A Computer Solar Analysis Tool for the Design and Manufacturing of Complex Architectural Envelopes: EvSurf"* 2002 Proceedings of the 6th Iberoamerican Congress of Digital Graphics p 105 – 109

Giani M, Belfiore C, Lobaccaro G, Masera G 2013 *"A Design approach for the Solar Optimization of Built Volumes"* CISBAT 2013 International Conference - Cleantech for Smart Cities and Building

Roudsari M, Pak M, Smith A, Gill G 2013 *"Ladybug: A Parametric Environmental Plugin for Grasshopper to Help Designers Create an Environmentally-Conscious Design"* 13th conference of International Building Performance Simulation Association Chambery

Shoory M 2016 "*The Growth of Apartment Construction in Australia*" RBA Bulletin, Reserve Bank of Australia