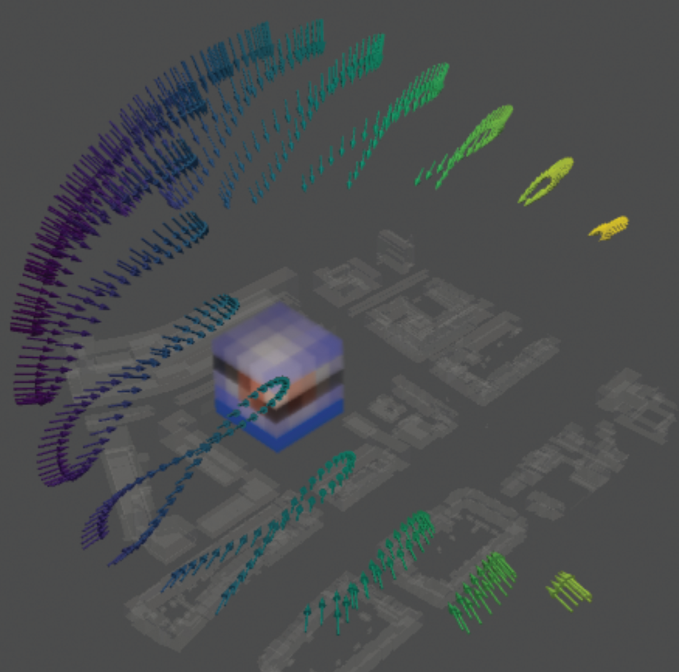
MSc thesis in Building Technology

Generating building envelopes using multi-criteria decision analysis

Max Ketelaar

2021/2022



TU Delft

GENERATING BUILDING ENVELOPES USING MULTI-CRITERIA DECISION ANALYSIS

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of

Master of Science in Architecture, Urbanism and Building Sciences

by

Max Ketelaar

December/January 2021/2022

Max Ketelaar: *generating building envelopes using multi-criteria decision analysis* (2021/2022)

Thesis Repository: https://github.com/Maxketelaar/thesis

ISBN ???–??–????–???–?

The work in this thesis was made in the:

Genesis Lab Graduation Studio
Department of Building Technology
Faculty of Architecture & the Built Environment
Delft University of Technology

Supervisor:	Dr. ir. Pirouz Nourian
Committee:	Dr. Eleonora Brembilla
	Dr. Verena Balz
	ir. Shervin Azadi

ABSTRACT

This thesis concerns the application of different multi-criteria decision analysis (MCDA) methods and optimization strategies for finding the optimal envelope for a given building plot and lighting performance indicators. More specifically, the PV potential, daylighting potential, and Sky view Factor of the building are maximized using different solvers. The method utilises an existing data framework called TopoGenesis and solves the problem using the PyGmo library.

A ray tracing is used to find all possible collisions between the objective test points and the building mass and environment. These collisions are saved in interdependency arrays.The problem is first presented as a standard integer programming problem, but solving this problem is not feasible if complexity needs to be kept at a reasonable level. An alternative method is therefore proposed that uses (meta)heuristics to find an optimal solution from the interdependency arrays. The occupation status of the massing is used as inputs for the decision variables.

After the application of this method on small scale toy problems, a few of the design options are selected and evaluated by their performance indicators, as well as the measure with which the option makes sense from a more traditional design perspective. The method is then also applied at a smaller on a test case which is a more detailed scale of the problem. The comparison of the performance and results of both methods give insight into the recommended workflow, settings, and pitfalls for finding an optimal solution to a multi-criteria design problem with visibility objectives. From the initial results, the Improved Harmony Search algorithm seems to be the best option for solving these types of problems, but more research is needed to verify this statement.

ACKNOWLEDGEMENTS

Thanks to Pirouz & Shervin for introducing me to this area of the field back in 2018 and having inspired me to pursue this and keep taking their courses ever since then. Specifically thanks to Shervin for all the practical help with coding and formulating my problem and to Pirouz for the philosophical views on computer science and mathematics within architecture. Thank you to Anastasia for all the discussions about history and the study sessions during the pandemic.

My family and friends for supporting me during my studies for many years. First my parents Arend and Birgit who have been very patient, along with my brother and sisters who would always know better: you have taught me to win (or lose) arguments.

A special thanks to Nina who simply refused when I said I was going to quit my running challenge, I'm now almost at the 1000! My other friends at WAVE especially Ivo whose work ethic was inspiring and Oscar who was always ready to study together. I would also like to thank Manon for keeping me motivated when things were looking down and sharing the joy when things were looking up.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

1 INTRODUCTION

Increases in labour, ground, and development costs have decreased the attractiveness of affordable housing projects in the Netherlands. In the past, government policies were instituted to create new subsidized housing projects, but these have largely been discontinued, decreased in scope or scale, or privatized. Combined with the current general housing shortage, high quality but affordable and high density housing has started to become a scarcity [23; 7].

Due to an increasing desire to limit greenhouse gas emissions and the significant share of almost 40% of the building industry in this aspect [1], making the building industry more sustainable is a high priority for governments and corporations alike. Both reducing emissions during construction and finding more sustainable ways of providing energy for the operation of buildings are strategies that can be employed to achieve these goals. One of the avenues for achieving these goals is implementing smarter designs with regards to energy usage, in particular the sun's energy.

1.1 LIGHT

Direct insolation can supply the building with energy from photovoltaics, as well as heat up the building (thereby decreasing heating demand). At the same time, daylight access has been proven to increase productivity in offices [19], increase the value of real estate [35], increase student performance in schools [15] and more generally provide an increase in mood and health [4]. Regarding daylighting and insolation, the Dutch Building Decree (Bouwbesluit) has determined that living spaces (verblijfsruimten) receive an area of daylighting according to a minimal amount of m2, or a percentage of that living spaces' area. The way this is calculated is through calculating the area of all valid openings in the facade and reducing this area by factors such as projection angle, shadings or obstructions, window frosting etc. The result of this is the equivalent daylight area. The calculation of this equivalent daylight area has certain disadvantages however.

Firstly, aspects that influence this value need to be known beforehand. This means overhangs, shadings, window angles etc. all need to be known when estimating if the minimum demands can be met by the current design. In the earliest design stages however, these factors may still be unknown or up for debate. Secondly and more importantly, these calculations only take factors into account that are inside the building parcel and only estimate the daylighting for the proposed building and not its effects on the surrounding area. This means that any shading by surrounding buildings and the shading by the proposed building on surrounding areas might be neglected in this phase of the design. To remedy this problem, municipalities usually employ their own norms for insolation and daylight access on top of the equivalent daylight area method. Most municipalities use (a modified version of) the TNO norms which state:

- at least 2 possible hours of insolation per day in the period of 19th of february – 21st of October (8 months) in the centre of the windowsill inside the window" (lenient norm)

- at least 3 possible hours of insolation per day in the period of 21st of february – 22nd of November (10 months) in the centre of the windowsill inside the window (strict norm)

The The Hague municipality for example modifies the 'lenient' norm by adding that only solar positions of 10° or more may be taken into account, while the entire facade may be taken into account, regardless of window position. [BRON:TNO norm]

Both passive heating and active energy generation by the sun are attractive options for designers, but balancing these factors with the other design variables is a complex task that has to be considered as early as possible in the design process since here the largest difference can be made on the eventual performance of the building [27]. Building envelopes represent the boundaries of the mass that a building can potentially assume. It is a tool used in early design stages to limit the maximum extents of a design and also to ensure certain design objectives can still be achieved. Generally, the objectives that these envelopes are created for pertain to solar access (solar envelopes). The goal of implementing a solar envelope in the design phase of a project is to ensure the building and its surroundings are exposed (or remain exposed) to the sun for a certain period of the year. A building envelope applied in the early design stages plays an important role regarding energy performance [13].

1.2 MAKING DECISIONS

We can recognize that the variables described in the previous section can be interconnected, conflicting, or reinforcing towards each other. To make decisions on multiple variables and diverging goals, several industries have successfully applied operations research and more specifically Multi-Criteria Decision Analysis (MCDA) in the past. There is an increasing trend to also apply these techniques to the built environment [16]. Generally, no standard method is taught in architecture or planning education but [constantly increasing performance of modern computers and advances in simulation techniques have made the technical limitations less evident as they once were] [source7]]. This means there exists a gap between what is currently standard practice and what is possible. This thesis is an attempt to research the nature of this gap, find a workable methodology for a specific set of energetic, climatic, and solar design objectives, and explore the merits of the different approaches to such problems as described in this paragraph.

The rest of the thesis is organised as follows: Chapter 2 establishes the research framework: Disciplinary approach and scope, current literature and methods, objectives, research questions, and problem statement are given. Chapter 3 describes the developed method: first by its constituent toy problems and then as a proposed complete method. Finally, an application of the method on the case study is provided. Chapter 4 concerns the evaluation of the developed methods. Conclusions and further research is presented here. Finally, Chapter 5 contains the personal and academic reflection. At the back, Chapter 6 there is an appendix containing the results and flowcharts, pseudocode, and visualisation of the developed algorithms.

2 RESEARCH FRAMEWORK

This chapter presents the research framework within which the thesis takes place. First, the approach and scope are given from a disciplinary perspective. Some relevant literature for the development of the method is then explored. After this, the general objectives, research questions, deliverables, and design objectives are described and the final section of the chapter formalises these objectives into a problem statement that can be used in the following chapter to devise the main method.

2.1 DISCIPLINARY APPROACH AND SCOPE

The context, problem statement, objective, scope and research questions are identified in the following sections. Along with this, the relevant scientific background is explored through the literature review. This includes topics such as the state of MCDA (both general and more specifically in relation to the building industry), solar simulations, and generative design in the building industry.

After this, the key design objectives are modified so they can be included in the method. For each of these, the scores need to be aggregated from their respective constituents, and these aggregated scores are then compared using different MCDA methods. The resulting values for each variable are then used to generate different massings (configurations) in the form of toy problems, and these massings can then be evaluated on how well they perform per each different MCDA method.

From this, the research will conclude with suggestions on what methods perform best for what purpose and in what context. The results are validated by reapplying the method to a case study to find if the method is generally (or even universally) applicable or if the solution only fits with this specific set of design problems. The problem as explored and described in detail from section 2.3 to 2.5 and is interdisciplinary in nature as illustrated by the Euler diagram in figure 2.1.

2.1.1 scope

The research concerns testing different strategies of MCDA with a specific set of criteria. It must describe a methodology for implementing MCDA methods to find (near) optimal massings at a building scale. This means that the scales above this (neighbourhood, city, region) as well as below it (room, building detail, component) are not included in the research. The results of the research should however be suitable for use as a baseline to make informed design decisions. The focus and purpose of the research is to aid in the earliest stages of the design process without limiting the freedom of the designer too much. This means that the end result will always concern a configuration of voxels and their performance towards the criteria applied in the method, and will never represent an actual zoning of the rooms or design of the building. The emblem below is used to illustrate the intended positioning on an axis that represents planning or design intent.

For practical reasons, some simplifications and assumptions have to also be made on the goals that are considered when applying the MCDA. By limiting the number of criteria we can keep the focus of the research on the performance and ease of use

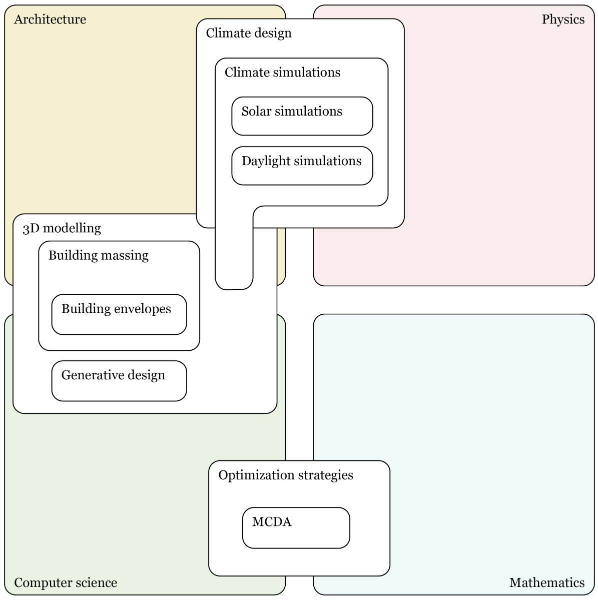


Figure 2.1: Euler diagram of the relevant disciplines and research areas

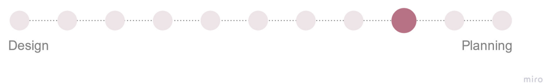


Figure 2.2: Positioning on design-planning axis (own image, emblematic)

of the MCDA methods in the context of generative building design. Specifically, the considered criteria are as follows:

1. PV potential
2. Daylighting potential of the building envelope
3. Daylighting potential of the area surrounding the plot
4. Building heat retention potential
5. Floor space index

In 2.4 and 2.4 and 3, how we deal with these criteria will be explained in further detail. All aspects beyond this that might influence the building performance are not to be taken into account during the research. Examples of this include but are not limited to: Structural design, facade design, (functional) zoning, construction techniques, shading strategies, alternative sustainable energy generation, artificial lighting strategies, (end) users, cost analysis, rent estimation, proximity to amenities etc. Python will be the only language used for programming the methodology in order to ensure reproducibility and more convenient library integration. Furthermore the writer has the most proficiency in this language.

Below is an overview of aspects included and excluded within the scope of the research:

Within the scope

- MCDA strategy and methods in Python
- Daylight and sunlight simulation methods in Python
- Heat retention strategy

- Generative design strategy
- Early design massing (conceptual stage and planning stage)

Outside the scope

- Structural design
- Facade design
- Zoning and function assignment
- Construction techniques
- Shading strategy
- Lighting strategy
- Energy generation strategy
- End user or actor research
- Cost and benefit analysis
- Centrality and proximity analysis
- Anything below the building plot scale (room/detail/component)
- Anything above the building plot scale (neighbourhood/city/regional)
- Late design development (design development stage and construction design stage)
- Usage stage building performance
- Post-usage stage building performance

2.2 EXISTING LITERATURE, METHODS, AND LIBRARIES

In the next section, the most important conclusions of the literature study on daylighting simulations and MCDA methods have been collected to give an overview of the topics and writers researched and the lessons learned from these texts. Searches were conducted using Google Scholar. At the end of the chapter, relevant libraries that apply these methods are listed.

2.2.1 daylighting simulations

The research into simulation of daylighting is predominantly focused around offices, schools and hospitals etcetera, i.e. non-domestic buildings. This is mainly because the activities pursued in these types of buildings require more and higher quality daylighting [34]. The most commonly used tool for these types of simulation is RADIANCE and trust in these types of tools is on the rise [Reinhart et al.]. When analysing the accuracy of different daylight simulation methods, Reinhart and Herkel find that the quality of a simulation is highly dependent on whether hourly illuminances are considered in the estimations.

The influence of daylighting on the thermal performance of a building is another field of interest for researchers in daylighting. A study attempts to reduce energy demands for heating and lighting finds that the correct design parameters can reduce these demands significantly while increasing daylighting performance [37]. Daylighting simulation can accurately predict actual daylighting values according

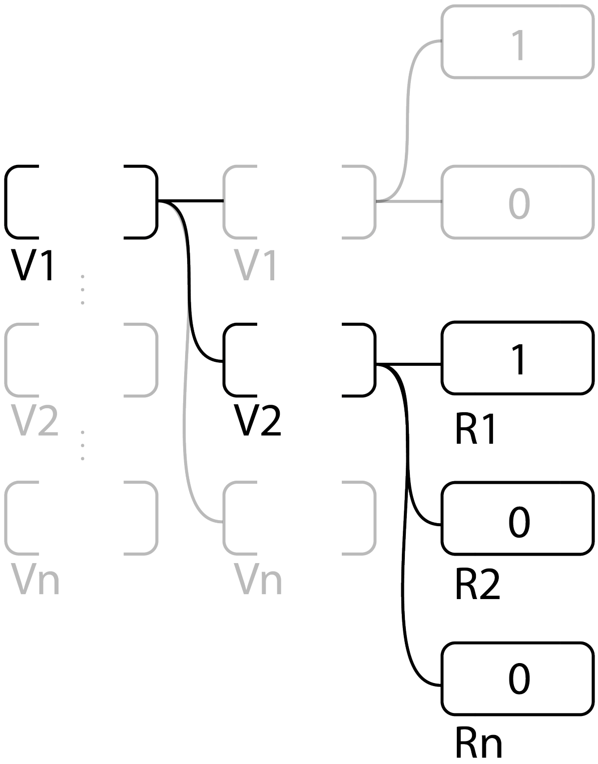


Figure 2.3: The structure of the interdependency graphs used throughout the developed method: nested arrays of hits and misses.

to Reinhart and Walkenhorst and Labayrade et al., but is highly dependent on finding the correct parameters for the simulations.

Another relevant piece of research in daylight simulations especially its application for generative design and MCDA is that of Anastasia Florou which describes a methodology for precomputing the rays that impact daylighting and insolation in order to use these values for a space allocation problem. These rays are coupled with a discretized representation of a building massing to create an 'interdependency graph' to find the relations between all possible rays and voxels in the model. See figure 2.3 for reference towards how this is structured. An array that contains, for each voxel V, towards every other voxel V, whether each ray R is blocked or not by that voxel.

2.2.2 multi-criteria decision analysis

In urban planning, decisions are made during several planning phases that have implications on various domains and scales for different actors. For an individual or group of people, it is (practically) impossible to understand all interdependencies within the decision space. Subjective or context-specific decisions cannot definitively be taken beforehand, but these decisions can be analysed beforehand. By anticipating how these decisions affect the performance of the building, they can be altered beforehand to produce better results.

Mixed-Integer Linear Programming (MILP) is suitable for this since it allows to quickly generate a multitude of plans and include both continuous and discrete decision variables. Schüler et al. successfully implement a MILP method on a neighbourhood (and lower) scale. Specifics are provided on what aspects of the model are standard and which ones are novel but no detailed guide on the implementation is provided. This method is a form of Mathematical Programming (MP) and similar or analogous methods to this include Linear Programming (LP), Constraint Programming (CP), Integer Programming (IP), Quadratic Programming (QP), and

	AHP	MAUT	DRSA	PROMETHEE	ELECTRE
ease of use	-	/	+	/	-
compensation	-	-	+	-	+
support	+	+	/	+	/
thresholds	-	/	+	+	+

Table 2.1: Overview the most popular of the examined MCDA methods. DRSA seems to be the most promising option

Binary Integer Programming (BIP)

Another key piece of literature reviews the use of MCDA in the context of architecture and urban planning with regards to energy efficient construction and concludes that a modified Analytic Hierarchy Process (AHP) (fuzzy) is applied the most in general, followed by Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) and Complex Proportional Assesment (COPRAS). For the selected methods, the authors examine the capability to decompose the decision problem, improve the transparency of the decision processes, facilitate comparison of various decision alternatives, and identify their strengths and weaknesses. Weaknesses include: For AHP/TOPSIS a lack of consideration of interactions between various design criteria. Simple Additive Weighting (SAW) and ÉLimination et Choix Traduisant la REalité (ELimination Et Choice Translating REality) (ELECTRE) were rated lower in terms of ability in pair comparison or ability to manage low quality input data. A hybrid approach is becoming increasingly popular. To make comprehensive assessments, it is better to use two or three different types of MCDA or a combination of the two [24].

In another inventory research, Multi-Attribute Utility Theory (MAUT), AHP, Preference Ranking Organization Method for Enrichment Evaluation (PROMETHEE), ELECTRE and Differential Evolution - Simulated Annealing (DRSA) are described by their performance in respect to ten criteria that sustainability assessment tools should satisfy. The review shows that MAUT and AHP are fairly simple to understand and have good software support. Only MAUT achieves robust results, while ELECTRE, PROMETHEE and DRSA are non-compensatory approaches, accept a variety of thresholds, but suffer from rank reversals. DRSA is less demanding in terms of preference elicitation. DRSA also emerges as the easiest method, followed by AHP, PROMETHEE, and MAUT, while ELECTRE is regarded as fairly difficult to use and understand [11]. A general grouping of MCDA approaches is proposed by [33] who distinguish the methods by three underlying psychological theories: utility function, outranking relation, and sets of decision rules. The table **??** gives a comparison of some of the more relevant of the methods researched in regards to criteria that we are interested in such as their ease of use or intuitiveness, their ability to compensate or use trade-offs, the support and documentation available, and the ability to use threshold values which can be used to model conditionals (such as the blockage of a ray of light).

One approach to multi-objective problems is using a heuristic method as opposed to an exact method such as a programming. Caldas applies a genetic algorithm and simulated annealing for finding the best trade-offs between conflicting objectives such as building costs, energy consumption, and embedded greenhouse gasses and successfully generates building geometries. Nagy et al. argue that applying such methods allow designers to explore a wider range of design options than would be possible using traditional methods. They apply the Non-dominated Sorting Genetic Algorithm 2 (NSGA-II) for generative design. This algorithm seems to perform well when compared to other solvers [12]. Many other heuristic global optimization methods exist and have successfully been applied for solving multi-objective problems in the built environment such as Modified Ant Colony Optimization (MACO)

[32] or DRSA [8].

We can conclude that there are a multitude of MCDA methods available that can be applied to the problem as described in previous sections. The main takeaway from the studied literature has to be that there is no definite answer to which method is best. However, due to inaccuracies and uncertainties inherent in MCDA, employing a hybrid method and testing and comparing multiple classes of methods generally seems to be the desirable strategy. An exact method is often desirable but heuristic methods are generally easier to implement and utilise.

2.2.3 python libraries for optimization

Several libraries were examined. Google's OR tools offers extensive documentation and tutorials and enables linear, integer, and constraint optimization. CVXOPT enables quadratic optimization capabilities, while SciKit-Criteria offers many different MCDA methods such as ELECTRE, Weighted Sum Method (WSM), and TOPSIS. Finally, PyGmo offers many evolutionary algorithms for optimization such as NSGA-II and MACO.

2.3 OBJECTIVES

The objective of the research is to devise a general methodology or workflow for generating building massings of high-performing solar and climatic configurations in dense urban contexts by using multi-criteria decision analysis. To achieve this, the problem must first be formulated in such a way that it can be understood on a mathematical level.

This method is to be applied in an environment that can represent the context of the Randstad area in the Netherlands; a highly developed, interconnected, dense urban region. The method is applied on a minor and major level to validate and compare the results between different scales of the problem. For this purpose, toy problems are used to research and illustrate the mechanisms at play on the minor scale, where a test case in Rotterdam is used to validate the results on the major scale. A conclusion has to be drawn on what method(s) and approaches of MCDA are most useful for these specific objectives. Ideally, this conclusion can be extended to a general recommendation on what MCDA method is most valid and practical for the more general type of socio-spatial-climatic objectives that are pursued in the early stages of building design and planning projects. A simplified representation of the process can be seen in image 2.4.

2.4 RESEARCH QUESTION

Taking into account the objectives of the research, we can formulate the following main research question to be answered:

> **"How can we utilize Multi-Criteria Decision Analysis to generate massings of high-performing solar-climatic configurations in a dense urban context?"**

Looking at this question and the research objectives in a more detailed manner raises sub-questions and objectives. These pertain to the terms **MCDA**, **high performance**, **massing**, and **solar-climatic configurations**.

1. What types of MCDA methods are available to us?

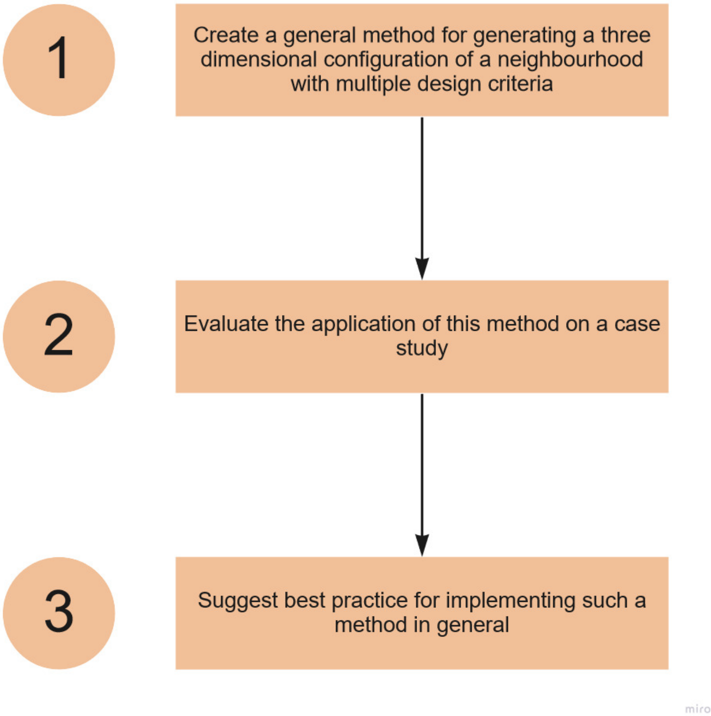


Figure 2.4: The research goals simplified

2. What type of problem are we trying to solve (in a mathematical sense)?

3. Of the available MCDA methods, which ones are suitable for our type of problem?

4. What do we understand by 'massing' in this instance?

5. What exactly are the (solar-climatic) objectives we are pursuing and what are the objective functions of these objectives?

6. How do we define the performance of our massing? In other words, when does our massing perform well enough to call a solution 'optimal'?

7. Can we generalize the application of these techniques for other configurational problems? In other words, can we apply the developed methodology on problems that have different objectives?

2.4.1 deliverables

At the end of the research process, the following products should exist:

- An inventory of MCDA strategies available to us. Of this inventory, the most relevant methods for our problem need to be identified and tested.

- A mathematical formulation of the design problem to be used as input for the MCDA solving methods when possible. Alternatively, if a mathematical formulation of the problem as a whole can not feasibly be used to generate the building massing, a mathematical formulation for finding the objective functions that can be utilized to generate the building envelope should be provided.

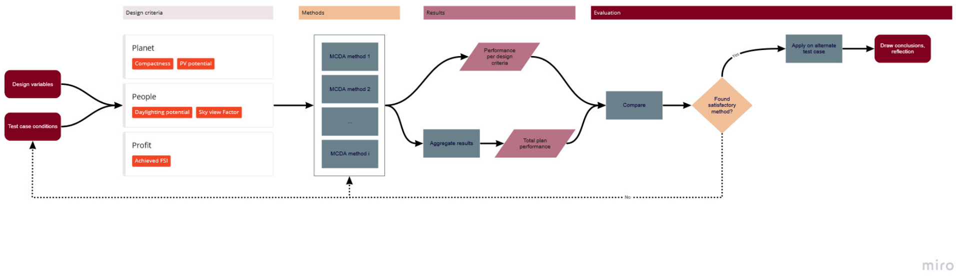


Figure 2.5: Flowchart for the methodology: The red rounded boxes are starting and ending in/outputs. The rectangular boxes represent (automated) processes, while the parallelograms are data and the diamonds are decisions. A larger version can be found in the appendix

- A methodology for applying MCDA on the criteria of solar, daylighting, spatial, and energetic performance. This will be created as a program that can be used independently of any CAD software so as to ensure accessibility and reproducibility. More concretely, this will be in the form of Jupyter notebooks. The input will consist of a planning area and its attributes represented mathematically as mentioned in the first point, while the output consists of the decision variables and the performance of the configuration that can be generated with these variables.
- In the thesis' conclusion, a judgment will be made on the examined MCDA methods and their usefulness for this particular set of decision criteria. Finally, a general recommendation is made on the application of these techniques on similar design problems.

2.4.2 design objectives

Traditionally, projects referred to a 'bottom line' (or financial feasibility) as the only metric of success of a project. With the increasing concern about environmental and social sustainability, Heschong et al. and others advocate a more holistic perspective on development and have defined a new 'triple bottom line'. This is known as People, Planet, Profit (or more recently, prosperity). These categories provide a framework for sustainable development by creating a bottom line for the social, environmental, and economical effects a project may have [15]. By taking this approach to the problem, we can define and group our own design objectives within this framework to make an attempt to approach the design problem integrally.

Creating a massing that has a high solar performance clearly relates to the environmental aspect of a project by enabling usage of the sun's energy in the form of passive heating and active (PV) energy generation, thereby saving on energy usage during the building's lifecycle. Researchers have argued that sustainable building corporations are more profitable [3], while other researchers find that the link between the socio-environmental sustainability of the building itself and its value is less pronounced but that there *is* an increase in premium price and rate of absorption [20] while others still have even found that there is no evidence supporting any of these claims [36]. Nevertheless, from Mangialardo et al. it can be inferred that even if the value itself of the building is not increased, the (economic) attractiveness of a project can be positively influenced by a more sustainable design development, both economically and socially.

From all of this, we can learn that while the direct financial gain from adopting the three bottom lines may be difficult to prove or quantify, it still is in the interest of the developer to invest in these aspects in order to increase the economic feasibility.

The characteristics of a building may influence multiple aspects of this triple bottom line theory but we can define the following objectives to emulate the approach a developer *may* have to such a project:

People:
Daylighting of the building & daylighting of the surroundings: Since as previously mentioned, daylight access increases student performance in school and generally increases health and mood [4; 15], it can be concluded that it is important to maximize the daylight access of the buildings' users as well as the residents and pedestrians in the surroundings of the building.

Planet:
PV energy yield of the building: Recent PV and PV planning developments ensure that net zero energy usage in urban contexts can be achieved under the right conditions [18]. Starting from the 1st of January 2021 (later delayed to 1st of July 2021), all newly built buildings in the Netherlands must be Bijna Energie Neutraal Gebouw (BENG) (Near Energy Neutral Buildings in Dutch) [BRON:rvo.nl]. These demands mean that (1) the maximum energy demand in kWh/m2.year of usage space AND the ratio between area of the facade and the heated area of the building, (2) the maximum amount of primary fossil fuel usage in kWh/m2.year of usage space and (3) the share of renewable energy % are all set at fixed values for each type of building. An example: for residential buildings with a facade area/heated area of less than 1.5, the maximum energy demand cannot exceed 55 kWh/m2.year, primary fossil fuel usage must be under 30 kWh/m2.year, and the share of renewable energy must be over 50% [25]. Practically, this means for us that we now know that new building projects need to take into account a multitude of demands that get stricter over time. For building planners this means that these demands have to be ensured in the design phase.
Heat retention of the building: As we know from the BENG demands mentioned above, the ratio between the area of the facade and the heated area of the building influences the ability of a (planned) building to pass the minimum requirements to be labeled (near) energy neutral. Another way to view this metric is the building's compactness. Compactness can mean multiple things: Catalina et al. handle it as relative compactness by taking the volume to surface ratio and comparing it to the most compact shape with the same volume, arguing that a more compact building loses less heat through its exposed surfaces. Others have simply analysed the ratio between facade area and volume and found a strong correlation between this value and the building energy consumption (r=0.91) [13]. This value only seems to hold up in harsh climates and the relation is less pronounced in milder climates.

Profit:
Floor Space Index of the building: The floor space index of a building plot is the total floor area of the building over the plot area. It is a direct indicator of the density of the urban fabric. Density of the urban fabric and land/real estate value are directly related according to [26]. This means that we can assume that it is in the interest of the developer to maximize the land usage.

[IMAGE:different FSI typologies]

The five objectives outlined above all have an optimal massing that correspond to the best performing building configuration. The exercise of the thesis is to find a method to generate a massing that performs well in all five of these criteria without compromising too much in any of the criteria. In the next section, each criterion as well as the total problem will be presented in formal terms.

2.5 PROBLEM STATEMENT

We can now move on to formalising the design objectives and afterwards the totality of the problem in mathematical terms. Starting with our most straightforward metric of performance, the Floor Space Index:

2.5.1 cost functions

The **floor space index** (FSI) of a building or massing can be calculated as follows:

$$FSI = \frac{\sum_{i=1}^{n} A_i}{A_{site}} \tag{2.1}$$

Where:

i indicates the index of the building floor
n indicates the total number of floors
A_i indicates the area of floor i in m^2
A_{site} indicates the total area of the site in m^2

For the **daylighting availability of the building**, and the **daylighting availability of the surroundings** we define the following four formulas:

$$L_{building} = \frac{\sum_{j=1}^{t} \sum_{k=1}^{p} V_{j,k}}{tp} \tag{2.2}$$

Where:

j indicates the index of the test point within the building
t indicates the total number of test points within the building
k indicates the index of the test point of the sky dome
p indicates the total number of test points in the sky dome
$V_{j,k}$ indicates the visibility status of the k_{th} point in the sky from the j_{th} test point

It should be mentioned that the result of this gives us an indication of what percentages of the sky we can see from each test point in the building and that this is a dimensionless number. Optionally, we can multiply the visibility status $V_{j,k}$ with the brightness of the corresponding point in the sky B_k giving us the formula:

$$L_{building} = \frac{\sum_{j=1}^{t} \sum_{k=1}^{p} V_{j,k} B_k}{t \sum B} \tag{2.3}$$

We have now taken into account the slight differences in sky brightness of the sky patches but remain dimensionless. At this point it should also be noted that we are not explicitly taking into account whether the visibility of the sky is blocked by the building itself. In a later section we will describe this problem in more detail. Similar to how we defined the daylight availability for the building, we can also formulate the daylight availability for the surroundings of the building.

$$L_{surroundings} = \frac{\sum_{l=1}^{s} \sum_{k=1}^{p} V_{l,k}}{sp} \tag{2.4}$$

Where:

l indicates the index of the test point on the surroundings
s indicates the total number of test points of the surroundings
$V_{l,k}$ indicates the visibility status of the k_{th} point in the sky from the l_{th} test point

As we have done previously, we can take the brightness of the sky into account by adding the brightness of point k in the sky as B_k:

$$L_{surroundings} = \frac{\sum_{l=1}^{s} \sum_{k=1}^{p} V_{l,k} B_k}{s \sum B} \tag{2.5}$$

For the building compactness, we have previously identified two options for calculating compactness: the simple method of floor area over surface area (shape coefficient or building coefficient C_f) which works for harsh climates used by Depecker et al. and the more advanced relative compactness R_c of the building compared to the most compact shape with the same volume used by Catalina et al..

$$C_f = S_e / V_b \tag{2.6}$$

[13]

Where:

S_e corresponds to the surface of the envelope
V_b corresponds to the inner volume of the building

And:

$$R_c = 6 \times V_b^{\frac{2}{3}} \times S_e^{-1} \tag{2.7}$$

[10]

Where:
S_e corresponds to the surface of the envelope
V_b corresponds to the inner volume of the building

We can see the similarities between the two methods clearly. Method 2 (R_c) is used in the remainder of the thesis to guarantee accuracy of results but within the context of a Dutch climate, method 1 (C_f) would certainly be valid as well in most cases.

For the PV potential we define the following two formulas:

$$P_{building} = \frac{\sum_{m=1}^{r} \sum_{q=1}^{h} V_{m,q} D_q}{rh} \tag{2.8}$$

Where:
m indicates the index of the test point on the roof of the building
r indicates the total number of test points on the roof area of the building
q indicates the index of the test point of the solar positions (hours of the year)
h indicates the total number of test points in the solar positions (hours of the year)
$V_{m,q}$ indicates the visibility status of the q_{th} solar position from the m_{th} test point

As we have done earlier with the daylight visibility, we can include weights for the direct normal irradiation of point q by including a value D_q into the equation:

$$P_{building} = \frac{\sum_{m=1}^{r} \sum_{q=1}^{h} V_{m,q} D_q}{r \sum D} \tag{2.9}$$

2.5.2 standard integer programming formulation

If we now want to optimize any configuration, it is desirable to bring it into a standard form. For the entire configuration, if we want to define the problem as a mathematical programming that we can quickly solve using for example the simplex method, we need to bring it to this form [21]:

Maximize:
$f(x_1, x_2, ..., x_n) = c_1x_1 + c_2x_2 + \ldots + c_nx_n$

subject to:
$a_{11}x_1 + a_{12}x_2 + ...a_{1n}x_n \leq b_1$
$a_{21}x_1 + a_{22}x_2 + ...a_{2n}x_n \leq b_2$
$\vdots$
$a_{m1}x_1 + a_{m2}x_2 + ...a_{mn}x_n \leq b_m$

$x_1, x_2, ..., x_n \geq 0$

Where:
f is the objective function
x is the decision variable
n is the total number decision of variables
c is the cost (or gain) value
a is the constraint value
m is the total number of constraints
b is the constraint boundary

In matrix form this becomes:
$\mathbf{x} = (x_1, x_2, ..., x_n)^T$
$\mathbf{c} = (c_1, c_2, ..., c_n)^T$
$\mathbf{b} = (b_1, b_2, ..., b_m)^T$

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \ddots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

This can then be written as:

Maximize: $\mathbf{c} \cdot \mathbf{x}$
subject to: $\begin{cases} \mathbf{Ax} \leq \mathbf{b} \\ \mathbf{x} \geq \mathbf{0} \end{cases}$

It is important to note that this notation corresponds to the standard notation of a (integer) programming problem and represents the objective score for the entire configuration and not its constituent parts. If we try to incorporate the cost functions as described in the previous section into this framework, we can quickly see how the problem is not linear nor straightforward to solve.
Lets assume that we discretize the totality of the potential building volume available to us into n number of voxels that can represent the presence or absence of building mass and define a variable x for each of these voxels. This gives us the following decision variables:

$0 \leq x_1, x_2, ..., x_n \leq 1$

and

$x \in \mathbb{Z}$

After all, a building either occupies a space or it does not occupy that space. The problem is now a (binary) integer programming problem. If we can discretize the volume into sufficiently small parts, we have a suitable resolution for our potential massing.

When we now attempt to apply this approach to our objective and cost functions, we quickly run into practical problems. We have shown what our cost functions should ideally look like in equations 2.1 through 2.9 and when we rewrite the cost functions to include these binary decision variables as in the standard formulation, we find that the occupation status of a voxel may influence the cost value of any of the other voxels, see image **??**:

It is theoretically possible to overcome this problem by adding auxiliary constraints. The nonlinearity problem can be solved by piecewise linearization and by employing the big-M method, we could theoretically add conditional constraints on all variables that sets the cost values for the daylighting performance and solar performance variable c_n to 0 when one of the variables that influences variable x_n is in an active state (i.e. one of these variables holds value 1) [2]. This requires an exponentially increasing number of constraints when the resolution of the model is increased by adding variables (voxels) that could all potentially block rays coming from the sky or sun towards the target voxel. Furthermore, if this method successfully were to be applied, a final problem arises. Lets define the objective function by simply adding the costs together as follows:

$f(x) = \sum_{i=0}^{n} x_i(c_{1i} + c_{2i} + c_{3i} + c_{4i} + c_{5i})$

Where:
c_1 is the FSI cost
c_2 is the building daylighting potential cost
c_3 is the surroundings daylighting potential cost
c_4 is the heat retention cost
c_5 is the PV potential cost
n is the number of voxels

This would theoretically give us (when the nonlinearity and interdependency problems are solved) a workable programming. However we are comparing 'apples to oranges' in this case. For instance, an increase of 10% past the required value in FSI is desirable but if the added square meterage also causes the other four criteria to lose a 2% points, the new solution is formally speaking *more optimal*, but would likely not be considered better in the eyes of an architect, client, or developer. More possibilities for weighting and normalization are required.

What we can conclude from the previous section is that the problem we are facing is actually not linear in nature and is not straightforward to solve. An alternative method for generating a solution to the problem will need to be found. Furthermore, the target values (goal) and weight for each objective are not included in this formulation. A more nuanced approach to the problem is desired that can find (near) optimal solutions, ensures more control on the scalability and guarantees realistic computation times. Due to the large solution space, implementation and solving difficulty has to be limited by dealing with the interdependency problem without applying a prohibitive amount of constraints and a heuristic is looking to be a more attractive approach to solving the problem. In the next chapter (3), we describe how the objectives are actually integrated into the method in more detail.

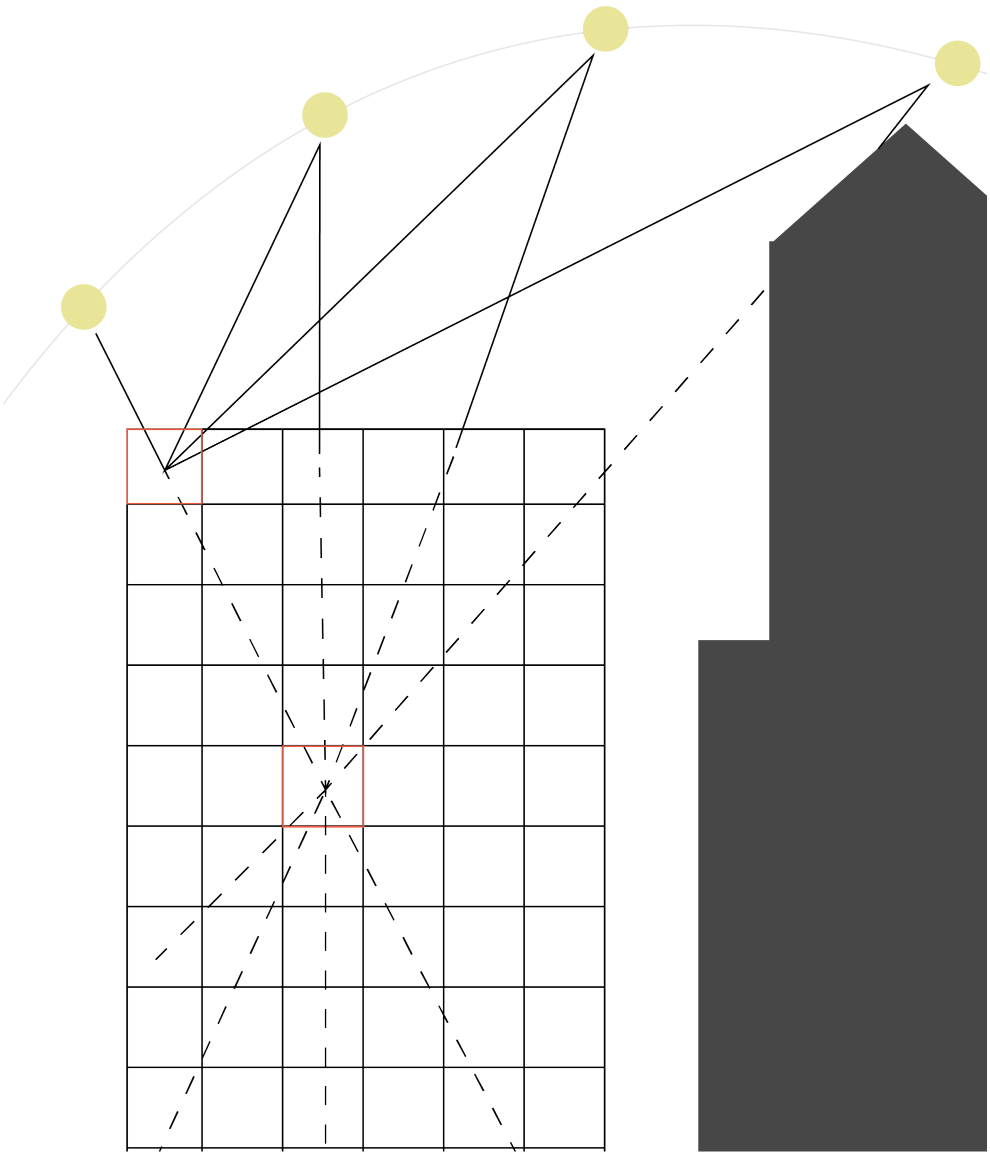


Figure 2.6: The interdependency problem. If we want to determine the value of a voxel (in red) we need to know how much sun it catches, as well as how much it blocks other voxels. However, if a voxel does or does not exist(i.e. it's occupation status is 1 or 0), the voxels that influence it and are influenced by it need to have their score updated as well. This in turn might again influence the value of the original voxel.

3 METHOD

We will now describe the implemented method to generate optimal configurations. In order to explain the method design process best, the subproblems will be solved before proposing the complete solution to the problem. To do this, the chapter is structured as follows: First, we define toy problems to illustrate how each design objective of the optimization is related to its spatial configuration. This is done through the creation of several toy problems that will be solved on a smaller scale. The final toy problem shows how we can utilize the cost functions to feed into a solver that outputs the configuration. These methods are then brought to 'real' scale for the case study location in chapter 4. In the final section of each toy problem, an overview is given of the computation and simulation speeds along with a visualisation of the result.

3.1 TERMINOLOGY

Before we can move on to the core of the implementation details, it is important to define some terms and make clarifications on what the words that often return in the chapter mean. Certain definitions are fuzzy and can be used interchangeably, while other specialized terms have no clear bounds. The following glossary attempts to elucidate the meaning of all of these concepts to help understand what is meant by them.

3.1.1 generative design

- Lattice: A numerical field within a discrete 2D or 3D space. Represented as an array of values.
- Voxel: A portmanteau of volume and element, like a pixel in 3D. In our work: a single value within the lattice array.
- Mass: A group (array) of voxels representing the (maximum) volume of the building.
- Configuration: An ordering (in 2D or 3D space) of voxels
- Massing: The process of obtaining the mass from a given design space.
- Envelope: Boundary of the mass
- Zone: A group of voxels within a mass representing a specific attribute such as spatial function.
- Zoning: The process of obtaining the zone from a given mass.
- Spatial agent: An autonomous entity which acts upon an environment by directing its behaviour towards achieving its spatial goals.
- Agent Behaviour: The actions which the agent can perform to achieve its goals.
- Stencil: The neighborhood definition in a discrete 2D or 3D space.

- Environment field: Scalar, simulated values on a 2D or 3D grid with the same structure as the voxel lattice.

3.1.2 decisionmaking

- Objective: The goal of the decision-maker. The objective consists of at least two design criteria.
- Decision variable: A quantity that the decision-maker controls.
- Decision space: The range of values these variables can take on.
- Global decision variable: A decision variable that affects the performance of the entire design space.
- Local decision variable: A decision variable that affects the performance of an aspect of the design space.
- Design criteria: Physical attributes of the final design.
- Design variable: A decision variable specifically applied to one of the design criteria.
- Design space: The range of values that these variables can take on.
- Performance indicator: An aggregated value that quantifies the performance towards a design criterion.
- Performance estimator: An estimated value of the performance towards a design criterion.
- Global performance indicator: A performance indicator that informs the state of the configuration as a whole concerning a certain design criterion.
- Local performance indicator: A performance indicator that informs the state of a segment of the configuration concerning a certain design criterion.
- MCDA: The evaluation of multiple conflicting criteria.
- MCDM: The decision-making on multiple conflicting criteria.
- Function: Mathematical function.
- Spatial function: Space usage or occupation type (in an architectural sense).
- Cost function: Mathematical function for calculating scalar values of a voxel towards a design criterion. 'Cost' is used to calculate desirable (for example sunlight received) as well as undesirable (for example sunlight blocked) performance indicator values.
- Objective function: Mathematical function for calculating the performance of a configuration towards the objective. This function combines at least two cost functions.
- Heuristic: A method to find an approximate solution to a problem
- Metaheuristic: A procedure to find or create a heuristic to solve a problem

3.1.3 simulation

- Radiance: The density of radiant flux per unit of surface area and unit of solid angle. W/sr/m2
- Irradiance: The density of radiant flux per unit of surface area. W/m2
- Luminance: The luminous intensity per unit of surface area. Candela/m2.
- Illuminance: Total luminous flux per unit of surface area and unit of solid angle. Lux.
- Visibility: The unobstructed view towards a target point from a point of interest.
- Closeness: The distance (euclidean) between two points in 2D or 3D space.
- Potentials: The ratio between the current performance of a configuration towards a design criterion and the maximum performance towards that design criterion.

3.1.4 clarifications

Now that these definitions have been clarified it is important to note a few things. The usage of MCDA, MCDM, MADM (multi-attribute decision making) and MODM (multi-objective decision making) is often confused in literature and practice. MCDA is the most commonly used term and covers the broadest meaning. To keep the language simple, from now on MCDA will be used for all of these uses i.e. the analysis and making of the decisions.

While heuristics are generally specific to a certain problem, metaheuristics are more general by nature and can be applied to a larger number of problems. Exploring the subtle differences and similarities between the two definitions can be an entirely separate subject of study. We will therefore refer only to heuristics from now on without diving into the details of correct terminology. When heuristic is used from now on, we refer to a: Non problem-specific strategy that guides the search process by efficiently exploring the decision space to find near-optimal solutions. [6].

Regarding the definitions of design criteria and decision criteria, design variable and decision variable, design space and decision space, cost function and performance indicator/estimator, mass, envelope and configuration, MCDA/MCDM/MODM/MADM and heuristics/metaheuristics, and design and decision space: These many definitions with only subtle differences tend to be confusing. From the literature studies, it seems the terms are often used interchangeably as well, adding to the muddle of words. It is therefore now important to refer back to the problem statement and generally chapter 2 for clarification. To reiterate and simplify: the goal of the research is to design a methodology for generating building **configurations** that perform well in regards to certain **design criteria**, by employing **MCDA** techniques. To do this, we need to define **cost functions** and an **objective function** that we can feed into a solver that can use a **heuristic** to generate outputs. The next section gives an illustration of this methodology design by the usage of toy problems.

3.2 TOY PROBLEMS

All coding of the problems as well as the test case application has been done in Microsoft VSCode using Jupyter notebooks. To ensure open access and reproducibility,

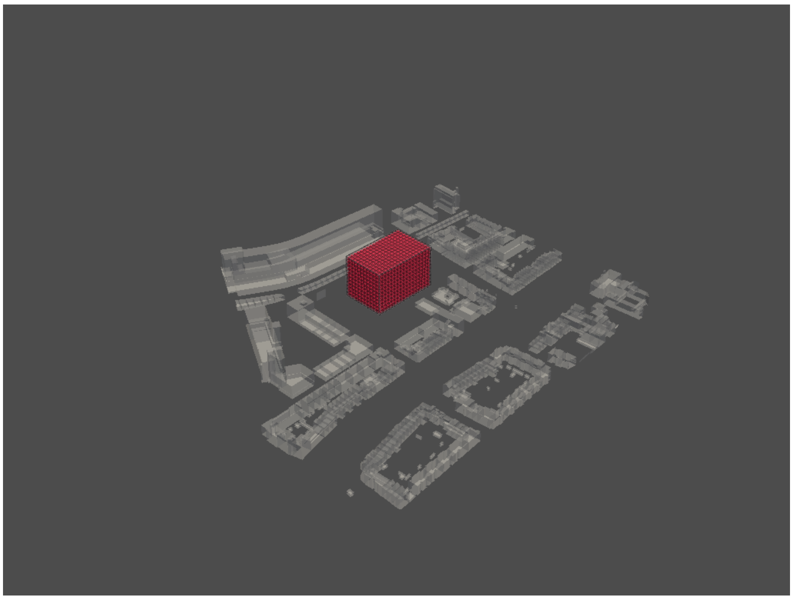


Figure 3.1: The output of the first toy problem: In red a simple 5x5x5 'rubik's cube' with voxels of size $15m^3$ and in light gray the imported environment.

the Thesis Repository is hosted on GitHub. This work builds on a structure called TopoGenesis which is *an open-source Python package that provides topological structures and functions for Generative Systems and Sciences for various application areas such as:*

- generative design in architecture and built environment
- generative spatial simulations
- 3D image processing
- topological data analysis
- machine learning

This framework allows for the quick and easy construction of lattices and enables us to use efficient methods and libraries such as NumPy arrays for the preparation and processing of our functions in 3D. [5]

3.2.1 T.P.1 environment and voxelization

The first toy problem concerns the preparation of the environment and voxelization of the maximum building volume. The user input is a mesh that represents the planning area, the dimensions and location of the building's maximum boundaries, and the number of divisions for the voxelization (i.e. the resolution of the method to be applied). In the example below (3.1), this is set as a cube with dimensions of 75x75x75 m, a division of 5, creating as output 125 voxels of 5x5x5 m as can be seen in figure 3.1.

3.2.2 T.P.2 skydome

The next toy problem concerns the creation of a skydome that is populated with points that can later be used for the calculation of the daylight availability. The is achieved by simply creating a sphere around the scene, defining a number of subdivisions which is set at 3 in this example, extracting the vertices, and then removing all vertices with a negative z-coordinate. The number of vertices increases

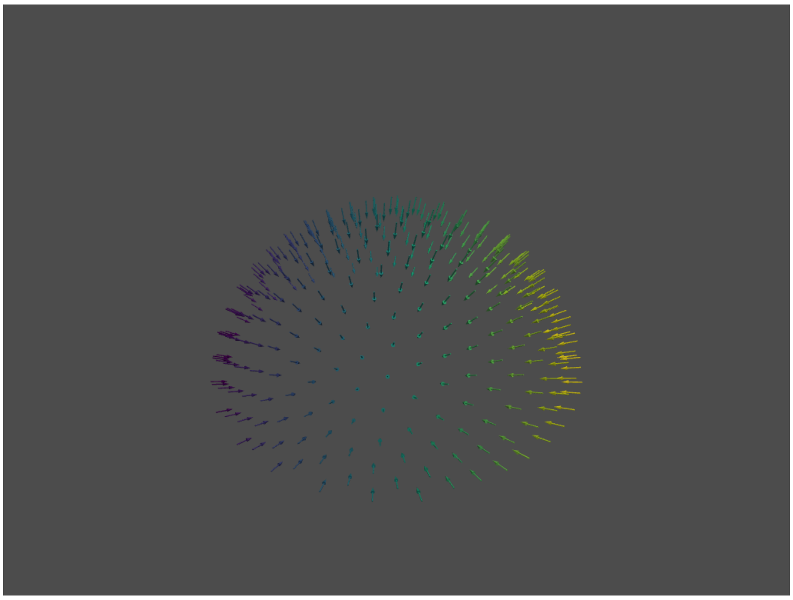


Figure 3.2: The output of the second toy problem: an array of points on a dome above the scene, here represented by coloured arrows pointing towards the centre of the lattice.

by $4^{s}ubdivisions$. The output is an array of points that will be used to calculate the daylight potential, see figure 3.2.

3.2.3 T.P.3 solar path

The next output we need is an array of points for the positions of the sun throughout the year. The user inputs a range of the year for which the sunpath is simulated. In this example, we use a period of the 21_{st} of March till the 21_{st} of September, approximately corresponding to the spring and fall equinoxes. Another input that is required is a weather file of the local climate, in this case a .epw file for Amsterdam is used. A location longitude (4.3571) and latitude (52.0116) is also needed. We then call the Sunpath function from Ladybug, an environmental simulation tool, to actually generate the hours of year we need. Each week for every sun position with a z-coordinate above 0, the current hour of the day, along with the solar position, direct normal radiation, direct normal illuminance, and global horizontal illuminance are outputted into arrays. For the settings mentioned above, this yields a total of 643 test points, see also figure 3.3.

3.2.4 T.P.4 daylight potential of the lattice

Now the cost functions are ready to be defined. Starting with the daylighting potential of the lattice, we will now need to deal with the interdependency problem as described in section 2.5. Florou has solved this issue by employing a method where she constructs a nested array called an 'interdependency graph' which is a collection of hits for each voxel through each voxel towards each test point. We precompute all the hits between each origin, the context, and the rest of the lattice itself. This means that the computationally expensive ray tracing only needs to be run once for each objective, the graph then being stored for later use during optimization. The cost function for daylight potential is simplified by ignoring the slight differences in brightness for the skydome and considering instead what percentage of the sky

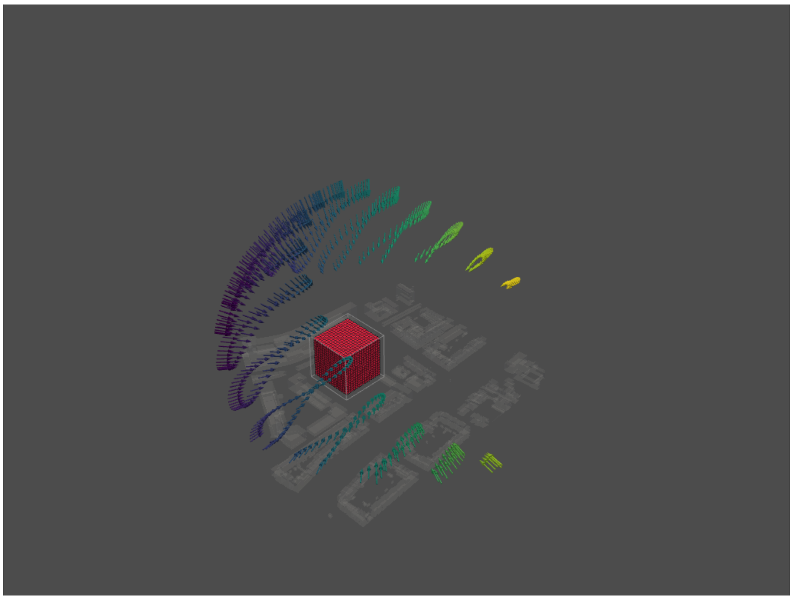


Figure 3.3: The output of the sun pathing toy problem: the arrows point towards the centre of the lattice.

that is visible with the current configuration.

To achieve this, the equation 2.2: $L_{building} = \frac{\sum_{j=1}^{t} \sum_{k=1}^{p} V_{j,k}}{tp}$ can be changed to an equation for the sky view factor for usage within the TopoGenesis framework with voxels.

First, create two arrays:

$\mathbf{a}$ of size n that sums the amount of times a voxel is blocked by another voxel
$\mathbf{b}$ of size n that sums the amount of times a voxel blocks another voxel

$$\mathbf{c} = \frac{(\mathbf{a} \otimes \mathbf{b}) - \min(\mathbf{a} \otimes \mathbf{b})}{\max(\mathbf{a} \otimes \mathbf{b}) - \min(\mathbf{a} \otimes \mathbf{b})} \tag{3.1}$$

Where:
n represents the number of voxels $\mathbf{c}$ is an array of size n that gives us a normalized value for the amount of times a voxel blocks another voxel, or is blocked by another voxel, in regards to the test points in the sky. The results can be seen in figure 3.4

3.2.5 T.P.5 sky view potential of the surroundings

Similar to the previous toy problem, we can also calculate the intersections between test points on the street around the lattice and the lattice and context in relation to the sky dome. Because in this case the interdependency does not play a role, we take a slightly different approach to the actual score calculation.

First, we calculate what percentage of the sky can be seen from each test point if there were no obstruction, creating an array $\mathbf{d}$ of size m, where m represents the number of test points. 24 test points directly next to the lattice were defined, while the same skydome as in the previous section was used. Then we create an array $\mathbf{e}$ of size n, m that corresponds to the amount of times a voxel blocks a sky patch for each test point. We can then take the dot product of these arrays to achieve a

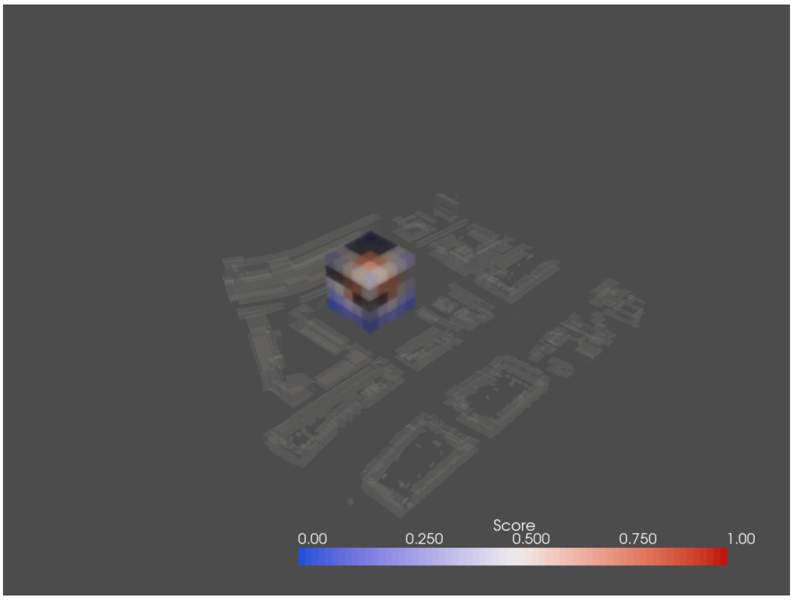


Figure 3.4: The normalized values for the daylight potential mapped onto the lattice. Red voxels have a high cost, meaning they block (and are blocked) relatively often.

weighted value for how much sky each voxel obstructs. This is then normalized to achieve the cost function f:

$$\mathbf{f} = \frac{\mathbf{e} \cdot \mathbf{d} - \min(\mathbf{e} \cdot \mathbf{d})}{\max(\mathbf{e} \cdot \mathbf{d}) - \min(\mathbf{e} \cdot \mathbf{d})} \tag{3.2}$$

f is an array of size n that gives us a normalized value for the amount of times a voxel blocks the view towards the sky from the street view around the lattice, in regards to the test points in the sky as can be seen by figure 3.5

3.2.6 T.P.6 pv potential of the lattice

The final visibility objective implemented is the PV potential of the lattice. This objective uses the weights for the direct normal radiation calculated in the third toy problem. The target is the sun path as defined in the same toy problem. We once again make a slight adaptation to the formula defined in chapter 2.5:

First, create two arrays:

g of size n that sums the amount of times a voxel is blocked by another voxel in relation to the solar positions
h of size n that sums the amount of times a voxel blocks another voxel in relation to the solar positions

$$\mathbf{p} = \frac{(\mathbf{g} \otimes \mathbf{h}) - \min(\mathbf{g} \otimes \mathbf{h})}{\max(\mathbf{g} \otimes \mathbf{h}) - \min(\mathbf{g} \otimes \mathbf{h})} \tag{3.3}$$

p is an array of size n that gives us a normalized value for the amount of times a voxel blocks another voxel, or is blocked by another voxel, in regards to the solar path points, relative to the intensity in w/m^2 of those solar points for each predefined hour of the year, see figure 3.6.

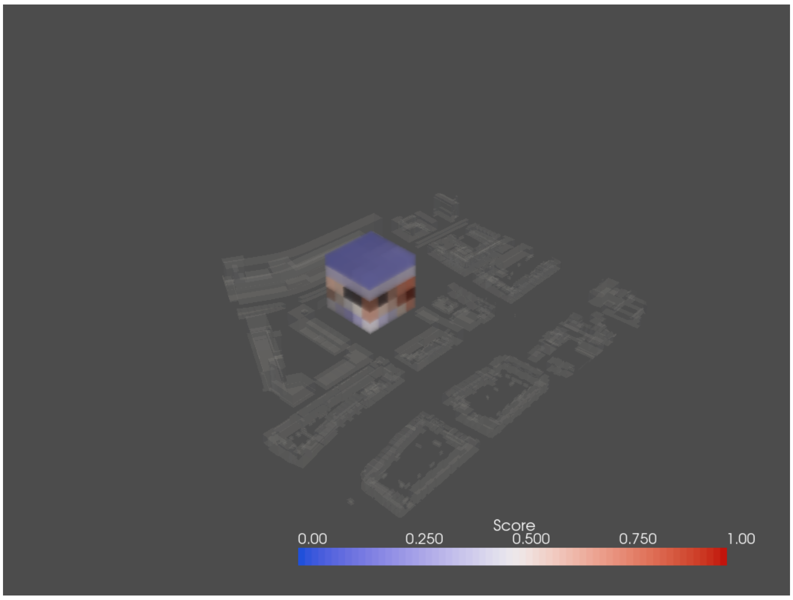


Figure 3.5: The normalized values for the sky view potential from street level. Note how the voxels at the top are blue, indicating they perform well. This is likely due to an error in the resolution that will be discussed in chapter 4.

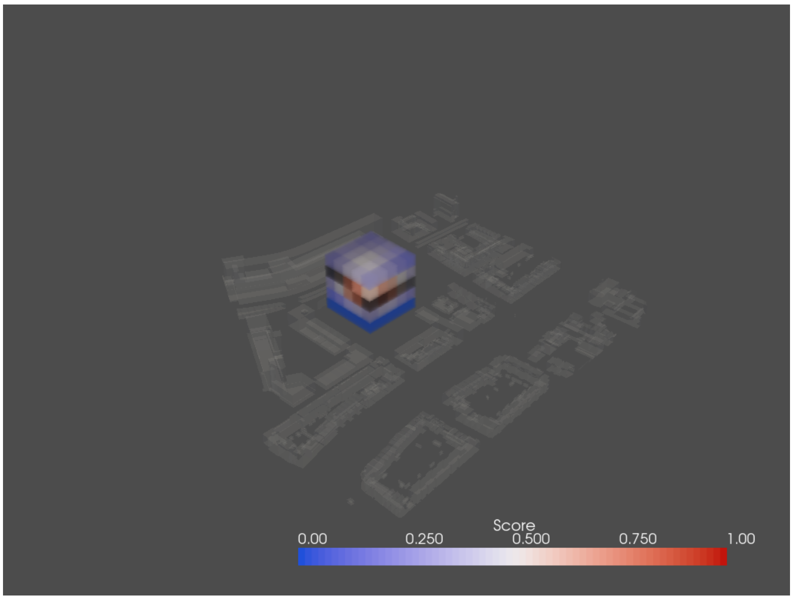
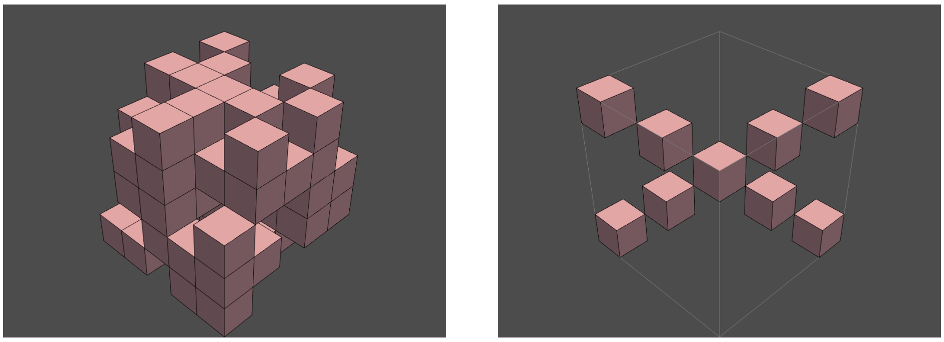


Figure 3.6: The normalized values for the PV potential with weights for the sun intensity in w/m^2.



(a) (b)

Figure 3.7: (a) This is a randomly generated configuration. The score according to the simple method is: 0.197 while according to the advanced method the score is: 1.956. (b) The score of this edge case configuration according to the simple method would be 0.37, while according to the advanced method, the score is 1.93

3.2.7 T.P.7 compactness example

As mentioned in chapter 2, there are two methods for calculating the relative compactness of a configuration that are both valid to some extent. For the purpose of demonstrating the difference in results between the two methods, a random configuration is generated and then analysed.

The first method according to Depecker et al.:

$C_f = S_e/V_b$

Where:

S_e corresponds to the surface of the envelope
V_b corresponds to the inner volume of the building

The second method according to Catalina et al.:
$R_c = 6 \times V_b^{\frac{2}{3}} \times S_e^{-1}$

Where:

S_e corresponds to the surface of the envelope
V_b corresponds to the inner volume of the building

When we generate random configurations to compare the results from these calculations, it is found that the output of both methods generally relate to each other in a 1:10 ratio. The first method gives values between

When the configuration becomes more extreme, the ratio between the two methods for computing compactness is no longer 1:10 The relative compactness is a more reliable method for edge cases such as these.

3.2.8 T.P.8 configuring

Now we can apply the MCDA to find an optimal solution according to the cost functions described above. A final design criterion that we have not mentioned in this chapter yet is the Floor Space Index. Adapting the formula for calculating the

	NSGA-II						IHS					
	speed (s)	**c1**	**c2**	**c3**	**c4**	**c5**	speed (s)	**c1**	**c2**	**c3**	**c4**	**c5**
3.9a,3.9b	1.9	-15.46	-20.26	-0.6	-16.71	-2.83	0.1	-15.13	-21.79	-0.74	-20.24	-2.22
3.9c,3.9d	7.3	-37.10	-49.58	-0.45	-43.52	-0.04	0.9	-16.52	-23.10	-0.74	-19.58	-2.2
3.9e,3.9f	2.2	-30.63	-39.25	-1	-27.19	-1.35	0.1	-30.633	-39.25	-1	-27.19	-1.35
3.9g,3.9h	1.1	-31.14	-39.05	.	-34.19	-1.19	0.2	-16.14	-22.03	.	-19.85	-2.28
3.10a,3.10b	0.3	-37.53	-50.11	.	-43.75	.	0.7	-37.53	-50.11	.	-43.75	.
3.10c,3.10d	0.2	-37.28	-49.64	.	.	.	0.6	-29.69	-38.12	.	.	.
3.10e,3.10f	1.4	-16.41	-21.42	.	.	-2.79	0.1	-17.86	-23.28	.	.	-2.20
3.10g,3.10h	0.9	-29.92	.	-1	.	-1.44	0.1	-15.93	.	-0.66	.	-2.36

Table 3.1: Toy problem optimization results: speed and objective values for PV potential **c1**, daylighting potential **c2**, FSI **c3**, Sky view Factor potential **c4**, relative compactness **c5**

FSI of a configuration is very straightforward :

$$FSI = \frac{\sum_{i=0}^{n-1} V_i A}{A_{site}}$$

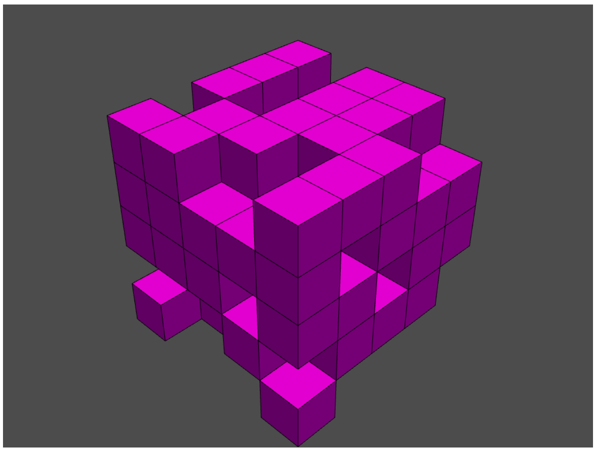
Where:

i indicates the index of the voxel
n indicates the total number of voxels
V_i indicates the occupation status of voxel i
A indicates the area of a voxel in m^2
A_{site} indicates the total area of the site in m^2

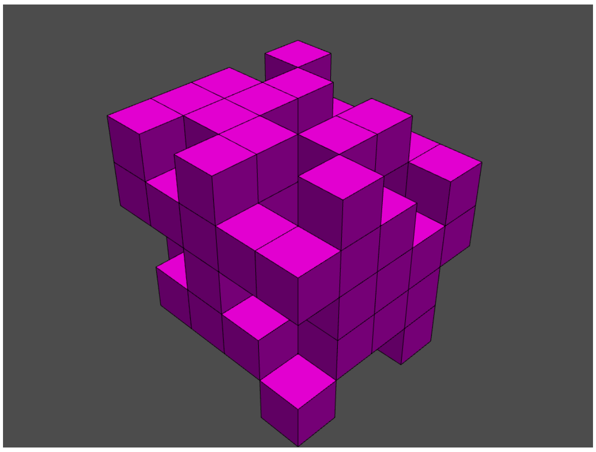
For optimization, the PyGmo library is used. This library contains many algorithms, but for demonstrative purposes the only algorithms shown in this chapter are the NSGA-II, Imporved Harmony Search (a different type of genetic algorithm), and Multi-objective Hypervolume-based Ant Colony Optimization, a type of MACO. For the dimension of the decision variables x we use n (the total number of voxels. The bounds are set between 0 and 1 and it is specified that there are n number of integer variables. The number of generations g is set at 150, while the starting population size p is set at 8, meaning there will be 8 individuals used for crossover and mutation. Pygmo can accept the cost functions we have defined earlier and we now simply run the optimization and extract the results.

Below, an overview 4.1,3.10 shows the different combinations of settings and objectives with their respective configurations. Table 3.1 gives the corresponding performance indicator values and computation speeds. We can learn a few things from these results. First, when the target FSI is removed as an objective, or when the number of generations is set at a high number, NSGA-II will fill in the entire lattice, such as happened in # 3.9c, 3.10a and 3.10c. This is likely due to the cost functions for PV potential and daylight potential being biased towards growing as much as possible, while the FSI objective only wants to grow until it has reached its desired value. When applying another algorithm, MACO, with the same settings (albeit with a larger starting population since that is required for MACO), we see the result from table 3.2. The output seems to generate compact configurations with shapes that generally make sense, however it is difficult to tell what is going on under the hood. Increasing the number of generations and the starting population does give increasingly better performing configurations as can be seen from the table, although computation time quickly runs out of hand.

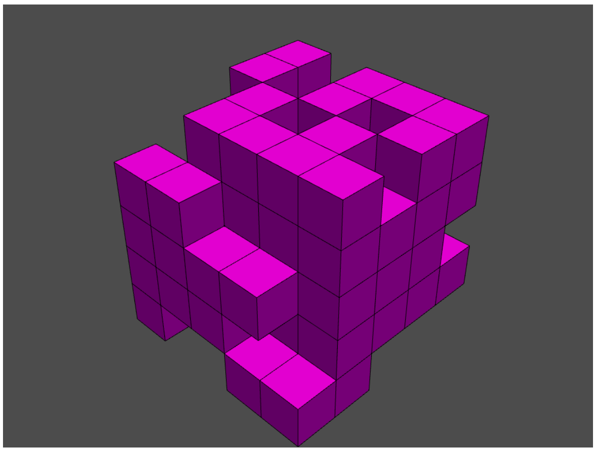
From these quick toy problem simulations, we can conclude several things. First, the question that is asked of the solver informs the answer: it is important to take a critical look at the cost functions since these determine the outcome of the optimization. Some cost functions seem to dominate when using the wrong objective function. A method of finding the correct weights for each objective is therefore rec-



(a)



(b)



(c)

Figure 3.8: the resulting configurations when applying ant colony optimization. (a) has 100 generations and a population of 100, (b) has 1000 generations and a population of 100, while (c) has 1000 generations and a population of 1000. For all three runs, FSI was ignored as an objective.

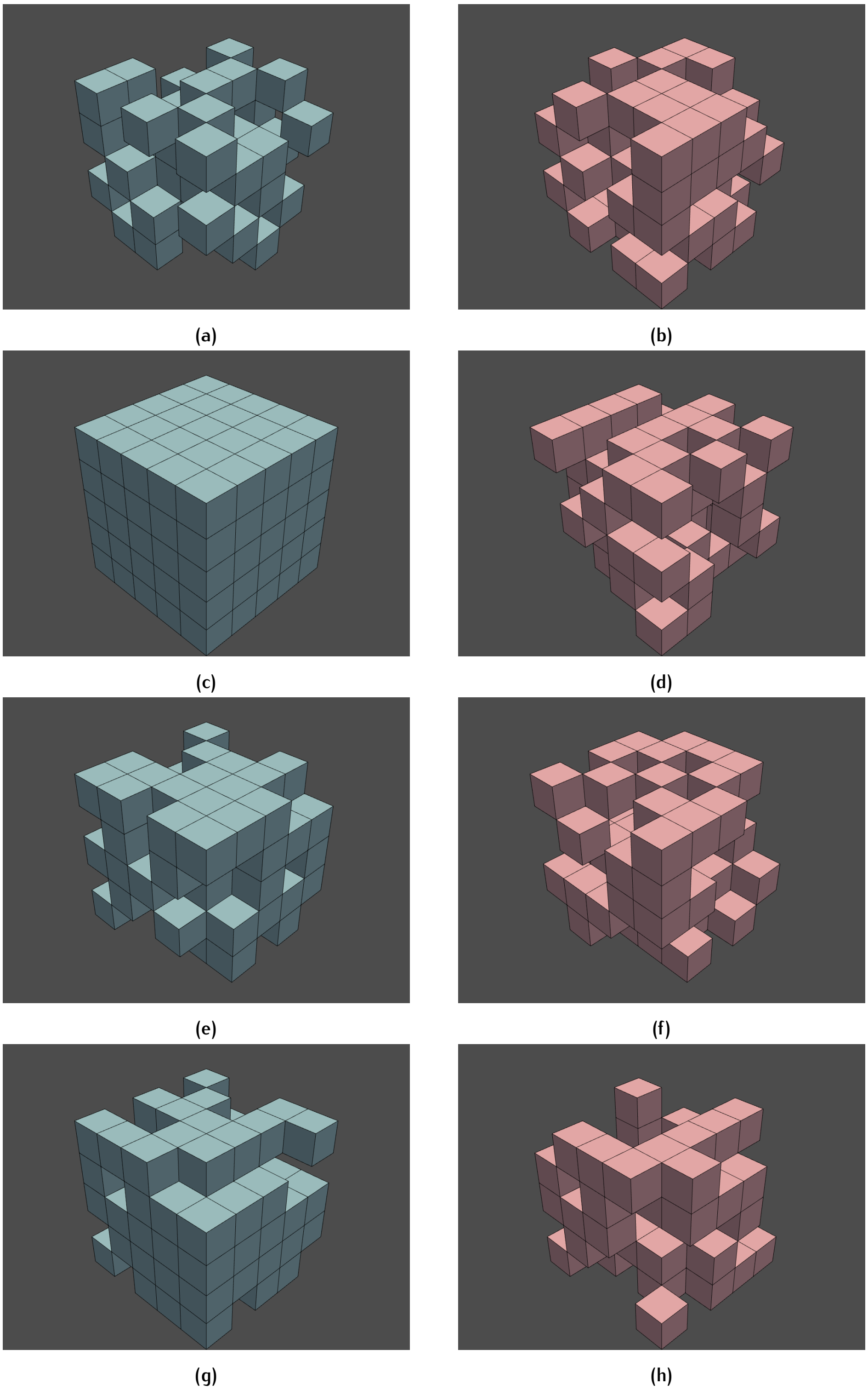


Figure 3.9: (a) NSGA with 5 objectives, 150 generations, 8 initial population. (b) IHS with 5 objectives, 150 generations, 8 initial population. (c) NSGA with 5 objectives, 1000 generations, 8 initial population. (d) IHS with 5 objectives, 1000 generations, 8 initial population. (e) NSGA with 5 objectives, 150 generations, 16 initial population. (f) IHS with 5 objectives, 150 generations, 16 initial population. (g) NSGA without FSI as objective, 150 generations, 8 initial population. (h) IHS without FSI as objective, 150 generations, 8 initial population.

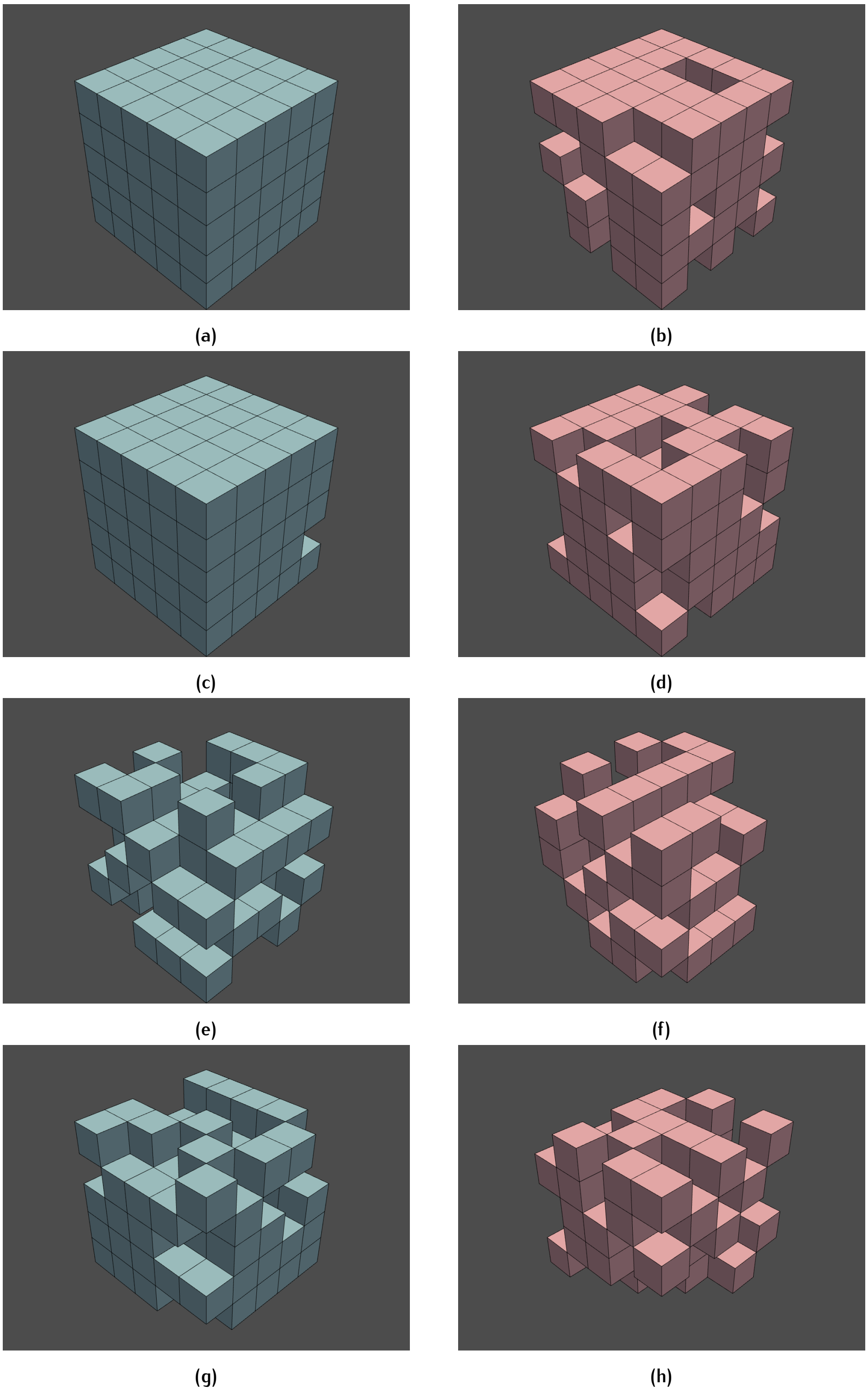


Figure 3.10: (a) NSGA without FSI and compactness objectives, 150 generations, 8 initial population. (b) IHS NSGA without FSI and compactness objectives, 150 generations, 8 initial population. (c) NSGA with only daylighting potential and PV potential as objectives, 150 generations, 8 initial population. (d) IHS with only daylighting potential and PV potential as objectives, 150 generations, 8 initial population. (e) NSGA with daylighting potential, PV potential, and relative compactness as objectives, 150 generations, 8 initial population. (f) IHS with daylighting potential, PV potential, and relative compactness as objectives, 150 generations, 8 initial population. (g) NSGA with FSI, PV potential, and relative compactness as objectives, 150 generations, 8 initial population. (h) IHS with FSI, PV potential, and relative compactness as objectives, 150 generations, 8 initial population.

	MACO speed (s)	**c1**	**c2**	**c3**	**c4**	**c5**
3.8a	8.7	-28.03	-35.62	.	-25.34	-1.62
3.8b	89.5	-29.44	-37.16	.	-28.61	-1.37
3.8c	974.3	-29.29	-37.16	.	-33.50	-1.26

Table 3.2: Ant colony optimization without targeting FSI: speed and objective values for PV potential **c1**, daylighting potential **c2**, Sky view Factor potential **c4**, relative compactness **c5**

ommended, but this does make the generalization of such solutions more difficult since this may change according to what objective is being persued. When picking the algorithm, MACO and IHS seem to perform with the least non-sensical results. IHS also seems to perform fastest of the three methods tested. In the next chapter, some improvements to the method in regards to the objective functions will be presented, as well as on a larger scale, smaller resolution implementation. The chapter will conclude with the most important lessons learned and suggestions for further development of the model.

4 | EVALUATION

In the previous chapter, we described the methods that can be used for calculating optimal massings in regards to certain objectives. The toy problems give an illustration of the techniques used, but the accuracy and validity of the results is difficult to assess at such a low resolution. For validation, a test case implementation and adaptation of the methods as described in the previous section is presented. The environment remains the same, but the size of the voxels has been reduced to 5x5x5 *m* to attempt to gain more accurate results. The next section describes the implementation of the methods explored in the previous chapter and the lessons learned from said implementation

4.1 PERFORMANCE

Computation of the high resolution lattices needs a large amount of memory which might not be acceptable. The current working value is a lattice of 35x35x50 m with a voxel size of 5, creating 490 voxels. Computation of the collisions with the skydome takes around 6 minutes of computation time while computation of the collisions with the sunpath takes 7 minutes, but the number of test points for the sunpath had to be reduced to biweekly values due to not enough memory being available. Around 400 test points and around 500 voxels seems to be the current limit to the problem size. Increasing the number of voxels means a tradeoff in the amount of points in the sky to trace towards, otherwise the computation time or memory usage becomes prohibitive. By using the interdependency graphs however, the computationally expensive simulations only have to be performed once and not during the (also expensive) optimization process. When optimizing, the following objectives are pursued: PV potential, daylight potential, FSI, and relative compactness. The algorithm used is the improved harmony search since this has previously shown to give the best and fastest results. The optimization takes a population of size 8, 150 generations and needs 45 seconds to produce the results seen in 4.1a:

The results are still a bit confusing: we can see the two objectives for daylighting and PV potential are in conflict with each other. For optimal daylight performance, the rays need to penetrate deep into the building, while for optimal PV performance, the top of the columns of voxels must be exposed as much as possible to the rays of the sun. The compactness of the building and FSI of the building seem to have no influence on the final configuration. We will now look at the objective functions individually:

When we remove one visibility objective from the equation (in this case the daylighting), with all other matters remaining equal, computation now takes 22 seconds. The scores are as follows: 0.674, 0.083, 0.035. We have previously normalized the PV potential score by dividing with the maximum number of hits that can theoretically be achieved. As can be seen by the scores from **c1**, **c3**, and **c5**, values for the compactness and FSI are very low, which is reflected by the configuration of 4.1b.

For the sake of experimentation, we have removed this previous normalization to produce the following configuration 4.1c with scores of 33491 for **c1**, 0.075 for **c3**, and 0.038 for **c5**. Due to the high objective value of the daylighting, the com-

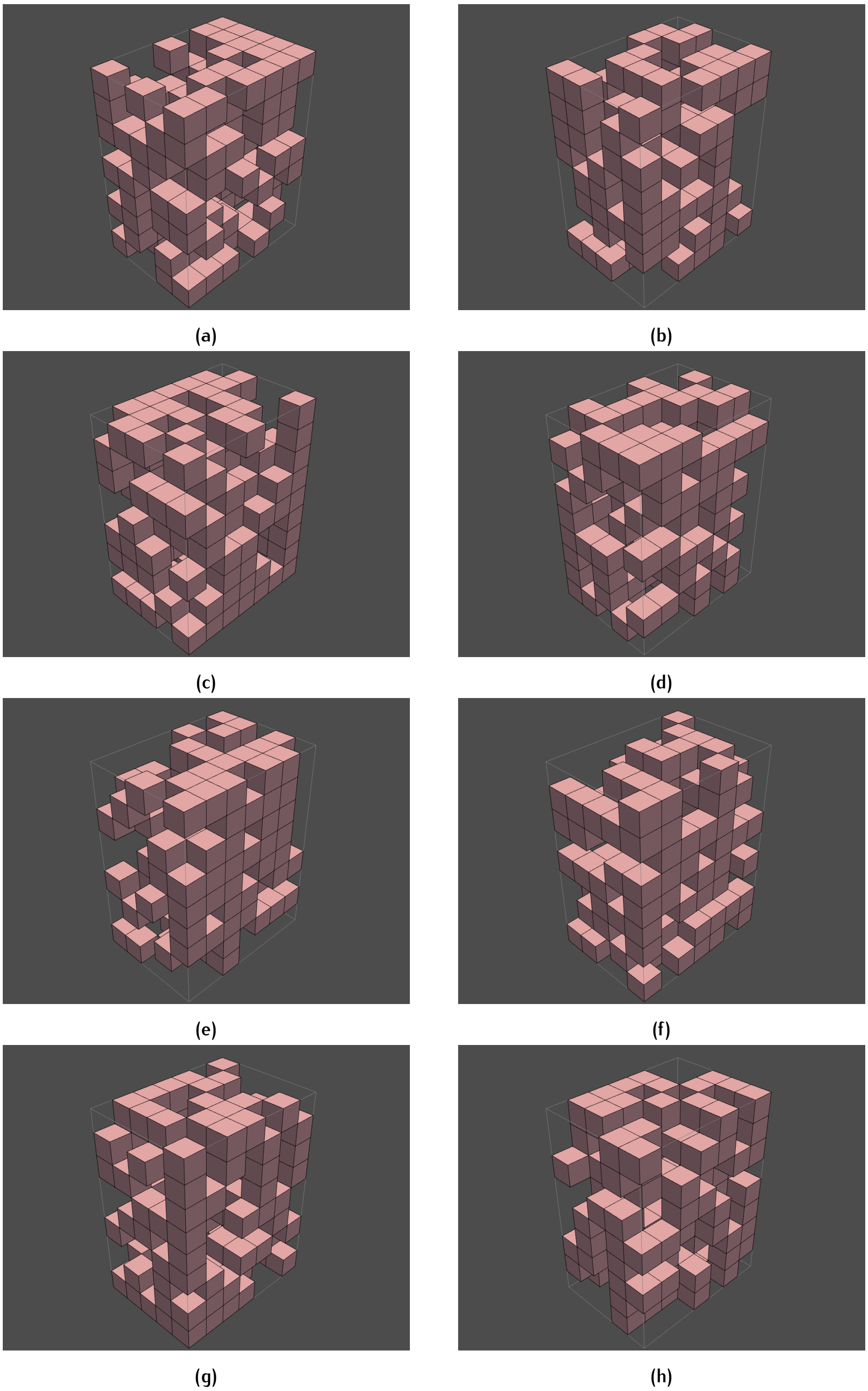


Figure 4.1: Test case optimization runs: IHS with 8 population and 150 generations. (a) is calculated in 45 seconds, (b) in 22 due to daylighting no longer being an objective. (c) scores 33491 for **c1**, 0.075 for **c3**, and 0.038 for **c5**. No visibility objectives whatsoever gives (d) which is not compact or has the correct FSI whatsoever. When we reintroduce the cost function (e), and add a weight to the cost function **c2** (f), we can see that the objective value is 0.065 for **c2**, 0.084 for **c3**, and 0.035 for **c5**. The same applied for PV scores (g) **c1** gives us scores of 0.07 for **c1**, 0.08 for **c3**, and **c5** is 0.035. Applying all objectives with weights gives a result of 0.073 for **c1**, 0.067 for **c2**, 0.081 for **c3**, and 0.036 for **c5**

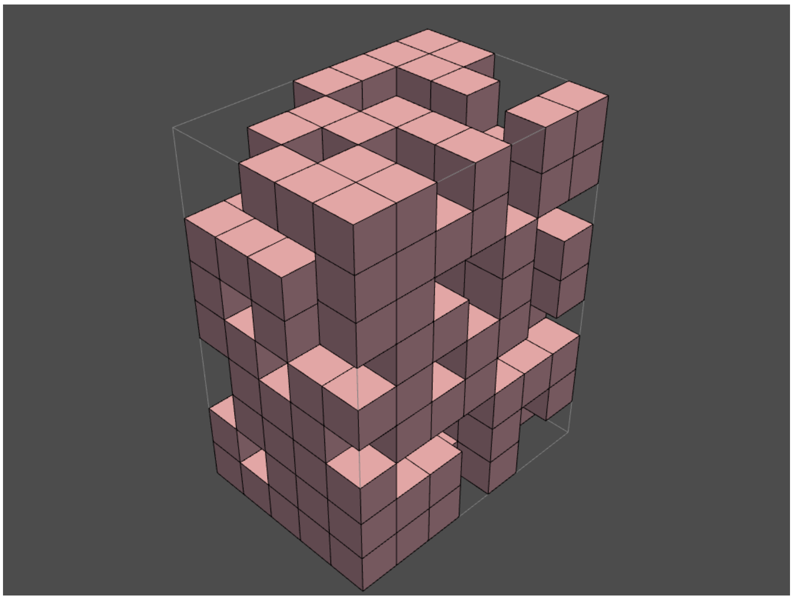


Figure 4.2: A massing generated using continuous decision variables instead of integer decision variables

pactness and the FSI of the building can no longer be guaranteed. When we ignore the visibility objectives altogether, we produce the following massing: 4.1d. This is surprising since we would expect the output to simply be a compact shape with the required FSI, a very straightforward result. The solver exits very quickly (0.1*s*) without achieving the desired objective scores. This is a worrying result for the validity of the method. When we reintroduce a visibility objective **c2** for the daylighting 4.1e, and we manually scale the objective value (i.e. we introduce a weighting of 1/10 for the daylighting score), the following result is achieved in 20.9 seconds: 4.1f. We can see that the objective scores here are more compensatory: 0.065 for **c2**, 0.084 for **c3**, and 0.035 for **c5**, suggesting that manual adaptation of the objective functions is desired. The configuration seems to be more contiguous, while daylight can reach deep inside the building. If we apply the same theory but try to optimize for PV score instead, the resulting score is 0.07 for **c1**, 0.08 for **c3**, and **c5** is 0.035, see 4.1g. Pursuing all objectives gives 4.1h as a result with a time of 45 seconds. The scores are 0.073 for **c1**, 0.067 for **c2**, 0.081 for **c3**, and 0.036 for **c5** Out of all configurations the last two might be the most promising, but many adaptations and tweaking had to be done to achieve the results.

Finding the correct weights and settings for generating results remains difficult and the validity of the results remains ambiguous. Several reasons for this could exist. First, the objective functions of the visibility objectives and those of FSI and compactness behave differently: While FSI and compactness always fall in a range between 0 and 1, the 'best' value for daylighting and PV is unknown at the beginning of the optimization, and only found during the process. This means that there needs to be some compensation of the total score towards the theoretically achievable maximum score. Simply summing all values is not a good enough solution to this, and manual weighting proves to be difficult. Second, the solver is set to an integer problem.

When we instead declare the variables as continuous, 4.2, we can still round the values of our variables to achieve results and continuous optimization seems to be more efficient but the accuracy of the final result may be an issue. Finally, the objective functions themselves might be problematic and need revision. When comparing the objective score of the configuration with the performance of a configuration after the fact in order to validate, daylighting and PV values do not seem

Amount of rays that can reach the sky		
	computation	validation
hits	33491	29057

Table 4.1: Total sky hits according to the optimizer (l) and total sky hits according to the ray tracing validation (r)

to match: 4.1 sky hits according to objective function of configuration 4.1c should be 33491 while the configuration actually achieves only 29057 hits after running the ray tracing again. A possible reason for this might be that the indexing for the input and output is not consistent, which is a matter that requires further research.

4.2 CONCLUSION

A method for finding optimal configurations in regards to multiple objectives has been described. In the previous section, we have attempted to apply this method to a larger scale in order to validate the results and find a generalized workflow for the application of the techniques studied. When comparing the objective scores from the output of the optimization and the simulated values after the fact, the validity of the proposed method is put in question because they do not match up. The reasons for this are as of yet not entirely clear and require attention. A more rigorous approach especially to the cost functions of the visibility objectives and their evaluation is required. The FSI and compactness objectives, being strictly speaking 'unitless' ratios perform more predictably than the visibility objectives, that correspond to a quantity of which we do not know what their maximum bound could be.

Besides the inconsistencies described above, the output of the model will always be dependent on the exact question asked. Taking a different approach in regards to the solvers, parameters, and objectives will always yield different results. Even if all factors remain consistent, the results might still be different since heuristics are used for solving the problem. The IHS algorithm seems to perform the fastest of all available integer solvers, although the reasons for this remain unclear.

At the same time, the framework itself is robust and allows quick simulation, computation, and adaptation towards different objectives which enables testing and exploration of design alternatives. Results are outputted at a reasonable time even in the improved resolution. The strength of the method lies in the fact that the decision variables directly correspond to the configuration mass, allowing a one-on-one translation of the output to the actual objective.

4.3 FURTHER RESEARCH RECOMMENDATIONS

The implementation results raise questions that need more attention in the final weeks of the thesis project, while other matters can already be identified that would be interesting for further research but fall outside of the scope of the thesis.

4.3.1 personal

The framework for a successful method is in place and overall it performs well and as expected. When given a more detailed inspection however, there seem to be some inconsistencies in the expected output and the actual results as mentioned in the previous section. Several reasons for this have been mentioned and the remainder of the thesis period will be spent on researching the following aspects:

1. Verify if the objective functions work as intended. To do this, it first needs to be validated if no errors have been made in the implementation of the objective functions.
2. If the problem persists: verify if the decision variables output and input indexing is consistent. Implementing a morton-ordering to ensure this consistency has been suggested for this in the past.

If this has been achieved and it can be verified that the output matches the expected values, several improvements can still be made to the model as well as some matters can still be explored:

1. Finding out why the IHS method performs best of the three methods studied.
2. Expanding the number of MCDA methods examined by:
    a) Branching out to another suite of solvers, such as SKCriteria.
    b) Increasing the number of methods studied from the PyGmo library. To do this, the variables need to be declared as continuous instead of integer.
3. Increasing the resolution at which we can feasibly run the simulation and optimization. Expanding the number of objectives by adding more cost functions.

Finally, to ensure the reproducibility of the method as well as encourage others attempting to apply MCDA techniques in the field of generative design, it is desirable to produce a sort of 'recipe' or infographic. This would describe exactly the steps taken and pitfalls to be evaded as explained in the thesis in an intuitive manner that can be used as a sort of cheat-sheet.

4.3.2 academical

As has been suggested by Ogrodnik and as mentioned earlier in the research, it is desirable to employ a combination of optimization strategies for achieving multiple objectives in generative and energy optimization. A database of all possible strategies with their respective strengths and weaknesses, as well as a unified library that combines the different options available from which to use these strategies would be a valuable addition to the field.

We now have the shape of the building as a decision variable and the performance of the building as an objective function. An interesting topic of further study would be if it could be theoretically possible to 'flip' the problem upside down. To take the values we want to achieve as a decision variable and evaluate the voxels that contribute the most to this objective.

The greatest challenge with the method as described above is the validation of the methods and the findings. A 1:1, real-world validation of an entire workflow would be highly valuable, albeit expensive and time-consuming.

5 REFLECTION

Due to the increasing pressure on the housing market, finding livable shelter for reasonable prices has started to become a major social, economical, political, and technical issue. As a future member of not only the housing market but also a citizen in the Netherlands and an employee in the field of architecture, this issue is important to me. Additions to the building supply need to be made but the challenge is to do so in a sustainable manner. To solve this, building planners will need to rely more and more on new techniques to build responsibly. The building industry however is notoriously inert when innovation is concerned and it is therefore essential to explore the potential of all aspects that can help with solving the crisis.

Within my studies at the faculty of architecture however, there seems to be a stereotype that you either are an architect, the artist who designs the buildings, or anyone else, who are concerned with the planning, technical details, financing, politics, simulations etc. of buildings and neighbourhoods. This black-white approach is detrimental to the field and to me, the ultimate architect can take the most integral perspective and look at all diverging aspects of a plan in a holistic way. This is where the field of computer science and mathematics can be a great boon, simply as a tool to support one's decision. However, there are not many courses that are offered that teach 'actual' generative design with regard to the many aspects that have to be considered when building, and beyond a few mandatory courses on 3D modelling, might go under the radar of a student.

Multi-criteria decision analysis has been widely applied to other industries, but in construction (engineering) less so. The large number of diverging actors and factors in construction design and engineering are fertile ground for research into MCDA methods. With an increasing adoption of digital methods from the industry, this topic within the field of Architecture and specifically Building technology is increasingly relevant with regular new research and development into space allocation, building massings, and energy systems. The benefits of this to the industry are promising but consistency, validation, and reproducibility remain issues. The goal of the research was therefore to develop such a methodology to learn about the benefits and difficulties of implementing such a method into an early design stage.

By education, I am not a mathematician or programmer. This has provided some difficulties while developing the method. Conventions, notations, best practices and even syntax, data structures, and general concepts that might seem trivial to others were unknown at the beginning of the research project. The research I have presented in the previous chapters is therefore first and foremost an exercise and demonstration, and not a proposal for the very best method to solve these kinds of problems. The lessons learned lie mostly in how I would approach a comparable problem if starting over again. I have learned about the general concepts involved in applying these methods as well as most importantly: the question you ask already holds the answer to the problem: the phrasing of the objectives (and therefore also the choosing of the objectives) is a critical aspect of finding a workable method. If the objective and variables relate to each other in a more straightforward way, modeling and therefore solving the problem becomes a much more straightforward matter. Having to do it all over again, I would therefore be much more mindful of what exactly it is I want to achieve by applying these methods.

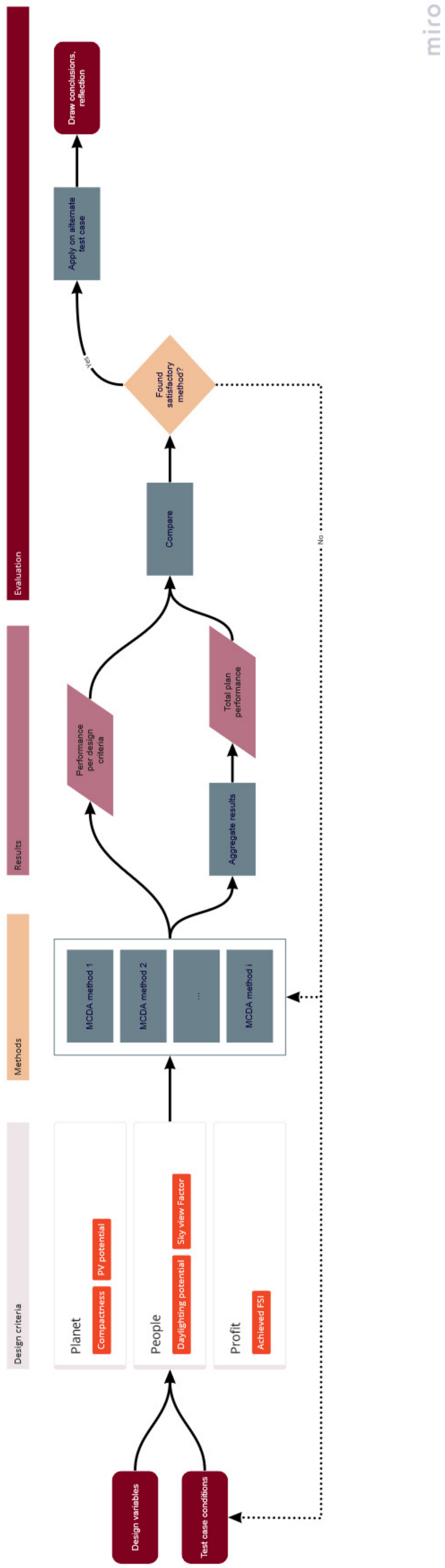
The research method as described was a valid approach to the problem in my opinion. Nevertheless, some issues were encountered and the graduation process was delayed. I did not at all times take full advantage of my mentors and was overly hesitant to ask for help or provide updates. The lighting especially is in my opinion an underdeveloped aspect of the research. The CoVid pandemic certainly did not help in this regard since a lot of the study was done at home, but in the future I should commit myself to more readily ask for help or support when I need it. Also, simply working in proximity to someone else has helped a great amount in regards to motivation. Another issue I ran into was that of having an overly ambitious scope at the beginning of the research process. Limiting the number of aspects to research was a good decision that could have been taken earlier in hindsight.

The results are as of yet not completely satisfactory. Finding out why there is a difference in expected and actual performance is very important to me in the weeks to come. Beyond this, I would be interested to 'take a look under the hood' and find out exactly what makes IHS perform significantly better than the other methods, and also find out if there is some merit to moving to a continuous formulation in stead of an integer formulation.

I am however content with the framework itself and all the new techniques and libraries I have learned to use. Beyond that, I think that it can be a valuable addition the the field by allowing planners and designers alike to make informed decisions and am excited to continue working on these kind of problems. When used correctly, and with the increasing performance of computers, new techniques, and new research, these methods can greatly improve the quality of the buildings we live in in the future.

6 APPENDIX

To be added: Pseudocode for the objective functions. For now, refer to My thesis repository where I will be uploading all relevant code, images, and the paper.


Design variables
Test case conditions
Design criteria
Planet
Compactness
PV potential
People
Daylighting potential
Sky view Factor
Profit
Achieved FSI
Methods
MCDA method 1
MCDA method 2
...
MCDA method i
Results
Performance per design criteria
Aggregate results
Total plan performance
Evaluation
Compare
Found satisfactory method?
Yes
No
Apply on alternate test case
Draw conclusions, reflection

BIBLIOGRAPHY

Agency, I. E. (2016). World energy statistics.

AIMMS (2016). AIMMS Modeling Guide - Integer Programming Tricks This file contains only one chapter of the book . For a free download of the complete book in pdf format , please visit www.aimms.com or order your hard-.

Ansari, N., Cajias, M., and Bienert, S. (2015). The Value Contribution of Sustainability Reporting-An Empirical Evidence for Real Estate Companies. *ACRN Oxford Journal of Finance and Risk Perspectives Special Issue of Social and Sustainable Finance*, 4(4):190–205.

Aries, M. B. C., Aarts, M. P. J., and van Hoof, J. (2015). Daylight and health: A review of the evidence and consequences for the built environment. *Lighting Research & Technology*, 47(1):6–27.

Azadi, S. and Nourian, P. (2020). shervinazadi/topoGenesis: Initial Pre-release.

Blum, C. and Roli, A. (2003). Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Computing Surveys*, 35(3):268–308.

Boelhouwer, P. (2020). The housing market in The Netherlands as a driver for social inequalities: proposals for reform. *International Journal of Housing Policy*, 20(3):447–456.

Buitrago, M., Adam, J. M., Alvarado, Y. A., Moragues, J. J., Gasch, I., and Calderón, P. A. (2016). Designing construction processes in buildings by heuristic optimization. *Engineering Structures*, 111:1–10.

Caldas, Luisa Gama (Maria Luisa da Oliveira Gama Caldas), . (2001). An evolution-based generative design system : using adaptation to shape architectural form.

Catalina, T., Virgone, J., and Iordache, V. (2011). Study on the impact of the building form on the energy consumption. In *Proceedings of building simulation 2011: 12th conference of international building performance simulation association, Sydney, 14-16 November*, pages 1726–1729.

Cinelli, M., Coles, S. R., and Kirwan, K. (2014). Analysis of the potentials of multi criteria decision analysis methods to conduct sustainability assessment. *Ecological Indicators*, 46:138–148.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Depecker, P., Menezo, C., Virgone, J., and Lepers, S. (2001). Design of buildings shape and energetic consumption. *Building and Environment*, 36(5):627–635.

Florou, A. (2021). *Generative solar-climatic configuration: A model for feed-forward optimization of building envelopes as to solar energy potential*. Student master thesis, TU Delft.

Heschong, L., Wright, R., and Okura, S. (2013). Daylighting Impacts on Human Performance in School. *Journal of the Illuminating Engineering Society*, 31:101–114.

Huang, I. B., Keisler, J., and Linkov, I. (2011). Multi-criteria decision analysis in environmental sciences: Ten years of applications and trends. *Science of The Total Environment*, 409(19):3578–3594.

Labayrade, R., Jensen, H. W., and Jensen, C. (2009). Validation of velux daylight visualizer 2 against cie 171:2006 test cases. *IBPSA 2009 - International Building Performance Simulation Association 2009*, 3:1506–1513.

Li, D., Liu, G., and Liao, S. (2015). Solar potential in urban residential buildings. *Solar Energy*, 111:225–235.

MacNaughton, P., Woo, M., Tinianov, B., Boubekri, M., and Satish, U. (2021). Economic implications of access to daylight and views in office buildings from improved productivity. *Journal of Applied Social Psychology*.

Mangialardo, A., Micelli, E., and Saccani, F. (2018). Does Sustainability Affect Real Estate Market Values? Empirical Evidence from the Office Buildings Market in Milan (Italy). *Sustainability 2019, Vol. 11, Page 12*, 11(1):12.

Murty, K. G. (1983). Linear programming. page 482.

Nagy, D., Villaggi, L., and Benjamin, D. (2017). Beyond heuristics: a novel design space model for generative space planning in architecture. In *ACADIA conference*.

Nijskens, R. and Lohuis, M. (2019). *The Housing Market in Major Dutch Cities*, pages 23–35. Springer International Publishing, Cham.

Ogrodnik, K. (2019). Multi-criteria analysis of design solutions in architecture and engineering: Review of applications and a case study. *Buildings*, 9(12).

Ollongren, K. H. (2019). Nadere informatie over energiezuinige nieuwbouw (BENG).

Ottensmann, J. R. (1977). Urban Sprawl, Land Values and the Density of Development. *Land Economics*, 53(4):389.

Paulson, B. C. (1976). Designing To Reduce Construction Costs.

Reinhart, C., , Environment, C. D. B., and undefined 2016. Urban building energy modeling–A review of a nascent field. *Elsevier*.

Reinhart, C. F. and Herkel, S. (2000). The simulation of annual daylight illuminance distributions — a state-of-the-art comparison of six RADIANCE-based methods. *Energy and Buildings*, 32(2):167–187.

Reinhart, C. F. and Walkenhorst, O. (2001). Validation of dynamic RADIANCE-based daylight simulations for a test office with external blinds. *Energy and Buildings*, 33(7):683–697.

Schüler, N., Cajot, S., Peter, M., Page, J., and Maréchal, F. (2018). The Optimum Is Not the Goal: Capturing the Decision Space for the Planning of New Neighborhoods. *Frontiers in Built Environment*, 3(February):1–22.

Shea, K., Sedgwick, A., and Antonuntto, G. (2006). Multicriteria Optimization of Paneled Building Envelopes Using Ant Colony Optimization. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4200 LNAI:627–636.

Slowinski, R., Greco, S., and Matarazzo, B. (2002). Axiomatization of utility, outranking and decision-rule preference models for multiple-criteria classification problems under partial inconsistency with the dominance principle. *Control and Cybernetics*, 31(4):1005–1035.

Tavares, S. and da Costa Silva, H. (2008). Daylight simulation in buildings. In *PLEA 2008 – 25th Conference on Passive and Low Energy Architecture,*, Dublin.

Turan, I., Chegut, A., Fink, D., and Reinhart, C. (2020). The value of daylight in office spaces. *Building and Environment*, 168:106503.

Warren-Myers, G. (2012). The value of sustainability in real estate: A review from a valuation perspective. *Journal of Property Investment and Finance*, 30(2):115–144.

Zhang, A., Bokel, R., van den Dobbelsteen, A., Sun, Y., Huang, Q., and Zhang, Q. (2017). Optimization of thermal and daylight performance of school buildings based on a multi-objective genetic algorithm in the cold climate of China. *Energy and Buildings*, 139:371–384.

COLOPHON

This document was typeset using LaTeX. The document layout was generated using the arsclassica package by Lorenzo Pantieri, which is an adaption of the original classicthesis package from André Miede.

TU Delft