Performance Management for Parametric Building Design Generated by a Genetic Algorithm

Brikene Berisha[1], Andrei Lobov[2], Bjorn Andersen[3]

[1]Norwegian University of Science and Technology, brikene.berisha@ntnu.no
[2]Norwegian University of Science and Technology, andrei.lobov@ntnu.no
[3]Norwegian University of Science and Technology, bjorn.andersen@ntnu.no

Corresponding author: Brikene Berisha, brikene.berisha@ntnu.no

Abstract. The paper analyzes the results of applying a genetic algorithm to generate building designs. The genetic algorithm developed by the authors uses shadow length as fitness function value to examine the impact on the building design. The input of the genetic algorithm for the creation of building designs is the volume of the building, the number of building blocks, as well as the area in which the building is constructed within. The model proposed in the paper serves to illustrate to the reader how parameters can be added, and the model can be extended to apply it to different use cases. Four parameters are chosen for the analysis: shadow, transportation, outdoor area, and view. The parameters are demonstrated in simplified mathematical equations and are introduced with the necessary basic information to understand the concept of formulating the equations. The results show that the model is useful for engineers in their decision-making process and can be extended by adding other parameters. From the perspective of building design, the study emphasizes the adaptability and changeable features of the model for different case studies, which can prove useful in terms of feedback.

Keywords: building design, parameters, genetic algorithm, shadow analysis
DOI: https://doi.org/10.14733/cadaps.2024.350-373

1 INTRODUCTION

1.1 Motivation

The performance of a product, process, design, system, or organization can be broken down into several parameters to provide feedback on the product. The use of parameters to describe a procedure is called parametric design. By changing the parameter variables, a range of alternative solutions can be created. A solution can be determined based on the purpose of the use of parametric design and the given criteria to support the decision-making process for measuring the performance of the building design ([14, 3]).

The authors suggest a model that uses the shadow cast by a building as a parameter to measure the performance of the building design. Shadow analysis refers to the study of how an object's shadow affects the surrounding environment based on the position of the sun. Such analyses can be used to determine whether buildings receive enough sunlight during the day. Furthermore, tall buildings can have a negative impact on their immediate surroundings. According to [23] various studies have shown that dark spaces resulting from shadows cast by tall buildings cause workers to suffer depression, increase incidences of "winter diseases," congestion in streets, and the possibility of fires. People tend not to gather for social activities in dark alleys or shaded streets. Instead, they are attracted to spend time in sunny places ([9]). Nevertheless, tall buildings are unavoidable due to population increases and land shortages. Therefore, to develop a healthy living environment for the population and to ensure urban functionality, urban planners should seek solutions based on the findings from shadow analysis ([23]).

1.2 Literature Review

Much effort has been invested to use genetic algorithms in the creation of building designs. In architecture, the use of genetic algorithms has become a necessity, due to the increasing complexity of information and human needs. Constraints inhibit designers' ability to comprehend such information and needs, and to anticipate solutions ([12]). To date, many research studies have focused on the energy efficiency or energy consumption of building designs, and they have emphasized the construction of smart buildings and maintenance of a healthy environment for buildings' users ([11]). [19] developed an interactive genetic algorithm (IGA) whereby the user can interact with the process and the population of the genetic algorithm. The IGA can evaluate and classify potential solutions through a real-time comparison of a building's energy rating, shading impact, and compact shape. A similar approach was used by [26] who developed a method called ParaGen. The method uses a genetic algorithm to find geometric solutions to architectural engineering problems and involves interactive participation by the designer.

More recent research has focused on building design parameters and the workflow of parametric design integrating genetic algorithms and energy simulations ([13]). [25] developed a simulation-optimization tool that connects a genetic algorithm to a building energy simulation engine to select optimal values for parameters such as shading. A similar approach was used by [24] in which the objective function integrated energy and lighting performance. The authors outline an automated workflow process using a genetic algorithm, including geometric parameters. Similarly, the analysis performed by [28] revealed that a concrete static shading system using a genetic algorithm involving geometrical parameters can be used to ensure good performance in terms of daylight saving and energy saving. [27] used a genetic algorithm to provide the optimal building form to achieve the highest natural ventilation potential in urban wind environments. Furthermore, genetic algorithms can be used to optimize building refurbishment scenarios, dealing with energy consumption, economic cost, comfort, and environmental impact ([22]).

[29] applied a genetic algorithm and simulated annealing to optimize the layout of building floors. The building floors were created automatically for use in building information modelling (BIM). [18] explored the use of BIM and genetic algorithm to support decision-making and optimization for sustainable building envelope design. They developed a BIM-GA (BIM-genetic algorithm) optimization method in which the objective functions were the envelope's overall thermal transfer value (OTTV) and construction cost.

Most of the above-mentioned studies have focused on energy consumption and energy efficiency, in which a genetic algorithm was combined with other machine learning tools or developed in such a way that engineers and architects could interact and choose the population for the outcome. Very few of their models incorporate parameters, and the authors do not show how parameters could be included in the models. Additionally, the proposed models and systems are not automated or are only partly automated. The processes are complex and influence computational time. Consequently, they slow down the automation process and discourage the running of more iterations to find optimal solutions. Pseudocodes are not presented in the cited papers in a form that would enable them to be used in separate studies.

2 OBJECTIVES

This paper presents a pseudocode of a developed genetic algorithm for generating building designs. The authors advocate the parameters of shadow, transportation, outdoor area, and view, which can be included in the proposed model for measuring the performance of generated building designs. Performance measurement is generally defined as quantification of the efficiency and effectiveness of an action ([2]). However, for both the building construction domain and the research presented in this paper, performance measurement is defined as the evaluation of different parameters that describe a building. The parameters considered in this paper are shadow transportation, outdoor area, and view. A parametric model is proposed in connection with the suggested formulas to measure the performance of a building design. According to [15], the parameters in such a model define and clarify the relationship between the model and the design of the building.
The post-occupancy building design architecture adapted to the use case presented in this paper is shown in Figure 1. The proposed model is integrated into the model as a "Parametric Design." Figure 1 also shows the connection between the customer's experience, the "Experience Performance," the process of designing the building, the "Building Design Performance."

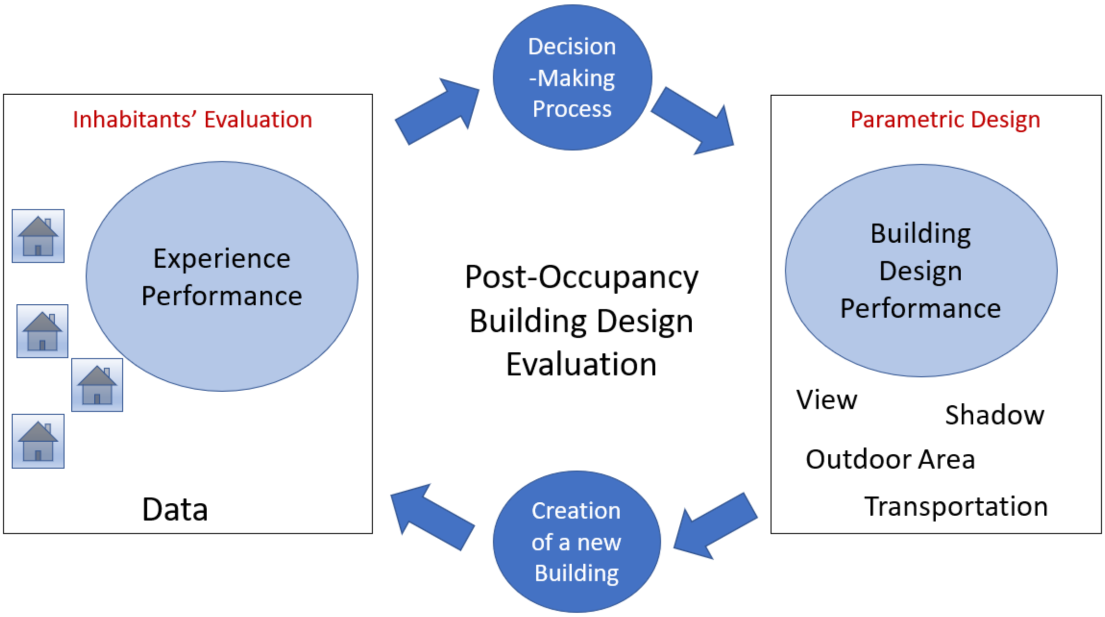

Figure 1: Illustration of the post-occupancy building design evaluation.

Data are collected in the form of feedback from customers regarding their experience of living in a building. These data then form the foundation of the decision-making process in subsequent building designs. Engineers and architects choose parameters for building designs according to the results of the analyzed data. For the design process, they can use artificial intelligence (AI) tools to automate the procedure.
Instead of using modeling tools such as Rhinoceros and Grasshopper, as has been done for most previous studies, Siemens NX was used to illustrate the results presented in this paper. The proposed model explains how parameters can be included to extend the model, and readers are provided with instructions on how to

apply it in separate case studies. The approach is documented with simplified mathematical equations.
The problem definition and the parameter indicators are described in section 3. The results (presented in section 4) serve to illustrate the application of a genetic algorithm, where the fitness function represents the shadows cast by generated building designs. The results are analyzed in the discussion and conclusion section (section 5).

3 METHODOLOGY

This section introduces the chosen parameters of the proposed model, which are: shadow, transportation, outdoor area, and view. The simplified equations of the parameters return a value that measures the performance of the building design to support the decision-making process. Additionally, the pseudocode used for the generation of building designs is explained. The approach in this chapter is the mathematical illustration of the chosen parameters and the explanation of how the genetic algorithm is configured.

3.1 Problem Definition

When engaged in the design of structures, architects encounter the obstacle of utilizing various software tools to acquire assessments on specific parameters. This utilization of diverse software not only consumes time but is also influenced by the preferences and aspirations of the client. As a result, the objective of the research is to create a tool that can generate architectural designs based on the outcomes of shadow analysis. Additionally, this tool should have the capacity to incorporate additional parameters to assess performance.

3.2 Parameter Indicators

For the model, a general formula is suggested, which is defined as:

$$Parameter_i = \sum_i \omega_i * F_i, \tag{1}$$

where F_i is the factor calculated by the suggested mathematical equations. The weight ω_i is essential for each factor in the sum and needs to fulfill the requirement of $\omega_i \in [0,1]$. The weights sum up to 1. The equation above is applicable if several criteria are used to describe the parameter. In this paper, some equations cannot be summarized within this structure and therefore must be analyzed separately. The weights $\omega_i \in [0,1]$ are chosen by the customer. In the following, we illustrate how four different aspects of building design can be modeled, but the same principles can be extended to any other dimensions.

3.2.1 Shadow

The shadow cast by a building impacts the daily life of the building's inhabitants. For example, the office building should be exposed to sufficient daylight, whereas in the evening, after most employees have left the office, it is unnecessary for the building to receive more light than during the day. In the case of residential buildings, evening sunlight mainly impacts the well-being of inhabitants who have returned from work or school. Especially rooms in which they spend the most time, such as the living room, should face the sun.
The computation of the shadow is approached by calculating the height of the building in relation to the length of the shadow (Figure 2), as described by [8]. For the understanding of the developed genetic algorithm, the subsequent calculations illustrate the approach of the authors.

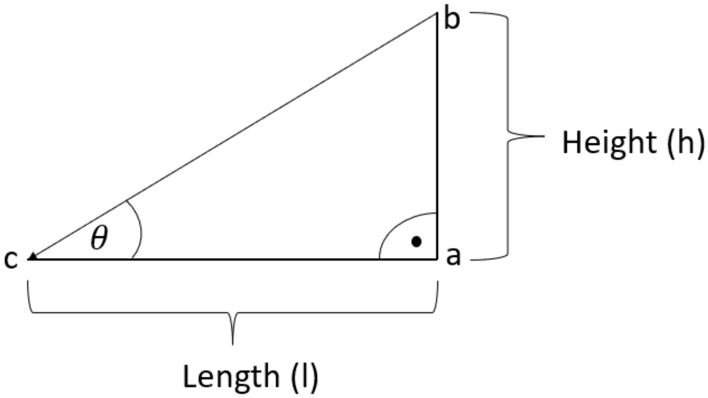

Figure 2: Relation between the height (h) and the shadow length (l) of a building.

In the following, a simple case is presented in which the ground slope equals 0. The parameters are as follows: the angle of the elevation of the sun, θ; the height of the building, h, which represents the distance from the point a to the point b; and the shadow length of the building, l, which describes the distance from point a to point c. The height is expressed as:

$$h = ab = \tan(\theta) * ac = \tan(\theta) * l, \tag{2}$$

where

$$\tan(\theta) = \frac{ab}{ac}. \tag{3}$$

It is also possible to calculate the shadow according to the height of the building. For this purpose, both the azimuth of the sun (abbreviated as "azm"), as well as the altitude (defined as "alt") are needed for the calculations. Both angles are illustrated in Figure 3.

The altitude angle is the angle measured between the horizontal plane and the sun's position in the sky, which changes throughout the day according to the season and time. The altitude angle varies according to different places on the earth. The azimuth angle is the horizontal angle between a celestial body and the north. In Figure 3, the length of the shadow on the ground is $|OP|$, where the point P represents the shadow point Q on the ground. To calculate the shadow length $|OP|$, the mathematical geometry of the triangle OPQ is used. Therefore,

$$\tan(\text{alt}) = \frac{h}{|OP|} \leftrightarrow |OP| = \frac{h}{\tan(\text{alt})}. \tag{4}$$

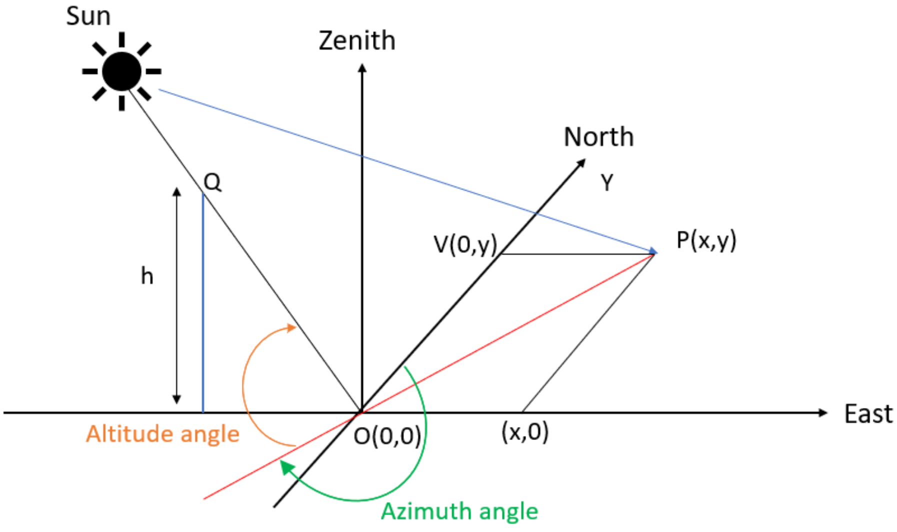

Figure 3: The construction for the calculation of the shadow.

The goal is to calculate the tip of the shadow, meaning the coordinates of the shadow located at P(x,y) (i.e., x units East and y units North) Using the triangle ORP,

$$\sin(\text{azm} - 180^\circ) = \frac{x}{|OP|} \leftrightarrow x = |OP| * \sin(\text{azm} - 180^\circ). \tag{5}$$

The length $|OP|$ of equation (4) is substituted into equation (5), such that,

$$x = \frac{h}{\tan(\text{alt})} * \sin(\text{azm} - 180^\circ). \tag{6}$$

As a next step, the expression $\sin(\text{azm} - 180^\circ)$ in equation (6) is simplified:

$$\begin{aligned} \sin(\text{azm} - 180^\circ) &= \sin(\text{azm}) * \cos(180^\circ) - \cos(\text{azm}) * \sin(180^\circ) \\ &= -\sin(\text{azm}) - \cos(\text{azm}) * 0 \\ &= -\sin(\text{azm}). \end{aligned} \tag{7}$$

Therefore, the final coordinate x-coordinate for the shadow is:

$$x = -\frac{h * \sin(\text{azm})}{\tan(\text{alt})}. \tag{8}$$

The same way, the coordinate y-coordinate is calculated for the shadow in the same way. Starting with the equation:

$$cos(\text{azm} - 180^\circ) = \frac{y}{|OP|} \leftrightarrow y = |OP| * \cos(\text{azm} - 180^\circ). \tag{9}$$

Replacing the shadow length $|OP|$ of equation (4) in equation (9) gives:

$$y = \frac{h}{\tan(alt)} * \cos(\text{azm - } 180^\circ). \tag{10}$$

The expression $\cos(\text{azm - }180°)$ is simplified as:

$$\begin{aligned}\cos(\text{azm} - 180°) &= \cos(\text{azm}) * \cos(180°) + \sin(\text{azm}) * \sin(180°) \\ &= -\cos(\text{azm}) + 0 \\ &= -\cos(\text{azm}),\end{aligned} \tag{11}$$

and replaced it in equation (11) such that the y-coordinate is expressed as:

$$y = -\frac{h * \cos(\text{azm})}{\tan(\text{alt})}. \tag{12}$$

Therefore, the coordinates of the shadow point $P(x, y)$, illustrated in equation (8) and (12), are summarized as:

$$P = (-\frac{h * \sin(\text{azm})}{\tan(\text{alt})}, -\frac{h * \cos(\text{azm})}{\tan(\text{alt})}). \tag{13}$$

Information about the date, time, year, and location is needed to calculate a shadow point. The shadow of the whole building is computed by calculating the shadow point of each corner of the building. First, the latitude to be used in the calculation needs to be known. The latitude is between +90 degrees north of the equator and -90 degrees south of the equator. The solar altitude angle is the angle between the line to the sun and a horizontal plane and is the complement to the zenith angle (i.e., the angle between the line to the sun and the vertical direction). Thus, the zenith angle refers to the angle of incidence of beam radiation when it strikes a horizontal surface. The solar azimuth angle is the angular displacement from the south of the projection of beam radiation on the horizontal plane. Displacements east of south are negative, and west of south are positive ([16]). Figure 3 shows an object located at $(0, 0)$. For the calculation of an object that is not in its place of origin, another factor needs to be added to the equation (14), to generalize it. According to [6] and [7], the object's original coordinates are added to the coordinates of the new place where the object is positioned, which is defined as $P_x^{'}$ for the x-coordinate, and $P_y^{'}$ for the y-coordinate. Therefore, the result of the shadow point is:

$$P = (P_x^{'} - \frac{h * \sin(\text{azm})}{\tan(\text{alt})}, P_y^{'} - \frac{h * \cos(\text{azm})}{\tan(\text{alt})}). \tag{14}$$

For the three-dimensional use case, the z-coordinate is added. The value of the z-coordinate is 0, as the shadow is projected onto the ground ([5]).

3.2.2 Transportation

Transportation is defined as the mode of transport to which inhabitants have access close to their building. Different modes of transport are bus, train, or plane, as well as walking and cycling along designated routes. The pedestrian routes must be safe and designed for expected traffic and modes of transport. In Norway, requirements relating to the design of safe pedestrian routes exist [1], but are not the focus of this paper.
The authors measured transportation with reference to existing public transportation service around a hypothetical case building. Additionally, the building design included parking lots for inhabitants who prefer to use their cars, such as families and people who do not use public transport. The week from Monday to Sunday was considered a suitable period for measuring the public transportation around the designed building. Nonetheless, the plan for public transportation varied from Monday to Friday compared with on Saturday and Sunday. In Norway, public transport services change during holidays and therefore holidays were not considered in the

[1] Ministry of Labour and Social Inclusion: Regulations concerning the performance of work, use of work equipment and related technical requirements.2011. https://lovdata.no/dokument/SFE/forskrift/2011-12-06-1357.

calculations. The authors approached the formulation of the equation of transportation with the collected information and assumptions made. It is a simplified mathematical equation which can be adjusted to become more complex.
In the model, the number of buses around the building, and the number of parking lots available for the inhabitants are expressed in two separate equations as follows:

$$\text{Number of Buses Parameter } = \frac{\text{Number of Buses per week}}{\text{Number of Inhabitants}}, \tag{15}$$

$$\text{Number of Cars Parameter } = \frac{\text{Number of Parking Lots in the Building Complex}}{\text{Number of Inhabitants}}. \tag{16}$$

The higher the result of the equation, the better the transportation around the building.
The number of buses available, and the number of cars parked in the parking lots within the building complex must cover the minimum number of inhabitants of the building. However, more people than the inhabitants use the bus stop in the building complex. According to the formulated equations above, the result is:

$$\text{Transportation Parameter } = \sum_i \beta_i F_i^{Transportation}, \tag{17}$$

where $F_1^{Transportation}$ is the "Number of Buses Parameter," β_1 is the matching weight, and $F_2^{Transportation}$ represents the "Number of Cars Parameter" with the chosen weight β_2. The equation is extendable by integrating the exact number of car seats in each car parked in the building complex according to the number of people in each household using the car(s) of the household. The same applies to the capacity of the bus, the number of seats, and the number of inhabitants who travel either daily or most often by bus. The higher the "Transportation Parameter," the better the transportation around the building. The transportation parameter can be extended by including access to, for example, trains, trams, planes, and taxis.

3.2.3 Outdoor Area

An outdoor area will enable a building's inhabitants to interact with each other and practice different activities. Such outdoor areas will have different functions, depending upon the type of building. For example, the primary function of the outdoor area of an office building is likely to be a parking area, whereas for a residential building, an outdoor amenity area will be suitable for recreation, play, and activities for people of different ages. In modern building constructions in Norway, parking areas are incorporated into buildings, so that cars can be parked in the basement, where they are protected from extreme weather conditions. The outdoor area can serve as a play-ground for children, include green areas, and possibly a community garden for young and old, space to walk the dog, social activities, sitting on a bench, or even offer a space to eat and have a barbecue in the open air. Lighting is necessary around and within the building complex [2] ([1]). The outdoor area includes balconies, terraces, and other detachable areas where the inhabitants are exposed to an open sky. In public buildings such as universities and schools, the outdoor area is public, which means that it can be used by different social groups ([4]). In the following, the authors formulate simplified mathematical equations to illustrate how the information of the outdoor area can be included in the model.
For the outdoor area to be an integral part of the building, the assumption holds that the total area is equal to the building area plus the outdoor area, such that: is equal to the building area plus the outdoor area, such that it results into:

$$\text{Total Area Parameter} = \text{Building Area } + \text{ Outdoor Area}. \tag{18}$$

[2] Ministry of Labour and Social Inclusion: Regulations concerning the performance of work, use of work equipment and related technical requirements.2011. https://lovdata.no/dokument/SFE/forskrift/2011-12-06-1357.

The outdoor area is chosen according to a percentage of the building area:

$$\text{Outdoor Area } = \alpha * \text{ Building Area}, \tag{19}$$

where α represents the percentage chosen by the reader. To link the calculation of the area to the governments or planners' regulations governing the use of outdoor area, equations can be added to the model. Regulations relating to the outdoor area include, for example, light and sun conditions, noise, environmental impacts, play areas shielded from traffic, room for wheelchair users, and where seating is constructed.

3.2.4 View

A building should have sufficiently large windows for the inhabitants. The size of window and view from it will depend on the building in question [3] ([1]). Office buildings tend to have views of other buildings within the city center or industrial area in which they are located. By contrast, the views from schools in Norway tend to be of nature, playgrounds for children, soccer fields, and other spaces for exercise, sports, and activity. A wall in front of a building is not normally considered a nice view. Additionally, if an object in front of the building blocks sunlight, the quality of the outside view will be considered low, as light will not reach the interior spaces of the building. Therefore, buildings should be located a certain distance from each other. Windows afford a sense of connection with the outside world for a building's occupants ([10]). The authors approach the parameter "view" as the outside view which can be measured by the design of the windows, and, in the case of residential buildings, by the ratio of the number of inhabitants to the number of windows. The glass area of the window is supposed to be one-tenth or one-eighth of the floor area, and the former is the most common reference point for the size of the window ([10]):

$$\text{Window Area Parameter} = 1/10 * \text{ Floor Area of the Building} \tag{20}$$

The window area of the building refers to the floors of the windowed rooms. As an additional parameter, the window width is considered. In the model, this is chosen as 1/10 of the sum of the width of each wall:

$$\text{Window Width Parameter } = 1/10 * \sum \text{ (Width of Walls of the Building)} \tag{21}$$

The higher the number of windows around the building, the more opportunities the inhabitants will have to view the outside area. Therefore, the number of windows is considered an additional parameter:

$$\text{Number of Windows Parameter} = \frac{\text{Number of Windows}}{\text{Number of Windows for each Inhabitant}} \tag{22}$$

The equations above can be summarized and expressed as "View Parameter":

$$\text{View Parameter} = \sum_i \delta_i F_i^{View}, \tag{23}$$

where δ_i represents the weight of each matching factor, such that

$$F_1^{View} = \text{"Window Area Parameter"},$$
$$F_2^{View} = \text{"Window Width Parameter"},$$
$$F_3^{View} = \text{"Number of Windows Parameter"}.$$

The parameter of the outside view can be extended by additional indicators, such as the energy level. The energy level of a building is affected by the size of the windows, the height of the windows relative to eye level and the outside landscape, the amount of sunlight that enters the rooms through the windows, the thickness of the windows, the material used in the windows' construction, and further details, all of which are beyond the scope in this paper. For further research relating to the outside view and a case study, see [17].

[3] Ministry of Labour and Social Inclusion: Regulations concerning the performance of work, use of work equipment and related technical requirements.2011. https://lovdata.no/dokument/SFE/forskrift/2011-12-06-1357.

3.3 Implementation of a Hypothetical Use Case

The use case is implemented in Python, and the results are illustrated in Siemens NX. For the application of Siemens NX, a knowledge fusion language is used. Knowledge fusion is an advanced knowledge-based engineering (KBE) tool to develop an object-oriented design. The objects of the model are cubicles used to construct a building, which are called "building blocks." Each building block has a length, width, and height of 100 units in Siemens NX, which was used because of its simplicity in simulating the use case and calculations. To assemble the building blocks, a genetic algorithm is used.

In the following analysis the parameters are not going to be analyzed in further detail due to being out of the scope of the paper. Global optimization algorithms can be considered to calculate the optimal values for the parameters which is a choice which depends case-by-case.

The first step is to choose the location of the analysis. The authors decide to place the imaginary building in Trondheim, Norway, where the latitude is $63.4°N$ and the longitude is $10.3°E$. Additionally, the azimuth and altitude of the sun are needed to calculate the shadow. [5] provides an algorithm to calculate the sun's position from sunrise to sunset at a specific latitude, for a given date, and a given interval of time in minutes. Another way of acquiring the altitude and azimuth of the sun is to use the Python package Pysolar for the calculation. Polar is a collection of Python libraries for simulating the sun's irradiation at any point on the earth. The Pysolar package stores information on altitude and azimuth for the period 2007 to 2020, and therefore, the year 2020 is used for the calculations. In the following, the authors explain the different parts of the algorithm, which together form the genetic algorithm based on the purpose to design the fittest buildings according to the defined fitness function. The genetic algorithm is divided into three main parts: initializing the building, selection, and crossover. All those three parts combined form the genetic algorithm which is presented as the last algorithm in this chapter. The input data of the algorithms are called parameters, whereas the outcome of the algorithm is called output values.

The reviewer suggests adding the exterior wall area as an algorithm output. However, as detailed in the paper, the authors are exclusively focusing on the shadow length within the algorithm. Thus, the authors would like to reiterate that no additional parameters are under consideration for inclusion in the algorithm. As previously stated, the paper's objective is to demonstrate the viability of the approach, with the shadow length as the chosen parameter. While further variables are not incorporated at present, they are intended for future research considerations. The authors appreciate the concern raised by the reviewer and the depth of understanding that will benefit future research.

3.3.1 Initializing

The first step in applying a genetic algorithm is to initialize the population, as shown in Algorithm 1. Lines 1-4 in Algorithm 1 define the parameters for designing the building: the volume of the building block, the number of blocks that can fit in the area, and the number of blocks that fill out the volume. Starting in line 5, the number of floors is a random number within the range 0 and the number of blocks for the volume. The number of blocks for the volume is divided by four, so the first floor of the building has at least four blocks. Depending on the building design, the x-coordinate and y-coordinate, or length and width respectively, are calculated using a similar logic. In lines 10-17, the values i, j, and k define each building block's coordinates by filling up the given volume and the set area. The finished building comprises a list of building blocks, each with their given coordinates. The building generated by the algorithm is placed at the coordinates $[0, 0]$. If several building designs are generated, coordinates are chosen so that the buildings do not intersect with each other.

3.3.2 Fitness Function

The fitness function evaluates how "fit" a parent, or the offspring is. The fitness function is defined as the length of the shadow, which is calculated as:

$$l = \frac{h}{\tan(\alpha)'} \tag{24}$$

where α is the altitude angle, h is the height of the building, and l is the length of the shadow. The shadow is supposed to affect its surroundings as little as possible, so that the inhabitants can receive sunlight. Therefore, the fittest building design is the one with minimal shadow. To observe the shadow as a fitness function in the simulations, the height of the building is calculated. The height of the building is the building block with the maximum z-coordinate. Given the maximum height of the building, the altitude is calculated with the Python package Pysolar, and the values are inserted into equation (24) ([5]). The altitude changes throughout the day and therefore needs to be updated.

Algorithm 1 Initializing of Building

```
    Input: Volume, Area
    Output: Finished Building
Finished Building = [ ]
Volume of Building Block = width * length * height
Number of Blocks in the Area = Area / (width * length)
Number of Blocks in the Volume = Volume / Volume of Building Block
procedure SETTING DIMENSIONS
    Number of Floors = Get a random integer in range (0,Number of Blocks for Volume / 4)
    width = Get a random integer in range(0,Number of Blocks for Area)
    length = Get a random integer in range(0,Number of Blocks for Area)
    flag = 0
    for k in range(0, Number of Floors) do
        for i in range (0, width) do
            for j in range(0, length) do
                Building Block = [ i * 100, j * 100, k * 100]
                flag += 1
                Append each Building Block to the Finished Building
                if flag == Number of Blocks for Volume then
                    return Finished Building
                end if
            end for
        end for
    end for
end procedure
```

3.3.3 Selection

The selection of the building design is made by using Algorithm 2. The input is the current generation of building designs and their matching fitness results. The algorithm's output is the array of pairs of selected parents that will intersect to create offspring. The algorithm is divided into three sections, each of which follows the same logic. The fittest building design with the minimum shadow is selected in lines 1-6. Once the building design is detected, it is removed from the list. The procedure is repeated in lines 9-13 to find the second-fittest building design. In lines 16-19, the third fittest building design is selected. The function "min()" in lines 2, 9, and 16 returns the item with the lowest value in the list of fitness results. The function "index()" applied in lines 3, 10, and 16 returns the index of a located element.

Algorithm 2 Selection

```
    Input: Current Generation, Fitness Results
    Output: [[Selected Parent 1, Selected Parent 2], [Selected Parent 1, Selected Parent 3]]
 1: procedure FIND THE FITTEST
 2:     Fittest = min(Fitness Results)
 3:     index = Index(Fittest)
 4:     Selected Parent 1 = Current Generation(index)
 5:     Remove Selected Parent 1 from Current Generation
 6:     Remove Fittest from Fitness Results
 7: end procedure
 8: procedure FIND THE SECOND FITTEST
 9:     Fittest = min(Fitness Results)
10:     index = Index(Fittest)
11:     Selected Parent 2 = Current Generation(index)
12:     Remove Selected Parent 2 from Current Generation
13:     Remove Fittest from Fitness Results
14: end procedure
15: procedure FIND THE THIRD FITTEST
16:     Fittest = min(Fitness Results)
17:     index = Index(Fittest)
18:     Selected Parent 3 = Current Generation(index)
19:     Remove Fittest from Fitness Results
20: end procedure
21: return [[Selected Parent 1, Selected Parent 2], [Selected Parent 1, Selected Parent 3]]
```

3.3.4 Crossover

The result of Algorithm 2 is used as input for Algorithm 3. Parents 1 and 2 are crossed to receive the offspring, the child, which is the output of Algorithm 3. As a requirement, both parents must have the same length and amount of building blocks. First, a list is created, where the elements of each parent are appended, as shown in lines 2-8. To ensure that no duplicates exist, a new list called "res" is created to remove them. The algorithm sorts, from lines 17-19, the elements in the list "res." The sorting starts from lowest and ends with highest. In Python, the list is sorted with the argument "sorted ()." The argument returns a list of sorted elements, where the "key" element is invoked. The "key" element refers to the single argument function to customize the sort order, defining to which element the sorting is applied.

The sorting is applied to every second element, representing the y-coordinate of each building block. Thus, the building blocks are connected, and they do not get separated from the overall building construction. The procedure is repeated by ordering the list by the third element. The third element represents the z-coordinate, and thus the height of each building block. The last step is to cut the list "res" by the length of the parent. The non-standardized approach ensures that no building blocks are disconnected from the rest of the building design. The ordering can be changed and adapted to the reader's visions or goals, as discussed in the following sections in this paper. The purpose of the crossover is to merge two different building designs into one, taking different building blocks from both parents into account.

Algorithm 3 Crossover

```
    Input: Parent 1, Parent 2
    Output: Child
procedure CREATE LIST
    List = [ ]
    for Elements in Parent 1 do
        Append each Element in the List
    end for
    for Elements in Parent 2 do
        Append each Element in the List
    end for
    Result = [ ]
    for i in List do
        if i not in Result then
            Append the i-th Element in Result
        end if
    end for
end procedure
procedure SORTING
    Result sorted by second element
    Result sorted by third element
    Child = Result sorted by third from Element 0 to the length of the Parent
    return Child
end procedure
```

3.3.5 Genetic Algorithm

Algorithm 4 is a genetic algorithm that assembles Algorithms 1, 2 and 3. The input is the volume, area, and number of buildings chosen to be initialized. The output is the children and their fitness results. The list of data is defined, and the list of the children is generated. In lines 3-5, the number of buildings is initialized to create a data-base.

As a next step, the fitness results of the data are calculated by using the fitness function. A current generation is created by using Algorithm 2 for the selection of the data and their matching fitness results. Each building of the current generation is crossed with another building to create offspring, removing duplicates. In line 9, the fitness results of the children are calculated.

Algorithm 4 Genetic Algorithm

Input: Volume, Area, Number of Buildings
Output: Children, Fitness Results of Children

```
Data = [ ]
Children = [ ]
for i in range(0,Number of Buildings) do
    Initialize Buildings with chosen Volume and Area and append to Data
end for
Calculate Fitness Results of Data with the fitness function
Current Generation = Selection of Data and the matching Fitness Results
Children = Crossover of Buildings of Current Generation
Fitness Results of Children = Calculate the Fitness Results of Children
return Children, Fitness Results of Children
```

4 RESULTS

For the analysis, the attributes date, volume, and area are changed, and the outcome of the building design is explained. The dimension of shadow is used illustrate the results. Additionally, the operator crossover is inspected, and the fitness function is studied. The results of the fitness function are rounded to two places after the decimal point. The key results of the analysis are summarized in a table as an overview for the reader at the end of the chapter.

4.1 Illustration of the Crossover

In the following subsection, the authors illustrate how the Algorithm 3, the crossover of two buildings, looks when it is applied. First, for the initialization of the building design, 24,000,000 units are chosen as volume and 60,000 units as area. Figure 4 shows a crossover where the fittest buildings are selected according to the shadow length. Figure 4(a) shows how crossing parent 1 with parent 2 results in the child, and Figure 4(b) shows how parts of parent 1 intersect with parent 2, and result in a child, such as how the length of 600 units from parent 1 is inherited. By contrast, the width of 500 units with additional blocks on the right-hand side is bequeathed from parent 2.

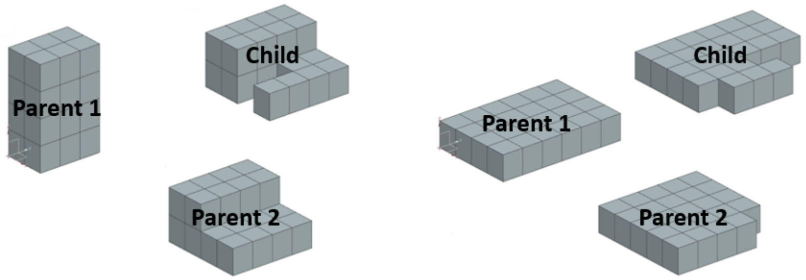

Figure 4: Two different examples of the crossover: (a) two different parents and the resulting child; (b) two different parents and the resulting child.

4.2 Illustration of the Shadow

For the illustration of the shadow green lines are chosen to cast the shadow from a direction where the sun is shining on the buildings. The building is placed at a latitude of $63.4°N$ and a longitude of $10.3°E$ with the volume and area as described in the preceding subsection (4.1). For the calculation and simulation, the authors chose 10 am on February 18, 2020. The results are illustrated in Figure 5.

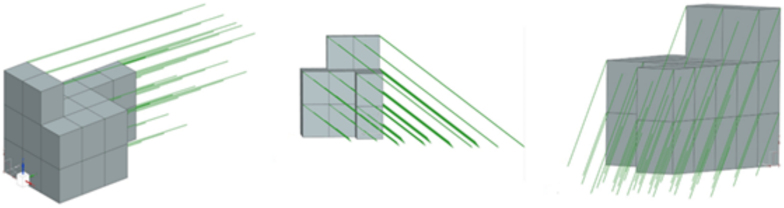

Figure 5: Example of a building design and its shadow cast at different angles at 10 am on February 18, 2020.

In winter in the Northern Hemisphere, the days are shorter, and the sun does not rise very high in the sky, whereas in summer, the days are longer, and the sun rises higher. The shadow length in June is illustrated in Figure 6. In June, the Southern Hemisphere is tilted away from the sun such that the sun's rays strike the ground at an angle, known as the angle of incidence. The angle of incidence is smaller in summer than in winter ([21]). Additionally, during the day, the shadow length changes: In the morning, the shadows are long and become shorter as the day progresses, then become longer in the afternoon. In the evening, shadows are cast in a different direction than the morning ([21]).

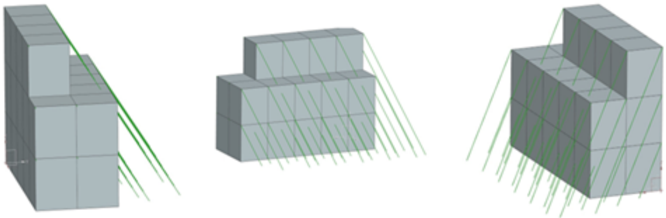

Figure 6: Example of a building design and its shadow cast at different angles at 10 am on February 18, 2020.

4.3 Analysis of the Fitness Function

Different variables, such as time, date, and month, influence the results of the fitness function. The same applies to changes in volume and area of the building design. For example, the taller the building, the longer the shadow length. Therefore, the results of the fitness function change.
Additionally, the operator crossover of the fitness function is examined, as it has an impact on the outcome of the building design. The results of the genetic algorithm after the 50th generation of building design are studied in the following subsections (4.3.1-4.3.3).

4.3.1 The Impact of Date on the Fitness Function

For the analysis of the fitness function, the time and date are set as 10 am on June 18, 2020. Furthermore, four building designs are initialized for the first generation. The fitness, of the initialized buildings is 180.99, 180.99, 271.49, and 271.49, respectively. By applying the operators' selection and crossover on the initialized buildings, six offspring labeled "Child" are constructed. The offspring is illustrated in Figure 7(a).

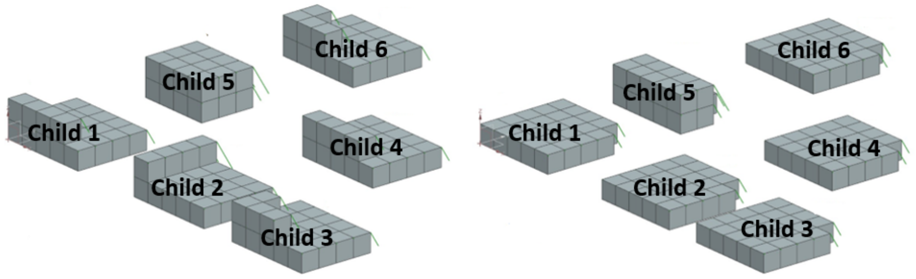

Figure 7: Two different examples of offspring for the same time (10 am on June18, 2020), but different generations: (a) the first generation; (b) the 15th generation.

The first offspring of the generation is placed at the coordinates [0,0], the second offspring at [900,0], the third at [1600,0], the fourth at [1000, 900], the fifth at [0, 800], and the sixth at [0,1600]. The coordinates of each offspring have enough distance between the buildings to avoid intersection. For further work, the placement of the building designs will be part of the input and strategic decision, therefore an additional parameter considered in the algorithm. According to the fitness results, all the offspring are fit enough to be considered a potential building design for the customer. Each of the building designs have the fitness result of 180.99.
After the 15th generation, the four buildings have the fitness results of 90.5, 452.48, 271.49, and 180.99, respectively. After applying the genetic algorithm, the results of the offspring indicate that all children qualify as fit. The fitness results of the offspring are 90.5, except for child 5, which has a fitness result of 180.99. Child 5 is not qualified as having sufficient fitness. The generation is illustrated in Figure 7(b).
The date and time are changed to 10 am on September 18, 2020. The results of the initialized buildings are 390.94, 390.94, 781.88, and 390.94, respectively. All fitness results of the first generation of offspring, illustrated in Figure 8(a), are 390.94. Therefore, all building designs are fit and qualify for consideration.

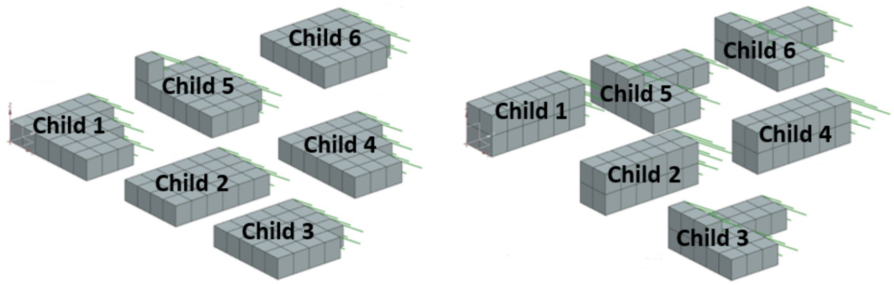

Figure 8: Two different examples of offspring for the same time (10 am on September 18, 2020), but different generations: (a) the first generation of offspring; (b) the 15th generation of offspring.

The 15th generation returns the following fitness results: 390.94, 390.94, 390.94, and 195.47. The offspring have a fitness result of 195.47, except for child 5, for which the result is 390.94. The results for the 15th generation are illustrated in Figure 8(b).

4.3.2 The Impact of Area and Volume

In this section, the impact of the area and volume on the buildings is analyzed. For the analysis of the fitness function, the time and date are kept constant, but the input data volume and area are changed. The date and time are 10 am on September 18, 2020.
The volume is changed from the initial volume of 24,000,000 units to 160,000,000 units, and then to 200,000,000 units. The increase in the volume, while the size of the area remains constant, results in taller buildings. Taller buildings cast longer shadows. The longer the shadow, the higher the fitness result numbers. Hence, the buildings' shadows impact the neighboring buildings. For the volume of 160,000,000 units, the fittest offspring has a value of 1759.23 units (see Figure 9(a)). When the volume is increased to 200,000,000 units, the fitness result for the fittest offspring is 195.47 units (see Figure 9(b)).

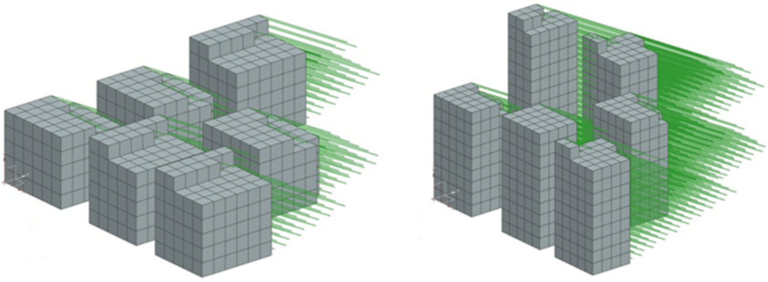

Figure 9: Two different building designs with different volumes: (a) with a volume of 160,000,000 units; (b) with a volume of 200,000,000 units.

For the analysis, the location of the offspring is changed for buildings that do not intersect each other. Figure 10(a) shows the building design with an area of 80,000 units, where the fittest offspring has a value of 195.47 units.
The offspring of the area of 120.000 units is illustrated in Figure 10(b), where the fittest offspring has a shadow length of 195.47 units. Regardless of how big the area is, the fitness results are the same. As the next step, the impact of volume change is analyzed for the fitness results.

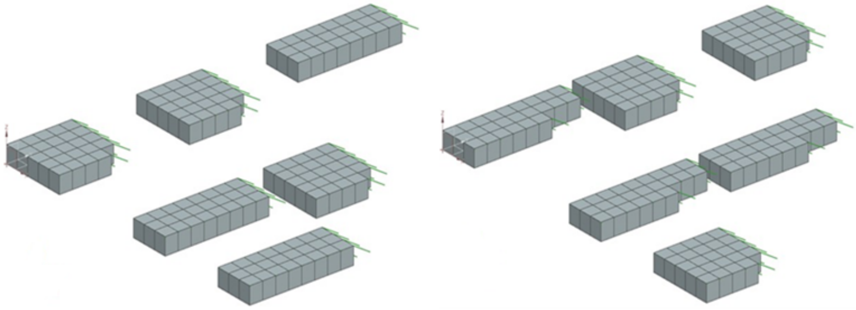

Figure 10: Two examples of building designs with different areas: (a) a building design with an area of 80,000 units; (b) the same building design with an area of 120,000 units.

The second offspring is placed at [1400,0], the third at [2900,0], the fourth at [1600, 1200], the fifth at [0,1400], the sixth at [0,2600], and "Child 1" stays at [0,0]. Changing the coordinates of the position of each building will ensure that the buildings do not intersect. The volume is equal to 160,000,000 units, and the area is 80,000 units. The result of the generation is illustrated in Figure 11(a), and the fittest offspring has a shadow length of 586,41 units. By increasing the area to 100.000 units, the output for fitness will increase to 977.35. The result of the fitness function is greater than the building design with an area of 80,000 units.

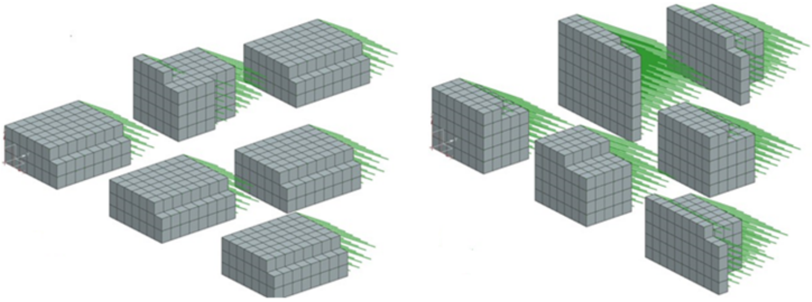

Figure 11: Two different examples of building designs with the same volume, but different area: (a) the building design with a volume of 160,000,000 units and an area of 80,000 units; (b) the building design with an area of 160,000,000 units and an area of 100,000 units.

Figure 12(a) shows the building design with a volume of 200,000,000 units and an area of 100,000 units. The fittest of these offspring has a shadow length of 586.41. By contrast, in Figure 12(b) the shadow length of the fittest offspring with an area of 120,000 units is 781.88 units. In conclusion, the larger the area, the less fit the offspring when the value of the volume is constant.

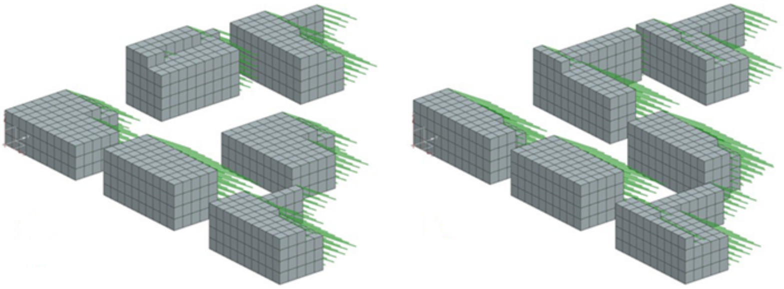

Figure 12: Two different examples of building designs with the same volume, but different areas; (a) the building design with a volume 200,000,000 units and an area of 100,000 units; (b the building design with a volume of. of 200,000,000 units and an area of 120,000 units.

4.4 Interpretation of Crossover Operator

Algorithm 3 represents the operator of the crossover of the genetic algorithm. It combines two parents into one list, and it orders the elements. However, the blocks can be ordered in multiple ways, depending on the scenario. For example, the ordering of the y-coordinate elements and then the z-coordinate elements are shown in Figure 12. The impact is visible on the right-hand building designs compared with those on the left, where the building blocks are added along the x-axis. Another example is the first building design (from the left) in Figure 12, where the rows are "full" along the x-axis. By ordering first by the z-coordinate and then by the y-coordinate, buildings are built up in height, which results in a longer shadow. The shadow is longer because

the order will focus on the first build-up to the highest building block possible and then fill it out along the x-axis. By ordering the building blocks first by the x-coordinate and then by the z- coordinate, the building blocks are assembled along the y-axis (see Figure 14).
The order of the building blocks depends on the case study and how the customer wants the building to be oriented. In Figures 13, 14, and 15, the first building design in the row is the same for each order. There are cases where the arrangement of assembling the building blocks does not make a significant difference. Nonetheless, it can be considered for the direction of the sunlight. The most common scenario is that the customer desires that as much sunlight as possible should enter the building complex.

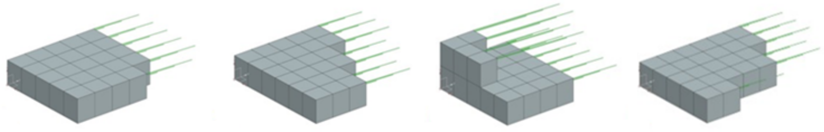

Figure 13: Ordering the coordinates of the building blocks from the lowest to highest, first by the y-coordinate and then by the z-coordinate.

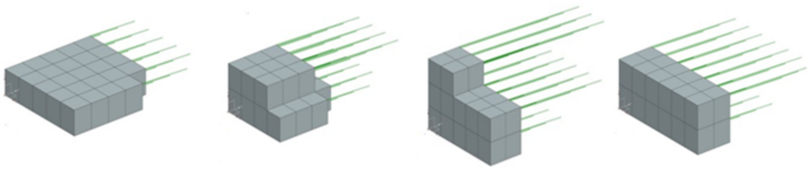

Figure 14: Ordering the coordinates of the building blocks from the lowest to highest, first by the z-coordinate, and then by the y-coordinate.

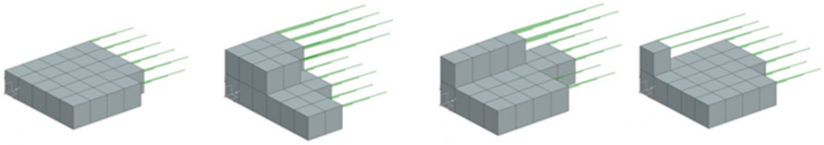

Figure 15: Ordering the coordinates of the building blocks from the lowest to highest, first by the x-coordinate, and then by the z-coordinate.

4.5 Effectiveness of Method

To understand the effectiveness of the method, the course of the fitness function values is analyzed in this paragraph of the paper. The authors have picked two examples of building designs to illustrate how fitness values change from generation to generation. The time which is picked for the analysis is the 18th of September, 2020, at 10 am. As mentioned at the beginning of the chapter, the building designs are generated until the 50th generation for analysis.
Figure 16 represents the building design with the parameters of 200,000,000 units for the volume, and 120,000 units for the area. On the x-axis, the number of generations is displayed, whereas on the y-axis the fitness

function values, or the output values, are presented. The offspring consists of 6 children which are marked with different colours. The pink area in the figure represents the difference in the fitness function values between the different children. As illustrated in figure 16, until generation 20 there a bigger jumps happening within the output values, and the difference is greater as well when the output values are compared between the different children. This is an indicator that some of the generated building designs are not fit enough to be selected in a closer selection of the fittest building designs. After generation 20 the pink area becomes smaller, and the output values of the different children are closer to each other. After generation 40 the fitness function values are established much more and the fittest building design can be chosen which fulfills the requirements of the design as needed. As it is detectable from the figure as well, the children can have similar output values as well, which in some cases are identical. Furthermore, it is noticeable that some building designs offer quite early in the course of fitness function values a very fit offspring. This is nonetheless influenced by the initial output value of the generation which it is started with and is not true for the overall process of the method to generate.

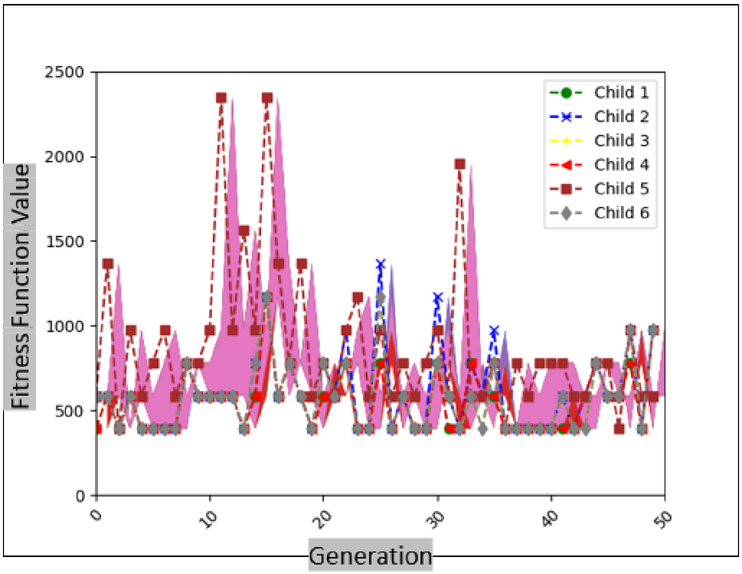

Figure 16: Graph of fitness function values for 50 generations with a volume of 200,000,000 units and an area of 120,000 units.

Figure 17 represents the building design with the parameters of a volume of 120,000,000 units and an area of 120,000 units. The highest jump within the course of the fitness function values occurs at the beginning of the fitness function value graph, which is before generation 10. After generation 10 the output values are more stable which can also be identified by the smaller range of output value fluctuations, and no further jumps like in the beginning. The output values stabilize some generations before 40 and after. The pink area is smaller as well which is a great indicator as well that the building designs are fitter and allows choosing from a bigger group of options.

It is important to mention that every time the genetic algorithm is run again, starting from generation one, the initial starting output values change slightly and therefore the output values over the progress are different. Nonetheless, the course of the fitness function values are similar and do not have much of a difference.
The results of the fitness function values measure the effectiveness of the method, which is validated by the simulation of the building design. As analyzed above and illustrated in Figures 16 and 17, the fitness function

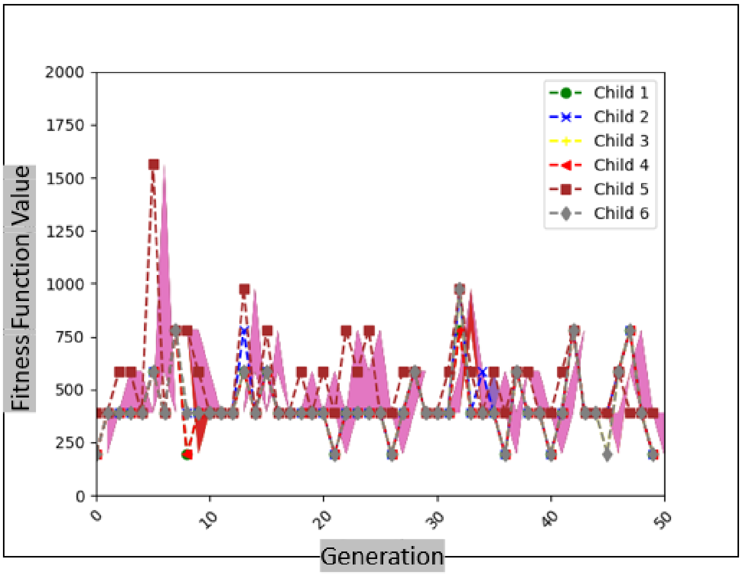

Figure 17: Graph of fitness function values for 50 generations with a volume of 120,000,000 units and an area of 120,000 units.

values become more stable the more generations pass. This is shown by more similar building designs being generated or re-occurring.
The analysis in this paper cannot be directly compared to other research studies due to the use of the simulation software Siemens NX, which is not commonly applied in building design. Additionally, the language of knowledge fusion has the potential to be utilized in different use cases, making the results a discovery with great potential for further research. While other studies have explored shadow analysis on building designs, none have used the same software or the concept of building blocks to construct buildings in various ways.
The study is replicable, and readers can obtain similar results by adopting it to other case studies, making it versatile and customizable for specific needs. The empirical evidence proves that the algorithm's performance progressively enhances over time, indicating its efficacy, which is documented in the improvement of the results and illustrated in the simulations.

5 DISCUSSION AND CONCLUSION

The proposed model in the paper is classified as a parametric object-oriented design. The building is illustrated as an object, and the behavior is analyzed through parameters in a three-dimensional setup. The attributes of the parameters - the volume, area, and building blocks - can be changed and can generate new buildings after each iteration. The parameters are implemented in Python, and the results are illustrated in Siemens NX. The knowledge-fusion language is used in Siemens NX as a tool for object-oriented design. It is possible to generate several buildings in one iteration and to apply the parameters to each building. The genetic algorithm is based on the shadow length as a fitness function. It is divided into three major sections. First, the building designs are initialized by using building blocks and constructing the design based on volume and area. With the operator of selection of the genetic algorithm, pairs are created based on the fittest parents. And lastly, the operator crossover sorts the elements of the building blocks according to the direction of the axis on which the building is being assembled.

Any parameters visualized by a customer can be added to the model, which makes the model extendable. The model is systematically and dynamically changeable, and it is adaptable for different use cases. The unique feature of the proposed model is that engineers receive feedback from the system and are in control of changing the attributes according to the needs and wishes of the customer. Specifically, the engineer does not need to switch between different software to analyze the shadow, transport, and other parameters but can do all the necessary calculations and evaluations in a single system.

Buildings are supposed to have a certain distance from each other, and the distance influences the view or impacts the shadow of the neighboring buildings. For future research, the position of the buildings should be included as an additional input for the model. The authors use simplified equations for the illustration of the parameters. However, the simplified equations can be changed and replaced by complex equations. The automation process presented in this paper runs efficiently to find optimal solutions. Further research needs to be conducted on how the automated process is impacted when more parameters are included. Furthermore, no data are stored after the calculations. Instead, they need to be added manually if the designer wants to keep track of the results.

The authors selected Siemens NX as the platform for their research. Looking forward, other software tools can be explored in future studies to assess their comparisons, impact on results, and performance relative to Siemens NX. An instance of such a tool is Grasshopper for Rhinoceros, which is utilized in [20]. Within this context, the authors utilized the software for the purpose of parametric modeling.

Overall, this paper demonstrates that such a theoretical approach can be applied to different use cases. With the current model, primarily flat and short buildings are generated, as the shadow is the main parameter considered for the building design. The fitness function focuses on the shadow length. The application of a genetic algorithm has the advantage that several solutions are generated. The customer can choose the building design that best fits their needs. The model can be used to analyze building designs to create an innovative and sustainable city. The goal is to initiate healthy living conditions for buildings' inhabitants and to construct a sustainable building complex.

ACKNOWLEDGEMENTS

This research project is part of the PhD studies at Norwegian University of Science and Technology in Trondheim, Norway.

ORCID

Brikene Berisha, http://orcid.org/0009-0001-1233-5479
Andrei Lobov, http://orcid.org/0000-0003-2729-489X
Bjorn Andersen, http://orcid.org/0000-0003-0533-2816

REFERENCES

[1] Ali, H.A.E.M.; Al-Sulaihi, I.A.; Al-Gahtani, K.S.: Indicators for measuring performance of building construction companies in kingdom of saudi arabia. Journal of King Saud University - Engineering Sciences, 25(2), 125–134, 2013. ISSN 1018-3639. http://doi.org/https://doi.org/10.1016/j.jksues.2012.03.002.

[2] Andersen, B.; Fagerhaug, T.: Performance Measurement Explained: Implementing Your State-of-the-Art System. ASQ Quality Press, 2002.

[3] Aranburu, A.; Cotillas, J.; Justel, D.; Contero, M.; Camba, J.D.: How does the modeling strategy influence design optimization and the automatic generation of parametric geometry variations? Computer-Aided Design, 151, 103364, 2022. ISSN 0010-4485. http://doi.org/https://doi.org/10.1016/j.cad.2022.103364.

[4] Authority, N.B.: Technical regulations, 2017. https://www.eftasurv.int/cms/sites/default/files/documents/regulation-TEK17-Draft-regulatory-provisions-101116-oppdatert_EN.pdf.

[5] Bhattacharya, S.; Braun, C.; Leopold, U.: A novel 2.5d shadow calculation algorithm for urban environment. 274–281, 2019. http://doi.org/10.5220/0007748602740281.

[6] Budin, R.; Budin, L.: A mathematical model for shading calculations. Solar Energy, 29(4), 339–349, 1982. ISSN 0038-092X. http://doi.org/https://doi.org/10.1016/0038-092X(82)90249-3.

[7] Chanadda, P.; Pattana, R.: Efficient algorithm and computing tool for shading calculation. Songklanakarin Journal of Science and Technology, 28, 375386, 2006.

[8] Cheng, F.; Thiel, K.H.: Delimiting the building heights in a city from the shadow in a panchromatic spot-imagepart 1. test of forty-two buildings. International Journal of Remote Sensing, 16(3), 409–415, 1995. http://doi.org/10.1080/01431169508954409.

[9] Edwards, L.; Torcellini, P.: Literature review of the effects of natural light on building occupants, 2002. http://doi.org/10.2172/15000841.

[10] Ernst; Neufert, P.: Architects' Data. 4th ed., 2008. ISBN 978-1-4051-9253-8.

[11] Farzaneh, H.; Malehmirchegini, L.; Bejan, A.; Afolabi, T.; Mulumba, A.; Daka, P.P.: Artificial intelligence evolution in smart buildings for energy efficiency. Applied Sciences, 11(2), 2021. ISSN 2076-3417. http://doi.org/10.3390/app11020763.

[12] Fasoulaki, E.: Genetic algorithms in architecture: a necessity or a trend?, 2007. https://www.researchgate.net/publication/228584949_Genetic_Algorithms_in_Architecture_a_Necessity_or_a_Trend.

[13] Huang, Y.S.; Chang, W.S.; Shih, S.G.: Building massing optimization in the conceptual design phase. Computer-Aided Design and Applications, 12, 344–354, 2015. http://doi.org/10.1080/16864360.2014.981465.

[14] Hudson, R.; hudson, r.: Strategies for parametric design in architecture. An application of practice led research. Ph.D. thesis, 2010.

[15] Jabi, W.: Parametric Design for Architecture, vol. 11, 2013. ISBN 9781780673141. http://doi.org/10.1260/1478-0771.11.4.465.

[16] John A. Duffie, W.A.B.: Solar Engineering of Thermal Processes. 4th ed. ed., 2013. ISBN 9780470873663.

[17] Kim, J.; Kent, M.; Kral, K.; Dogan, T.: Seemo: A new tool for early design window view satisfaction evaluation in residential buildings. Building and Environment, 214, 108909, 2022. ISSN 0360-1323. http://doi.org/https://doi.org/10.1016/j.buildenv.2022.108909.

[18] Lim, Y.W.; Majid, H.; A Samah, A.; Ahmad, M.; Ossen, D.; Harun, M.F.; Shahsavari, F.: Bim and genetic algorithm optimisation for sustainable building envelope design. International Journal of Sustainable Development and Planning, 13, 151–159, 2018. http://doi.org/10.2495/SDP-V13-N1-151-159.

[19] Marsault, X.: A multiobjective and interactive genetic algorithm to optimize the building form in early design stages, 2013. http://doi.org/10.26868/25222708.2013.2217.

[20] Meiklejohn, E.; Hagan, B.; Ko, J.: Rapid sketching of woven textile behavior: The experimental use of parametric modeling and interactive simulation in the weaving process. Computer-Aided Design, 149, 103263, 2022. ISSN 0010-4485. http://doi.org/https://doi.org/10.1016/j.cad.2022.103263.

[21] Musser, G.: To every season there is a reason. Astronomical Society of the Pacific, 390 Ashton Avenue, San Francisco, CA 94112, 4–5, 1995. https://astrosociety.org/file_download/inline/be43a628-7247-444a-bc31-59f765bd070c.

[22] Pernodet, F.; Lahmidi, H.; Michel, P.: Use of genetic algorithms for multicriteria optimization of building refurbishment. Building Simulation, 2009. https://www.aivc.org/sites/default/files/BS09_0188_195.pdf.

[23] Saiful Islam, K.; Islam, T.: Analysis of building shadow in urban planning: A review. 1122, 2015.

[24] Trubiano, F.; Sadeghipour Roudsari, M.; Ozkan, A.: Building simulation and evolutionary optimization in the conceptual design of a high-performance office building, 2013. http://doi.org/10.26868/25222708.2013.1449.

[25] Tuhus-Dubrow, D.; Krarti, M.: Genetic-algorithm based approach to optimize building envelope design for residential buildings. Building and Environment, 45(7), 1574–1581, 2010. ISSN 0360-1323. http://doi.org/https://doi.org/10.1016/j.buildenv.2010.01.005.

[26] Von Buelow, P.: Genetically enhanced parametric design for performance optimization, 2011. https://www.researchgate.net/publication/257858831_Genetically_Enhanced_Parametric_Design_for_Performance_Optimization.

[27] Wang, B.; Malkawi, A.: Genetic algorithm based building form optimization study for natural ventilation potential, 2015. http://doi.org/10.26868/25222708.2015.2197.

[28] Zani, A.; Andaloro, M.; Deblasio, L.; Ruttico, P.; Mainini, A.: Computational design and parametric optimization approach with genetic algorithms of an innovative concrete shading device system. Procedia Engineering, 180, 1473–1483, 2017. http://doi.org/10.1016/j.proeng.2017.04.310.

[29] Zhu, G.; McArthur, J.: Automatic generation of architectural layouts using genetic algorithms in bim. 17–26, 2020. http://doi.org/10.46421/2706-6568.37.2020.paper002.