Abstract
This research presents a proof of concept for a framework that optimises urban neighbourhood geometry based on performance and floor area ratio. Using an open-source Artificial Neural Network and Cluster Oriented Genetic Algorithm, the framework predicts solar radiation on buildings. It optimises solar radiation and floor area ratio at the urban scale. The framework shows promise for predicting other simulation aspects at the neighbourhood scale. The final testing phase resulted in finding the 5th optimal solution for a higher floor area ratio and lower solar radiation in a pool of 23,383 different urban configurations, taking approximately 24 hours. This represents an efficient approach to optimising neighbourhood design in its early stages.

Highlights
- a framework for optimising urban neighbourhood geometry,
- Using an Artificial Neural Network (ANN) to predict solar radiation on buildings,
- A novel geometrical classification method is used to create training datasets for the ANN,
- The Cluster Oriented Genetic Algorithm (COGA) tool optimises neighbourhood design in its early stages more efficiently than traditional simulation methods,
- Saves significant computational time and allows for performance-based design decisions in the early stages.

Introduction
Optimisation seeks the best iteration in a group of iterations or options (De wilde, 2018). This is a decision-making process taken within the project's design and analysis phase. In his book Building Performance Analysis, De Wilde (De wilde, 2018) discusses the utilisation of this decision-making process, either to choose a particular element like an optimal material or to find some optimal performing designs of design based on a preset benchmark or to combine both approaches in the search for the optimal solution for a design problem. This design nature had an open discussion in (Cross, 2001) review about the design discipline versus design science. He provides a review of the nature of design problems in the built environment and the debate about ways to “scientise” it. This research can partially relate to this dialogue as “quantifying” the design problem. That review also discusses design methods and definitions and how it can be addressed in different ways. It differentiated between the knowledge of design, which needs to be transferred, copied and inherited, and the design act itself, which must not be copied or inherited from previous practices. The urban context is a complex topic. Yet, analysing and optimising urban performance is a quantifiable process in nature which can be reused and inherited between different design practices. Given the complex nature of urban models, urban numerical optimisation would be exposed to ideas like hierarchal optimisation techniques (Choudhary, Papalambros and Rigo, 2014) provided three tiers of white, grey and black boxes. White box frameworks rely primarily on physics-based simulation, while black box frameworks rely mainly on data mining and analysis. Grey box frameworks, on the other hand, attempt to combine both physical-based models and statistical data models. This introduced a different use of the classification of black, grey, and white box tools introduced by (Wortmann and Nannicini, 2016, 2017), which was based on user interaction. Both classifications have agreed on the potentiality of managing a middle ground of allowing user interaction with optimisation tools while benefiting from the advancements of data mining and analysis methods in addition to the conventional physics-based simulation and modelling.

(Nguyen, Reiter and Rigo, 2014) reviews the simulation-based optimisation methods and tools design stages, the different algorithms used for the simulation-based performance optimisation, and how frequently each algorithm or method is applied. These algorithms represent a way of looking into a consistently changing search space due to the generative nature of the simulation inputs. This generative computational addition added some flexibility to the fact that optimisation is usually
This research introduces a proof of concept for a novel framework that optimises urban neighbourhood geometry based on its performance and floor area ratio. This framework adopts an open-source Artificial Neural Network (ANN) to predict direct solar radiation on buildings for newly generated neighbourhoods in Aswan, Egypt, as an example of a hot arid zone. A novel geometrical classification method was the foundation of creating the training database for the ANN and for classifying newly generated neighbourhood geometries for optimisation goals. Both classification method and ANN development and results were first introduced and discussed in (A. M. H. Lila et al., 2021; A. M. H. Lila & Lannon, 2019).

This research introduces a proof of concept for a novel framework that optimises urban neighbourhood geometry based on its performance and floor area ratio. This framework adopts an open-source Artificial Neural Network (ANN) to predict direct solar radiation on buildings for newly generated neighbourhoods in Aswan, Egypt, as an example of a hot arid zone. A novel geometrical classification method was the foundation of creating the training database for the ANN and for classifying newly generated neighbourhood geometries for optimisation goals. Both classification method and ANN development and results were first introduced and discussed in (A. M. H. Lila et al., 2021; A. M. H. Lila & Lannon, 2019).

**Figure 1** a) show case top view, b) show case perspective, c) show case classification tag

This proposed framework focuses on its user interaction with its multiple stages of generation, prediction and optimisation. At the same time, it manages to be coupling between the building physics simulation for training an ANN to predict the solar radiation of neighbourhhood designs. This paper discusses the different trials of applying Cluster Oriented Genetic Algorithm (COGA) to optimise urban neighbourhood design in its early stages, providing a performance-based framework and illustrating the results of this process.

**Methods**

This research utilises urban geometry classification method that built the training datasets for solar radiation simulation results and generates the new tested iterations for the optimisation stage. The study aims to save computational time to allow performance-based design decisions to be included in the early stages of urban design. The first stage of this study was setting the parameters and input for the framework. The framework was conducted in Grasshopper modelling platform (Scott Davidson, 2017). The inputs varied between generating parameters like setting the neighbourhood boundaries, street network and urban void locations: and analytical parameters like the individual building’s surrounding height analysis, buildings’ exposure to urban voids and building plot number of edges. Table 1 shows the generation modelling different parameters and its capability to be set by the framework users. The classification and generation phase is discussed in detail in (Lila and Lannon, 2019).

**Table 1 Different modelling stages and their contribution to scale and role type**

<table>
<thead>
<tr>
<th>Analytical</th>
<th>Generation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial boundary input</td>
<td>urban, user input enabled</td>
</tr>
<tr>
<td>Street network &amp; buildable areas</td>
<td>urban, user input enabled</td>
</tr>
<tr>
<td>Urban blocks orientation</td>
<td>urban</td>
</tr>
<tr>
<td>Building orientation</td>
<td>typology</td>
</tr>
<tr>
<td>Urban void generation</td>
<td>urban, user input enabled</td>
</tr>
<tr>
<td>Urban void exposure</td>
<td>urban</td>
</tr>
<tr>
<td>Number of edges</td>
<td>typology</td>
</tr>
<tr>
<td>Building courts</td>
<td>typology, user input enabled</td>
</tr>
<tr>
<td>Height distribution</td>
<td>typology, user input enabled</td>
</tr>
<tr>
<td>Surrounding heights comparison</td>
<td>urban</td>
</tr>
</tbody>
</table>

During the simulation, the weather file used was the Egyptian Typical Meteorological Year (ETMY) for Aswan city, located in southern Egypt at 24.0889° N and 32.8998° E. The geometries under test were divided into grid cells measuring approximately 15 meters by 15 meters. These cells were used to determine the amount of direct solar radiation falling on the surfaces of the geometries. The sky matrix used in the simulation was the Tregenza sky matrix, which is the default setting of the tool and was introduced in a study by (Lee, Geisler-Moroder and Ward, 2018).

Following that, an ANN based on this novel geometrical classification of urban solar radiation and using it to feed for a COGA tool for optimisation goals is a new method to save simulation time. After developing an open-source python-coded ANN node inherited in Grasshopper modelling platform. It could utilise the classified database to predict the performance for both the building and urban scale (Lila, Jabi and Lannon, 2021). the used tool, Biomorpher (Harding, 2017), proposed a K-Means application to achieve clustering on an NSGA II genetic algorithm. The combination of the two principles...
provided the clustering capabilities to the conventional application of GA. COGA is used in this framework to conduct the optimisation part to look for the optimally performing urban geometry based on the prediction inputs coming from the ANN output.

The criteria for this optimal solution were the solar radiation performance for the urban scale geometry and the Floor Area Ratio (FAR) of these configurations. The two factors should conflict as the test site (Aswan, Egypt) falls in a hot arid zone. This causes the solar radiation to be higher with higher buildings that need a high value of FAR. The tool used to conduct this optimisation in the framework is called “Biomorpher” (Harding, 2017). It is a tool that applies a GA method named a cluster-oriented genetic algorithm. The significant difference of this method is its capability of approximating the available pool of iterations into regions to find the closest optimal area instead of running the GA evaluation on each individual in its population. This helps save time for the generation. Biomorpher has been embedded into the framework to implement GA optimisation because it controls the process and visualises the options. Additionally, the clustering approach of the GA makes the implementation computationally efficient. The tool is also an open source tool and allows the control of the process at each generation allowing for further user interaction through the evolutionary solving process.

The initial stage of testing

The first test to implement the GA was set to have a 1,000 pool of iterations as the test's limit. It aimed to reach the largest possible FAR from these iterations with the lowest solar radiation possible. These two values were input from the framework’s different classes. The FAR for each configuration was calculated by multiplying the buildable area by the number of buildings, with each building having a floor height equal to 3 metres high. Then the sum of this process is divided by the total area of buildable plots in the configuration at hand. The trained ANN class of the framework calculates the performance prediction. The sum of the values is the prediction fed to the GA component to act as fitness for the generation process.

The other inputs to Biomorpher’s GA component are the slider responsible for controlling the iteration process as a “genome” and the geometry of the urban configuration itself after converting it to a "mesh" geometry. The genome is the name used to refer to the identifying features of each tested iteration. The mesh is one of the geometrical modelling representations used by the software platform.

The first trial was to test the use of Biomorpher finding an optimal solution within 1,000 urban configurations. This sample of 1,000 urban configurations was part of a 23,383 pool of iterations as a starting point for this testing stage. The generation settings were to avoid all mutation, the crossover was set to be 10% of the population, and 100 populations for each generation were set for this testing phase. This test aimed to find the result of using Biomorpher as a typical GA application where the generation and offspring selection is conducted automatically based on fitness preference. As mentioned earlier, the fitnesses were set to reduce solar radiation while achieving the maximum FAR possible.

Another generation trial was done by selecting the optimal clusters for each generation to create the offspring of the following generation. This test was done for a greater number of generations and consequently for a longer time. Second framework test with 12,000 available iterations

The large pool of iteration testing

The next test followed the same procedure to find an optimal solution within a larger pool of iterations. In this test, the collection of iterations was half of the available iterations by the generation parameters (12000 iterations). The optimal solution chosen from this test took 13 generations with an hour for each generation, and the setting was a 50 population per generation with no mutation and 10% crossover.

The last stage of this test was to test the optimisation capability within the boundaries of the whole pool of iterations allowing Biomorpher to look for an optimal solution within the large database of configurations. In this test, Biomorpher setting was 100 populations for each generation and the same settings as the previous test when

Figure 2 general diagram of the proposed workflow and its different stages
it came to crossover, which was 10%, and no mutation was permitted.

**Results & Discussion**

**First stage results**
The test has aimed to get only five continuous generations to test the time consumption too. The whole GA run took around 500 minutes. This test result came up with a high value of the FAR within the highest 15% of the iteration pool. Yet, the solar radiation was not within the same ratio of the lowest solar radiation saved results of the iteration pool.

The second trial of the same pool shows the visual difference between the GA-found result and the top two optimal choices with priority to FAR values over the whole 1,000 results pool.

**Figure 3 The generation plot of the two fitnesses in the first GA testing**

However, this run has shown the capabilities of the tool. The sequence of development through the five generations is shown in Figure 3 for both the fitnesses. It shows the continuous reduction of the solar radiation fitness in blue. Furthermore, it shows the increase of the FAR value, in red, then it declined for the 5th generation, which was led by the prioritisation of lower solar radiation value. This process gave a clear understanding of the performance of the tool. Another trial was conducted with the same settings.

The second trial of the same pool had ten generations, and in each generation, the largest FAR and the lowest solar radiation value clusters were selected to continue the process. It took around 12 hours to finish this trial. The final 12 clusters of the last generation of this test are shown in Figure 4; each configuration represents a cluster.

**Figure 4 the last 12 clusters of the algorithm’s 10th generation for the 1000 iterations pool test**

The representative iteration that had the highest FAR value was shown in cluster 0. It had a higher value of FAR of 14.01707 with solar radiation of 900.5328 kWh/m². This iteration falls in the four highest FAR values in the whole pool of 1,000 iterations. One of the higher iterations with a higher value of FAR has a higher value of solar radiation too, which excludes it from being an optimal choice for the two performing aspects. Figure 5 shows the visual difference between the GA-found result and the two performing aspects. The gap between the two fitnesses throughout

**Figure 5 highest 10 FAR values in grey with its associated solar radiation results in blue, and the COGA found iteration in a dashed line**

the plot shows the progress made by the algorithm looking for the highest FAR in red, and the lowest solar radiation result, in blue. Although the FAR peaked in the 9th generation, the relook of the next generation’s cluster resulted in a slightly lower solar radiation option by the 13th generation, as shown by the difference in the gap between the two fitnesses in Figure 6.

Another note on the results of this test is that the highest FAR representative cluster does not have the best-performing solar radiation. The higher FAR cluster had a solar radiation of 1294.7 kWh/m² with a FAR of 16.64; this was cluster number 6 within the last generated 12 clusters, while cluster 11 had better performance for both fitnesses, not just FAR. The representative cluster 11 had significantly lower solar radiation of 906.3 kWh/m² while the FAR value was also lower, with a small difference. This representative had a 16.2 FAR value which is not different from the highest FAR achieved by this generation. This was the cluster with the number 11, as shown in Figure 7.
When comparing this result with the saved simulation result, this urban configuration had an optimal performance considering the two fitness performances. To show this, the list of the saved performances and FAR values was resorted in order from larger to smaller based on the FAR values. This is done to prioritise the FAR value for neighbourhood design, which is a usual constraint for the designer to achieve a higher FAR, allowing for more economic value for the land use.

As shown in Figure 8, the selected representative cluster achieved the lowest solar radiation result within the highest 40 FAR values in the pool of iterations for this test, marked with a dashed line, while its FAR value is the 12th highest value in a pool of more than 12,000 iterations. The following higher values of FAR in the graph do not have a lower solar radiation than the selected representative cluster.

The final test was conducted over the whole configuration, while the prediction results are the sum of predictions of each building in that configuration. Its number does not just select the configuration; the urban generation parameters are also checked to be the same features for discussed configurations.

This larger number of populations made the generation take a longer time than previously. In this test, each generation took around one hour and 40 minutes. The 4th generation showed a near-optimal performing representative cluster in comparison with the previous test results. This iteration has a 1081.61 kWh/m² solar radiation value and FAR value of 15.21.

Biomorpher did not find a better-performing solution until generation 20. Figure 9 shows the representative clusters of the last four generations of the process and their results for solar radiation in kWh/m² with a blue dot beside it and the FAR values of the iterations with a red dot beside it. For this reason, the test reinstated the generation 4 results and tried to drive the generation process towards a better-performing option by only selecting the most optimal representative clusters. Reinstating means getting Biomorpher back to the stage of generating a certain generation. After this the test continued for another four generations with this new selection of clusters. Yet, it has shown a slightly better-performing iteration. This iteration has a solar radiation value of 1204.8 kWh/m² and a FAR value of 15.84.

The final trial was to start a new GA test, but this time instead of selecting a random set of the population for the initial generation, it was set to start with this solution. This time it took three generations to get a better-performing iteration. It is important to know that these three generations did not try to look far from the starting point of the first set of populations. This might raise queries about the tool’s capability of examining the selected iterations and the diversity of its generation process if it is only focusing on iterations with relative settings. The third generation had a cluster with a solar radiation value of 1147.65 kWh/m² and a FAR value of 16.57, as shown in Figure 10.
This phase of testing has shown the importance of user interaction with the GA generation stage of the framework. As shown in the multiple trials, the test took different directions to look for better-performing iterations while it reduced the time consumed by directing the generation breeding process, which is a rare feature in the available tools applying GA principles in the same platform. Moreover, it has shown a limitation of the tool. When it starts the proceedings with a given set of iterations, it does not look further within the pool of available iterations. It is important to note that this generation process was done without reviewing the simulations in the saved database of urban configurations. After finishing this optimisation process, the comparison was made as done in previous tests with the saved simulation results to get the location of the GA optimal solution against the optimal solution of the saved results. The highest FAR iteration results are shown in Table 2, with the GA-selected iteration highlighted in colour. Its order as the 5th optimal solution can be seen more clearly in Figure 11. This result shows the potential of using this framework to provide guidance through the early stages of urban design is capable of reaching this efficient level of accuracy considering the time saved from simulating the full number of iterations to reach the optimal solution.

Table 2 shows the top 20 FAR configurations from the whole pool of 23,383 iterations, with the optimal solution highlighted.

<table>
<thead>
<tr>
<th>Order based on higher FAR</th>
<th>Solar Radiation Results in kWh/m²</th>
<th>FAR Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.048.129</td>
<td>17.60369</td>
</tr>
<tr>
<td>2</td>
<td>1.013.694</td>
<td>17.5942</td>
</tr>
<tr>
<td>3</td>
<td>1.063.583</td>
<td>17.37746</td>
</tr>
<tr>
<td>4</td>
<td>994.0439</td>
<td>17.22447</td>
</tr>
<tr>
<td>5</td>
<td>1.085.866</td>
<td>17.16701</td>
</tr>
<tr>
<td>6</td>
<td>1.554.55</td>
<td>17.13628</td>
</tr>
<tr>
<td>7</td>
<td>1.020.48</td>
<td>17.07668</td>
</tr>
<tr>
<td>8</td>
<td>919.1503</td>
<td>17.0214</td>
</tr>
<tr>
<td>9</td>
<td>1.046.329</td>
<td>17.01338</td>
</tr>
<tr>
<td>10</td>
<td>1.009.887</td>
<td>16.95517</td>
</tr>
<tr>
<td>11</td>
<td>922.0213</td>
<td>16.77168</td>
</tr>
<tr>
<td>12</td>
<td>932.8184</td>
<td>16.71138</td>
</tr>
<tr>
<td>13</td>
<td>975.5042</td>
<td>16.64382</td>
</tr>
<tr>
<td>14</td>
<td>977.3871</td>
<td>16.56767</td>
</tr>
<tr>
<td>15</td>
<td>778.3127</td>
<td>16.55688</td>
</tr>
<tr>
<td>16</td>
<td>955.3986</td>
<td>16.44351</td>
</tr>
<tr>
<td>17</td>
<td>1.080.064</td>
<td>16.4123</td>
</tr>
<tr>
<td>18</td>
<td>886.6475</td>
<td>16.39362</td>
</tr>
<tr>
<td>19</td>
<td>1.157.974</td>
<td>16.30851</td>
</tr>
<tr>
<td>20</td>
<td>785.624</td>
<td>16.216</td>
</tr>
</tbody>
</table>

After these different trials, the GA pointed out the configuration with the 14th-highest FAR value. This is the 5th lowest solar radiation in the highest 20 FAR values in the 23,383 urban geometrical iterations pool. The highest FAR results comparison to its Solar Radiation Performance of the database.

Figure 11 shows the highest 40 FAR values in grey with its associated solar radiation in blue from the full pool of iterations and the COGA found iteration in the dashed line.

It’s important to note that some trials of this framework’s optimal solution did not fully cover the rectangular boundary due to the generation of urban voids and some configuration process limitations. This allowed for its high value of FAR. The limitation of the geometry generation tool produced some configurations that need to be deselected from the GA selection process. Furthermore, some configurations do not fully restrict the boundary rectangle due to some limitations in the package used in the geometry generation process. This was handled by excluding these iterations before feeding it to the prediction and optimisation phases.

Another potential limitation of the study is that there is a difference in values between the GA selection of optimal performing solar radiation predictions and the saved database of solar radiation simulation results. This is due to the difference between ANN predictions and the actual simulation result and the fact that the database of urban configuration performance results was saved out of simulations.

**Conclusion**

The literature has discussed the continuous development of computational modelling for the built environment and highlighted the complexity of challenges in creating urban models. Several studies have reviewed other aspects of urban geometry modelling and its impact on urban performance aspects using different scale models for simulation. Some of these studies have investigated the possibilities of utilising this impact in an optimisation process or method. As mentioned in the literature, time consumption is also another challenge in searching for performance-based design on an urban scale. This has directed the framework to be built as a proof of concept utilising parametric modelling data flow to break down the urban modelling complexity. The whole framework aims to resolve the time consumption challenge and allow...
for an early stage of design performance-based optimisation. The tests set the proof of concept stage of the framework to aim for optimising the neighbourhood geometry based on solar radiation, and it was conducted on one weather file for the city of Aswan, Egypt, which was classified as a hot arid zone climate condition.

The first stage of building the framework was to create a parametric workflow that could break down the neighbourhood model into classified geometry based on its features. The features were based on the literature and pilot studies focusing on location, orientation, building area, height, typology and the surrounding geometry context condition. The parametric flow was divided into a series of classes for conducting this classification process. The framework created a classification tag for each building on a parallel sequence to contain this data following the same classification roles. Only in this sequence it added and manipulate text rather than building models. It showed the framework’s capability to generate different geometrical iterations and allowed the settings and parameters of the generated iterations to be changed to open up for further modelling and generating capabilities. In order to utilise these capabilities in urban solar radiation performance, the following stage was to build a database of classification tags attached to their solar radiation simulation results.

The third stage of creating this proof of concept framework was to adopt machine learning principles within the framework to provide the capability of computational prediction of solar radiation performance, aiming for this to save the simulation time consumption without sacrificing the accuracy achieved by the conventional simulation engines. The last version of this application was initially an adaptation of an open-source code of an Artificial Neural Network (ANN) with some adjustments to meet the framework goals. The results of testing this ANN prediction had a positive linear correlation to its equivalent simulation results for both architectural and urban scales. The prediction accuracy results were within acceptable measures when compared to recent literature. The time-saving for the framework was significant when replacing the prediction with the traditional simulation process. These significantly positive accuracies and time-saving results allowed the framework to go to its last aimed step, answering the last question about the capability of optimisation in the early stages of design.

The fourth and last stage of building this framework was to include the node responsible for conducting the optimisation process. This was done by a cluster-oriented genetic algorithm application in Grasshopper parametric platform called Biomorpher. The initial phase of testing the GA principles in the framework involved comparing the optimised results with the saved database of FAR and solar radiation results for three different database sizes. The framework was run on 1,000 urban configurations in the first test and obtained the third optimal solution compared to the saved simulation database. The test ran for five generations and took about four hours using Biomorpher clustering GA application with ANN prediction.

In the second test, more than half of the saved database was used, and the importance of selecting the generation clusters feature in Biomorpher was emphasised to eliminate the iterations that were not affected by the generation process limitations. This test achieved the lowest possible solar radiation in the top 20 FAR from the testing sample.

In the final test, the entire saved urban configurations database was used. The test revealed a limitation of the GA tool in terms of its ability to overcome its local optimal solution and search for better results with different settings. This limitation can be overcome by user interaction and alternative selection. The final database testing phase found the fifth optimal solution for higher FAR and lower solar radiation in a pool of 23,383 different urban configurations. The researcher interacted with the genetic algorithm to activate the cluster selection option, taking about 24 hours to run. These results were conducted within less than 15% of the time consumed in simulating the available iterations.

The following step of testing the framework will utilise this set of classifications, prediction and optimisation on an existing neighbourhood boundary and test its integration effectiveness to optimise an already designed neighbourhood. Another future aim of this research is to test its applicability in similar weather files with the same climate conditions and test its accuracy and time saved against conventional brute force simulation methods.

References


Lila, A. M. H. and Lannon, S. (2019) ‘Classifying urban...
geometry impact on solar radiation, in Corrado, V. et al. (eds) the International Building Performance Simulation Association (IBPSA) 2019 16th Conference, Rome, Italy.


