

# MULTI-OBJECTIVE FACADE OPTIMIZATION FOR DAYLIGHTING DESIGN USING A GENETIC ALGORITHM

Jaime M. L. Gagne<sup>1</sup> and Marilyne Andersen<sup>1</sup>  
<sup>1</sup>Massachusetts Institute of Technology, Cambridge, MA

## ABSTRACT

A building's facade design has significant impact on the daylighting performance of interior spaces. This paper presents a tool based on a genetic algorithm (GA) which facilitates exploration of facade designs generated based on illuminance and/or glare objectives. The method allows a user to input an original 3d massing model and performance goals. The method assumes that the overall building form remains the same while the facade elements may change. Ten facade parameters are considered, including glazing materials and geometric characteristics of apertures and shading devices. A simple building information model (BIM) is used to automatically generate a 3d model of each individual. Results from single and multi-objective case studies are presented to demonstrate a successful goal-driven design exploration process.

## INTRODUCTION

The facade design of a building is possibly the most critical element in creating a successful daylighting scheme on the interior. Numerous studies have already demonstrated the potential for genetic algorithms (GAs) to facilitate performance-based facade design exploration (for example, Caldas and Norford 2002; Torres and Sakamoto 2007; Wright and Mourshed 2009). However, previous studies have restricted the scope of the problem by fixing the initial geometry of the space and the main optimization objective (typically minimizing energy consumption). Such restrictions are limiting in an actual design scenario, as users may not be able to model a problem that is relevant to their specific design goals and aesthetics.

This paper presents a GA-based method for facade design exploration which can be integrated into the design process. The proposed method allows an unprecedented number of user inputs, including an original 3d massing model. A simple building information model (BIM) has been created to allow the user-defined massing model to be understood by the

system. The BIM enables the system to recognize the geometrical characteristics of the user's initial model and to automatically generate new 3d models as specified during the GA process. BIMs have also been proposed in the past as a way to integrate optimization into the design process by allowing designers to use optimization in familiar CAD-based settings (Geyer 2009).

The proposed method also allows the user to define his or her own performance objectives. The method uses two daylighting metrics, one for illuminance and one for daylighting-based glare, to enable a comprehensive understanding of daylighting performance in a space. The user defines the number and location of illuminance and glare sensors within his original model, and he or she also inputs a desired illuminance goal range for each illuminance sensor plane.

GA studies, due to the large number of iterations involved, are typically time-consuming processes. The proposed method attempts to create a more efficient GA-based tool in two ways: it uses an efficient simulation and rendering engine which has been validated against Radiance, and it uses micro-GA algorithms, which use a very small population size to greatly reduce the total number of necessary simulations.

This paper presents two case studies for which the proposed GA-based approach has been used. The first case study is a single-objective problem (illuminance only) while the second case study is a multi-objective problem (illuminance and glare). In both situations, the proposed method was able to successfully explore the design space and present the user with a design solution or set of solutions which approach the user-defined performance objectives.

## PROPOSED APPROACH

### **Efficient Simulation Engine**

Due to the nature of GAs, in which many populations must be simulated before a solution has been reached,

an efficient simulation engine is a necessity. The engine used in the proposed approach, the Lightsolve Viewer (LSV), is a hybrid global rendering method which combines forward ray tracing with radiosity and shadow volumes rendering (Cutler et al. 2008). This engine was chosen because it allows for rapid calculation of the daylighting metrics described in the previous section. Cutler et al. found that a rendered scene in LSV took approximately 10 seconds while an analogous “fast rendering” in Radiance was completed in approximately 5 minutes. Early validation results indicated that rendered images by LSV displayed a pixel difference of less than 10% from Radiance for a variety of scenes, camera positions, and daylighting conditions (Cutler et al. 2008).

To make the whole-year simulation more efficient, the LSV engine divides the year into 56 periods and calculates the illuminance during each time period under four different sky types ranging from overcast to clear using the method described in Kleindienst et al’s paper (2008). The climate-based illuminance is then calculated for each time period as a weighted average of illuminances from each sky type. In this study, the total computation time for a full-year simulation with illuminance and glare results ranged from less than 1 minute for a simple model to about 5 minutes for a more complex model. An analysis comparing

illuminance data calculated on point sensors in Radiance with area-based patch sensors in LSV indicated similar values (5% median, 7% mean, and 28% maximum relative difference) for a model similar to those considered in the present study (Lee et al. 2009).

### Daylighting Fitness Metrics

To allow for a comprehensive understanding of daylighting performance, the proposed approach features two different types of fitness metrics, one for illuminance levels and one for glare. The goal-based illuminance metric requires the user to input four illuminance values: acceptable low, desired low, desired high, and acceptable high; the user must also specify which time periods of day and seasons he or she is interested in: morning, mid-day, afternoon, and winter, spring/autumn, summer. This metric is a numerical version of the graphical metric presented in (Kleindienst et al. 2008) and uses the same logic for climate and temporal simplifications. The metric assumes a user-defined sensor plane which will be divided into small patches during the simulation process. For a single patch, the goal-based illuminance metric is defined as the percentage of the user’s times and seasons of interest in which daylight provides an illuminance within the user’s specified range. The final goal-based illuminance for a sensor is an average of the performance over all patches on a sensor plane. For illuminance levels which fall between the “acceptable” and “desired” values, partial credit is given (Figure 1a). A value of 100% indicates that the entire area of the sensor plane sees an illuminance in the user’s desired range over all periods of day and seasons of interest.

The glare metric used in the proposed approach is a model-based approximation of Daylighting Glare Probability (DGP) developed by Kleindienst and Andersen (2009). The DGP metric, originally introduced by Wienold and Christoffersen (2006), indicates the percent of occupants disturbed by a daylighting glare situation. The model-based DGP approximation method (DGPm) is an efficient way of approximating the DGP which uses the LSV engine, and when compared to the DGP as calculated using the evalglare program in Radiance, the DGPm has been found to perform within a 10% error over 90% of the time (Kleindienst and Andersen 2009). This method was tested only on rectangular spaces and one limitation is that models should not include window mullions. The metric assumes a user-defined vertical sensor plane whose normal indicates a direction of view, and it considers windows as the only sources of glare. To evaluate glare risks, the present study uses the most recent glare thresholds described by Wienold

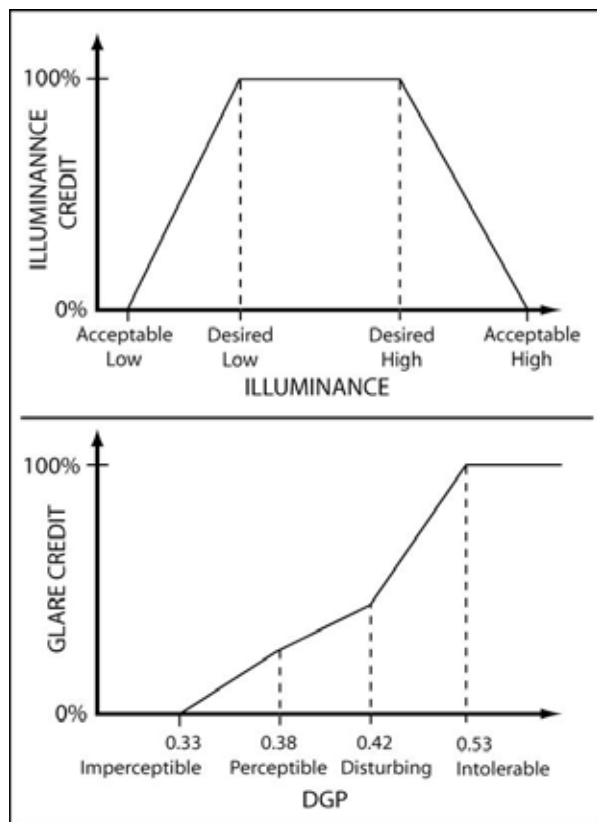


Figure 1 Diagrams indicating system of full and partial credit for (a) illuminance and (b) glare

(2009), where any value below 0.33 (imperceptible glare) is considered a “no glare” situation and given a glare credit of 0, any value above 0.53 (intolerable glare) is given a glare credit of 1, and all values in between are weighted accordingly (Figure 1b). These glare credits are averaged across all glare sensors which face the same general direction within the model. A value of 0% indicates that the specified view is likely to produce no glare due to daylighting.

Because the daylighting performance metrics are defined as percentages, the objectives for any user-defined problem should be the same: maximize the goal-based illuminance on all illuminance sensors and minimize the model-approximated DGP on all glare sensors. This formulation allows for the same search algorithm to be used for any set of user-specified goals.

**User Inputs: Massing Model and Performance Goals**

The proposed method allows a much larger number of user inputs than a typical GA study. In particular, one innovation is to allow the user to create a 3d massing model in Google SketchUp instead of requiring them to use a default model. This user-defined massing model should specify all desired opaque material properties. Those facades which will be generated by the GA must be labeled with the material name “GA\_WALL”.

Within the 3d model, the user must also specify 2d sensor planes on which either illuminance or glare will be calculated. The model can accommodate any number of these sensor planes. The user must also indicate a desired illuminance goal range in lux (lx).

This set of user inputs allows a designer to customize both the design and performance goals in a way that requires only modeling, not programming. This feature is a significant step towards enabling a GA-based design approach that can be used by designers with no programming skills.

**Facade Variables**

Ten different facade variables are considered, as indicated in Table 1, along with the minimum and maximum values they can take and the step sizes. These parameters were chosen because they are typically considered early in the design process and frequently have a large impact on both exterior aesthetics and on daylighting performance. The full set of values is encoded into a string of 30 bits for each separate facade considered.

*Table 1 Facade parameters considered*

<b>Façade Parameter</b>	<b>Min</b>	<b>Max</b>	<b>Step</b>
Window-to-Wall Ratio	0.1	0.8	0.1
Number of Windows	1	8	1
Aspect Ratio*	Thinnest	Widest	-
Vertical Location*	Lower Bounds	Upper Bounds	-
Horizontal Location*	Right Bounds	Left Bounds	-
Window Distribution*	Windows Touching	Far Apart	-
Overhang	No	Yes	-
Fins	No	Yes	-
Length of Shading Devices	0.5ft	4ft	0.5ft
Total Glass Transmissivity	10%	85%	5%
Percent (Specular) Transmission	0%	100%	12.5%
<i>* Absolute values of max and min for these parameters will depend on user-defined geometry</i>			

**MICRO-GENETIC ALGORITHMS**

Genetic algorithms, or GAs, (Goldberg 1989) have been applied to many types of architectural problems. During the GA process, a set of initial feasible solutions (a population) is chosen or generated at random. Each member is evaluated and members that result in good performance are used as “parents” for a new generation. Since this new generation is based on the best performing feasible solutions in the previous solutions, we assume that some members of the new generation will perform better. Once evaluated, the good performers are used as parents while the poor performers are discarded. The cycle is continued until a suitable solution or set of solutions is found.

Genetic algorithms typically require large population sizes and numbers of generations to converge. In order to improve efficiency of the proposed approach, a micro-GA was used. A micro-GA is a GA in which population size is very small, thus reducing the total number of simulations necessary. Micro-GAs have been successfully implemented for building performance optimization by Caldas (Caldas and Norford 2002; Caldas 2008). The proposed approach allows for both single- and multi-objective problems, which both utilize a micro-GA algorithm. The single-objective problem considers illuminance only, while the multi-objective problem considers both illuminance and glare.

### Single-Objective Micro-GA

The proposed approach uses the original micro-GA algorithm as described by Krishnakumar (1989) for a single-objective problem. Encoding is done using binary strings. The algorithm uses a very small population size (5 members). For the single-objective problem, fitness is defined as the goal-based illuminance for a single sensor plane or the average goal-based illuminance over multiple sensor planes. The process is as follows:

1. Generate a random population of 5 binary strings.
2. Evaluate fitness and carry over best string into the next generation (elitist strategy).
3. Use deterministic tournament selection for adjacent pairs to select remaining four strings for reproduction, i.e. the member of each pair with the best fitness is used to produce the next generation. The current best string is also allowed to compete.
4. Apply uniform crossover with no mutation. This strategy creates two child strings from two parent strings by swapping individual bits with a probability of 0.5
5. Check for bitwise convergence (which occurs when all strings differ by 5% or less). If converged, keep best string and randomly generate 4 new ones.
6. Go to step 2.

### Multi-Objective Micro-GA

For a multi-objective problem, it is assumed that the first objective is to maximize the the goal-based illuminance on all illuminance sensors and the second objective is to minimize the model-approximated DGP on all glare sensors. The micro-GA has been successfully used for multi-objective problems (Coello Coello and Pulido 1993) by including external memory which stores pareto solutions generated over the course of the process. For this study, the algorithm used is similar to that described for single-objective problems, with the addition of an external memory similar to that described by Coello Coello and Pulido. This external memory enables the retention of all non-dominated solutions generated. At each step, the memory is updated to include new non-dominated solutions, and any previous solutions which are dominated by new ones are then removed. An pseudo-pareto front is approximated as those solutions contained within the external memory after a certain number of generations. The multi-objective process is as follows:

1. Generate a random population of 5 binary strings.

2. Evaluate fitness and carry over one non-dominated into the next generation (elitist strategy).

3. Save all non-dominated solutions into external pareto memory. Remove all previous solutions in memory which are dominated by the new solutions.

4. Use deterministic tournament selection for adjacent pairs based on non-dominance to select four strings for reproduction.

5. Apply uniform crossover with no mutation.

6. Assume convergence after 4 generations. If converged, keep one non-dominated string and randomly generate 4 new ones.

7. Go to step 2.

It is important to note that while this process does produce a set of non-dominated solutions which may approximate the pareto front, it does not necessarily generate a true pareto front with evenly distributed solutions. However, a true pareto front may not be required for designers who wish only to see a range of possible solutions. Additionally, the large number of random solutions introduced into the population after convergence should create better distribution of results.

### BUILDING DATA MODEL

One of the novel features of the proposed approach is the ability for the user to provide a 3d model as input instead of requiring programming or the use of a default model. To provide this functionality, a building data model was developed whose values are automatically assigned once the process is initiated. The model contains information about each building element in a 3d model and the relationships between them. The general structure of the data model is indicated in Figure 2. Each building element object contains information about its location, geometry, orientation, and material.

The building data model allows the algorithms in the proposed approach to understand which walls are to be manipulated by the GA and what the boundary conditions of those walls are. It also allows the system to automatically create new 3d models of each GA population member which can then be simulated during the GA process. These user is thus automatically provided with a 3d model of any solution found by the GA process.

### Automated Building Model Population

In the proposed approach, a building model can be automatically populated using a 3d model in Google SketchUp. Identification of each building element

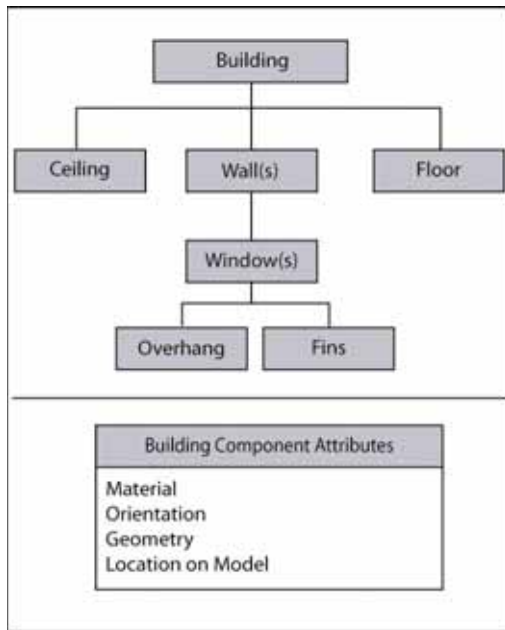


Figure 2 Simple BIM schematic: (a) relationships between components and (b) object attributes

occurs using a series of logic statements, and element attributes are then determined using information available from SketchUp about each face. The logic assumes that the model uses a few basic guidelines: any plane that represents a sensor (for either illuminance or glare) must have the word “SENSOR” in its material name, any plane that represents an external shading device must have the word “EXTERNAL” in its material name, any plane that is manipulated by the GA must have the words “GA\_WALL” in its material name, and the normal vectors of all faces should point towards the interior of the space.

Assuming these guidelines are met, the logic used to identify each element is as follows (assume all elements are faces):

1. If the face is not opaque and not called “SENSOR”, it is a window.
2. If the face is opaque and called “EXTERNAL”, it is a shading device.
  - a. If the normal points up or down, it is an overhang.
  - b. Else, it is a fin.
3. If the face is opaque and not called “EXTERNAL”:
  - a. If the normal points up, it is a floor.
  - b. If the normal points down, it is a ceiling.
  - c. Else, it is a wall.
4. If the face called “SENSOR”, it is a sensor plane.

Once the individual building elements have been identified, a second set of logic is used to determine the appropriate relationships between elements. This logic

determines the child-parent relationships between walls and windows and between windows and shading devices. The logic for determining these relationships is as follows:

1. Assigning windows to walls: For each window, cycle through all walls. If both elements have the same orientation, and if the window location lies between the edge boundaries of the wall, assign that window to that wall.
2. Assigning shading devices to windows: For each shading device, cycle through all windows. If two vertices of the overhang is located 2 inches or less from two vertices of the window (top two vertices for overhangs, right or left vertices for fins), assign that shading device to that window.

An initial massing model may or may not include windows and shading devices. If the model does include these elements, they will remain the same through the GA process. Only those walls that have been labeled “GA\_WALLS” will have generated facades.

### Automated Model Generation

Another feature of the proposed approach automatically generates 3d model representations of the binary strings created during the GA process. These models are created in Google SketchUp using the following process:

1. Add a single window of the given window-to-wall ratio to the facade using the same aspect ratio as the wall itself to ensure fit.
2. Divide into the given number of windows.
3. Calculate the highest and lowest possible aspect ratios that the windows can take based on the window size and wall dimensions. Change aspect ratio of all windows based on given value.
4. Calculate the largest distance that can exist between each window based on window size and wall dimensions (assume smallest distance is 2 inches). Change distribution based on given value.
5. Determine upper, lower, left, and right wall boundaries. Change window group location based on given value.
6. Add shading devices of the given length, if applicable.
7. Change window material given values.

Because the geometrical parameters (window aspect ratio, location, and distribution) are calculated based on



the boundary conditions of a given facade instead of being based on absolute values, the proposed approach can generate models using any type of original massing geometry that features vertical walls facing cardinal directions. The user can also choose to rotate the sky so as to simulate models whose walls are orthogonal but which are not aligned with the cardinal axes. This feature provides the user with a great deal of flexibility when creating the original massing model.

## VALIDATION

To ensure that the micro-GA algorithm was behaving as expected, a set of test studies were performed on a simple box model with a single illuminance sensor plane located in the center of the space at workplane height. For each of these studies, the south and east facades were generated by the GA while the north and west facades remained opaque. Two situations were explored: “no minimum” (200 lx desired max, 400 lx acceptable max, no minimum values) and “no maximum” (400 lx desired min, 200 lx acceptable min, no maximum values). In both cases, one or more solutions to the problem were known to exist.

For each case study, the GA process was run three times to determine the general behavior of the algorithm. For the “no minimum” case, the micro-GA found a solution that met the goals over 100% of the time and sensor area considered in all three trials within 10 or fewer generations. For the “no maximum” case, the micro-GA was run for 10 generations each time and the three trials yielded solutions that met the goals for 96.7%, 98.8%, and 99.4%.

Both case studies were considered successful, although some limitations to the GA method were seen in these initial trials, including inconsistencies in the number of generations required to find a good solution and the possibility for the algorithm to get “stuck” in one part of the solution space. However, these studies also demonstrated the potential for the micro-GA to effectively search a broad design space and to converge onto successful designs quickly.

## CASE STUDIES

### Single-Objective Case Study (Illuminance Only)

The proposed GA approach was applied to the massing model shown in Figure 3 in Boston, MA. This model has a non-rectangular footprint and a slanted roof condition. The facades of interest in this model are those facing north and south. It has two illuminance goals that were not considered conflicting. Both sensor planes are located at workplane height. The illuminance goals for the west sensor are 200 lx

(acceptable) and 400 lx (desired) lower bounds; no maximum. The goals for the east sensor are 100 lx (acceptable) and 200 lx (desired) minimum; 800 lx (desired) and 1000 lx (acceptable) maximum.

The micro-GA process was run for a total of 25 generations. The fitness in this case study was calculated as the mean of the goal-based illuminance metric for both sensors. Therefore, a value of 100% would indicate that the entire area of both sensors would be within the specified illuminance ranges throughout the whole year. The population average and best fitness for each generation are shown in Figure 4. After 25 generations, the best solution was found to have an average fitness of 90.2% (individual fitnesses for the two sensors were 96.7% and 83.7%). The final solution facades both have windows concentrated towards the west side of the space as expected based on the specified goals.

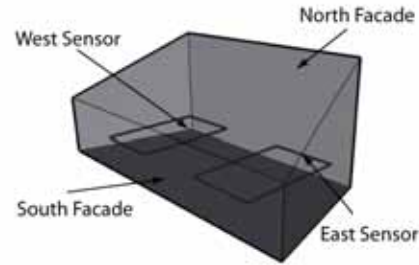


Figure 3 Case study #1 original massing model and sensor locations

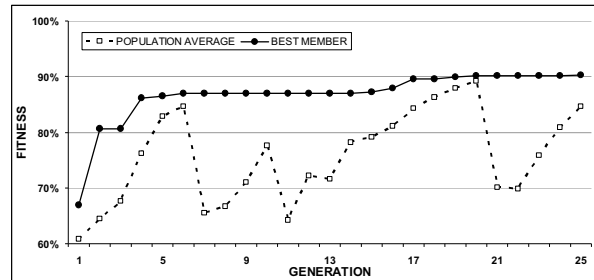


Figure 4 Case study #1 - Population best and average fitness over 25 generations

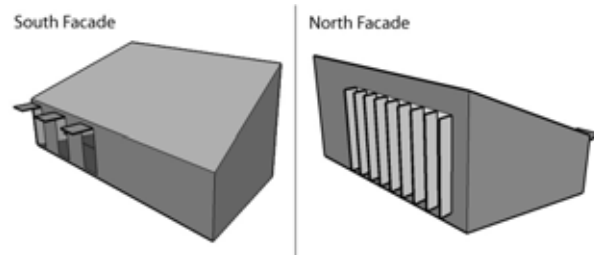


Figure 5 Case study #1 - Best performing design after 25 generations

## Multi-Objective Case Study (Illuminance and Glare)

The multi-objective approach was applied to the massing model shown in Figure 6 in Boston, MA. In this model, the two facades of interest are facing east and west. An additional constraint is added to this problem in that the two facades of interest must maintain a uniform aesthetic. This constraint ensures that a single optimal solution for both illuminance and glare would not be found. To enforce this constraint, the same binary string was used for both facades. Two illuminance sensors are included, each with the same illuminance goal ranges (200 lx acceptable low, 400 lx desired low, no maximum). Additionally, glare sensors facing towards the east and west facades are considered. These sensors are indicated in Figure 6.

A pseudo-pareto front was created after running the micro-GA process for a total of 20 generations, as indicated in Figure 7. It is clear from the pareto front that the two goals are conflicting, although many designs have been found which come very close to meeting the illuminance goals while still keeping glare relatively low. A subset of solutions from the pseudo-pareto front has been selected to show the variety of solutions found (Figure 8).

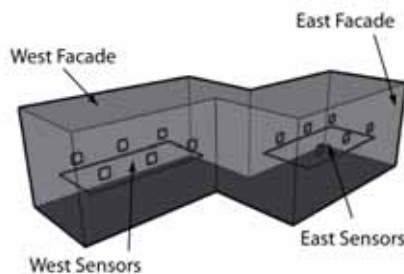


Figure 6 Case study #2 original massing model and sensor locations

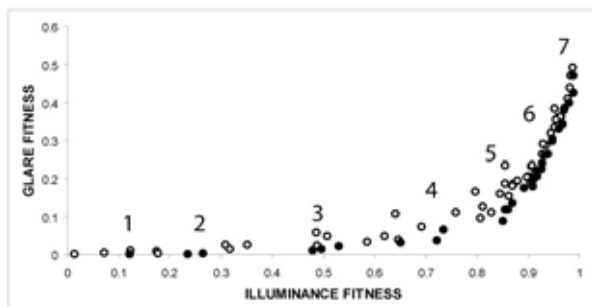


Figure 7 Pareto front after 20 generations (with subset of 7 designs selected)

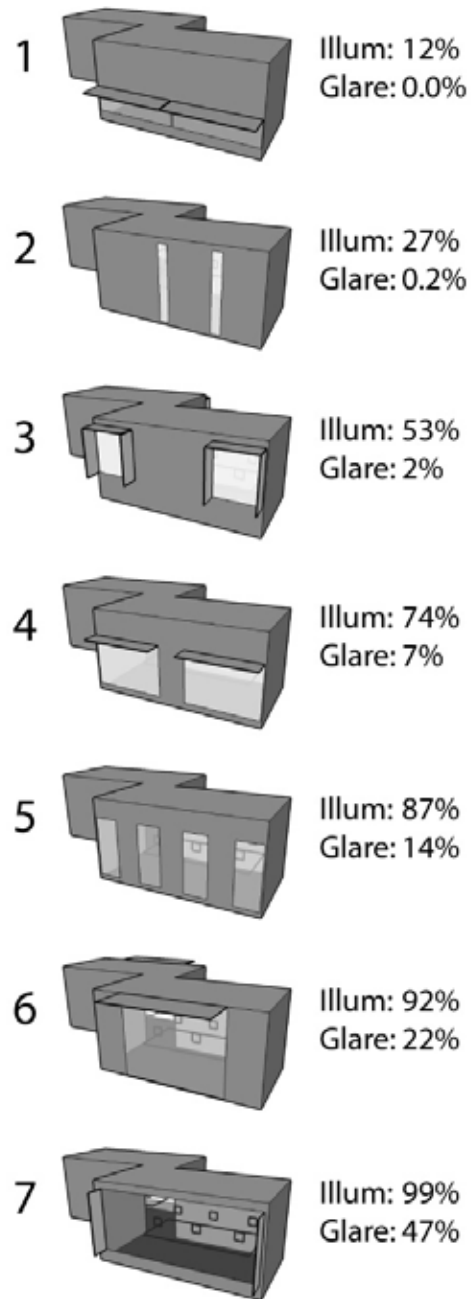


Figure 8 Seven example designs from the Pareto front

## CONCLUSION

This paper has presented a GA-based approach which enables performance-based exploration of facade designs. This method combines an efficient micro-GA algorithm with a large number of user inputs, including an original 3d massing model and user-specific performance goals. Such an approach is powerful

because it allows an infinite number of possible design scenarios to be considered without changing any code. In doing so, it allows users who only have modeling experience, not programming experience, to use GAs during the design process.

Two case studies were presented which showed the performance of the single and multi-objective micro-GA search processes. The multi-objective case study in particular demonstrated the range of possible design solutions that a user can obtain using the pareto front.

GA-based approaches still have several limitations. One of these is the lack of consistency in the final solutions found, since the randomly generated initial design solutions play a large role in determining which subsequent designs are found. This limitation can be solved to some degree by running many generations, but this approach adds additional time to an already time-consuming process. One other limitation is the tendency for GAs to get “stuck” in a solution that is only a local minimum or maximum. However, for the purposes of performance-based design exploration, it is not necessary to find a global optimum; rather, it should be sufficient to present the user with a design or set of designs which the user will then use as an initial design rather than a final one.

The approach demonstrated in this paper is a first step towards integrating GA-based search into the design process. Future work could focus on the addition of penalty functions to enforce user-specified constraints, the creation of a more flexible encoding method to allow the user to choose which parameters to consider on each facade, or the inclusion of additional performance criteria such as solar heat gains or energy consumption.

## REFERENCES

- Caldas, L., 2008. Generation of energy-efficient architecture solutions applying GENE\_ARCH: An evolution-based generative design system. *Advanced Engineering Informatics*, 22(1), 59-70.
- Caldas, L.G., Norford, L.K. 2002. A design optimization tool based on a genetic algorithm. *Automation in Construction*, 11(2), 173-184.
- Coello Coello, C.A., Pulido, G.P. 1993. A micro-genetic algorithm for multiobjective optimization, *Lecture Notes in Computer Science* no.1993.
- Cutler, B., Sheng, Y., Martin, S., Glaser, D., Andersen, M. 2008. Interactive selection of optimal fenestration materials for schematic architectural daylighting design, *Automation in Construction*, 17(7), 809-823.
- Geyer, P. 2009. Component-oriented decomposition for multidisciplinary design optimization in building design, *Advanced Engineering Informatics* 23(1), 12-31.
- Goldberg, D.E., 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley.
- Kleindeinst, S., Andersen, M. 2009. The adaptation of daylight glare probability to dynamic metrics in a computational setting, *Proceedings of LuxEuropa 2009, Istanbul*.
- Kleindeinst, S., Bodart, M., Andersen, M. 2008. Graphical representation of climate-based daylight performance to support architectural design, *Leukos*, 5(1), 39-61.
- Krishnakumar, K. 1989. Micro-genetic algorithms for stationary and non-stationary function optimization, *SPIE Proceedings: Intelligent Control and Adaptive Systems*.
- Lee, J., Andersen, M., Sheng, Y., Cutler, B. 2009. Goal-based daylighting design using an interactive simulation method, *Proceedings of Building Simulation 2009, Glasgow*.
- Torres, S.L., Sakamoto, Y. 2007. Facade design optimization for daylight with a simple genetic algorithm, *Proceedings of Building Simulation 2007, Beijing*.
- Wienold, J. 2009. Dynamic daylight glare evaluation, *Proceedings of Building Simulation, 2009, Glasgow*.
- Wienold, J., Christoffersen, J. 2006. Evaluation methods and development of a new glare prediction model for daylight environments with the use of CCD cameras. *Energy and Buildings* 38(7), 734-757.
- Wright, J., Mourshed, M. 2009. Geometric optimization of fenestration, *Proceedings of Building Simulation 2009, Glasgow*.



SimBuild 2010

I or we, the author(s) of the submitted paper, have read the following Copyright Transfer and Disclaimer, and agree to them by submitting the submitted paper.

- (1) The author(s) affirm that the paper has not been published elsewhere and, if the paper is accepted, will not be published elsewhere prior to SimBuild 2010.
- (2) If the paper is accepted, the author(s) will automatically grant to IBPSA-USA a nonexclusive, royalty-free, perpetual, worldwide, irrevocable, sub-licensable, and transferable license to publish the paper (in unmodified form) in any fashion (including but not limited to inclusion in the SimBuild 2010 CD proceedings, via electronic means such as the world wide web, and inclusion in future compilations of papers). This "nonexclusive" license means that the author(s) are not restricted as to future use of the material except that exclusive rights cannot be granted to another.
- (3) The author(s) affirm that they have the right to grant the license specified in (2), that is, publication by IBPSA-USA or its licensees will not be in conflict with copyright or other restrictions.
- (4) The author(s) acknowledge that acceptance of the paper does not imply IBPSA-USA's endorsement of or agreement with the ideas presented. Under no circumstances shall IBPSA-USA be liable for any damages resulting from use information included in the paper.

*Figure 2 This text MUST be included in the body for paper submission, at the end of your paper on a separate page*