An Interactive Performance-Based Expert System for Daylighting in Architectural Design

by

Jaime M.L. Gagne

Bachelor of Arts, Engineering Sciences (2003)
Harvard University

Master of Architecture (2006)
University of Pennsylvania

Master of Science, Architecture (2007)
University of Pennsylvania

Submitted to the Department of Architecture
in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy in Architecture : Building Technology

at the

Massachusetts Institute of Technology

February 2011

Signature of Author: ____________________

Department of Architecture
January 7, 2011

Certified by: ____________________

Leslie K. Norford
Professor of Building Technology
Thesis Supervisor

Certified by: ____________________

Marilyne Andersen
Associate Professor of Sustainable Construction Technologies, EPFL
Thesis Co-Supervisor

Accepted by: ____________________

Takehiko Nagakura
Associate Professor of Design and Computation
Chairman, Department Committee on Graduate Students

Thesis Committee

Leslie K. Norford
Professor of Building Technology
Department of Architecture
Massachusetts Institute of Technology
Thesis Supervisor

Marilyne Andersen
Associate Professor of Sustainable Construction Technologies
School of Architecture, Civil and Environmental Engineering
École Polytechnique Fédérale de Lausanne
Thesis Co-Supervisor

Luisa Caldas
Associate Professor of Architecture
Faculty of Architecture
Technical University of Lisbon
Reader

An Interactive Performance-Based Expert System for Daylighting in Architectural Design

by

Jaime M.L. Gagne

Submitted to the Department of Architecture on January 7th, 2011
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Architecture: Building Technology

Abstract

Design practitioners are increasingly using digital tools during the design process; however, building performance simulation continues to be more commonly utilized for analysis rather than as a design aid. Additionally, while simulation tools provide the user with valuable information, they do not necessarily guide the designer towards changes which may improve performance. For designing with daylighting, it is essential that the designer consider performance during the early design stage, as this is the stage when the most critical design decisions are made, such as the overall building geometry and façade elements. This thesis proposes an interactive, goal-based expert system for daylighting design, intended for use during the early design phase. The system gives the user the ability to input an initial model and a set of daylighting performance goals. Performance areas considered are illuminance and glare risk from daylighting. The system acts as a "virtual daylighting consultant," guiding the user towards improved performance while maintaining the integrity of the original design and of the design process itself.

This thesis consists of three major parts: development of the expert system, implementation of the system including a user interface, and performance assessment. The two major components of the expert system are a daylighting-specific database, which contains information about the effects of a variety of design conditions on resultant daylighting performance, and a fuzzy rule-based decision-making logic, which is used to determine those design changes most likely to improve performance for a given design. The expert system has been implemented within Google SketchUp along with a user interface which allows a designer to fully participate in the design process. Performance assessment is done in two ways: first by comparing the effectiveness of the system to a genetic algorithm, a known optimization method, and second by evaluating the success of the user interactivity of the tool, its use within the design process, and its potential to improve the daylighting performance of early stage designs.

Thesis Supervisor: Leslie K. Norford
Title: Professor of Building Technology

Thesis Co-Supervisor: Marilyne Andersen
Title: Associate Professor of Sustainable Construction Technologies, EPFL

Dedication

This thesis is dedicated to the memory of my mother, June Warren Lee, whose gentle strength is ever with me.

Acknowledgments

The work presented in this dissertation would not have been possible without the help and support of a great number of people. First and foremost, I would like to thank Marilyne Andersen for being a wonderful advisor and mentor and for inspiring this research. It was a privilege to work with her, and I would like to thank her for her excellent advice, attention to detail, sense of humor, and endless support.

I would also like to thank Les Norford and Luisa Caldas for their constructive and insightful feedback, with special thanks to Les Norford for embracing his late hour role as a co-advisor and for always helping me to remember the bigger picture. Thank you to all three of my committee members for always believing in my work and pushing my research one step further.

Thank you to the many people who worked on the Lightsolve program and who helped to provide the framework for my research: Barb Cutler and Yu Sheng from Rensselaer Polytechnic Institute for all their work on the Lightsolve rendering program, Phil Seaton for creating the Lightsolve and expert system interfaces, and especially Sian Kleindienst for implementing Lightsolve and for truly paving the way.

I would like to thank the MIT Department of Architecture, the Building Technology program, and Marilyne Andersen for financial support, as well as the Martin Family Society of Fellows for Sustainability.

Although I cannot name them directly, I would like to thank all the participants of my user study for testing my system and for providing me with valuable feedback.

One of the best things about doing research at MIT is having the opportunity to meet so many talented and interesting people. I had a wonderful time working in the Building Technology Lab, and I would like to thank all the BT faculty, students, and staff for creating such a warm and supportive environment. Thank you to my fellow daylighters, Sian Kleindienst, Shreya Dave, Kevin Thuot, and Siobhan Rockcastle, for the many conversations about our work, and in particular to Shreya and Kevin for feedback on the user study design. Thanks to David Quinn for introducing me to LaTeX, and to Steve Ray for being such a friendly lab neighbor. I would especially like to thank Jennifer Zessin, Carrie Brown, and Alejandra Menchaca, who were with me from the beginning until the end, and who were always available to help me work through new ideas, despite our differences in research areas.

A huge thank you goes to my family and friends for the many hugs along the way. Thank you to my father, William Lee, who instilled in me a love of knowledge and who taught me the value of education. Thanks also to my brother, Daniel Lee, who never doubted me for a moment. Most of all, I would like to thank my husband Josh, whose support and encouragement never wavered and whose patience is apparently boundless. I could not have done it without him.

Contents

List of Figures

Part I

Introduction

Chapter 1

Introduction

1.1 Motivation for Daylighting

Throughout history, daylighting has been integral to architectural design, and its importance to occupant health and comfort has long been understood in the contexts of both architectural theory and practice. Daylighting strategies may enhance occupant experience of a space by providing light, views, temporal and seasonal cues, and interesting visual effects such as sun patches and shadow patterns. Successful strategies will also maintain occupant comfort by shading direct sunlight, controlling glare, and limiting solar thermal gains. Illumination or shadow can be used to complement architectural features and augment the total visual effect of the space. Due to its influence on architecture through history, daylight is often considered a critical design element.

The earliest architects understood the importance of daylighting in their designs. Vitruvius, writing in the first century BC, believed that one should determine the function of interior rooms based on their orientations towards the sun, considering both daily and annual cycles (1960). Renaissance architects such as Alberti reiterated that idea, while adding that when orienting spaces, one must also consider the specific climate and site of the building (Alberti, 1988). During the 20th century, Le Corbusier and Frank Lloyd Wright both expressed their belief that light was essential to one's perception of a work of architecture. Wright referred to a building as "a creation of interior space in light" and believed that all spaces within a house should receive daylight at some point during the day (1955). Le Corbusier agreed, calling architecture "the masterly, correct and magnificent play of masses brought together in light" (1986). At a design level, there has always been reason for architects to consider daylight when creating spaces.

In recent decades, architects have been presented with additional motivation to use daylighting in buildings. Our concerns about global energy use have inspired many to investigate the potentials for daylighting and solar heat to replace or offset electric lighting and mechanical heating. In 2006, electric lighting accounted for an average of 17% of total site energy and 25% of total primary energy use in commercial buildings in the United States (U.S.DOE, 2009). The energy used for electric lighting also accounted for 25% of total carbon emissions from commercial buildings. The primary energy used for lighting is roughly equal to primary energy expenditures for space heating and space cooling combined.

As buildings consume more primary energy than any other sector in the U.S. (39% of the U.S. total) and as the U.S. consumes more primary energy than any other country (22% of the world total) (U.S.DOE, 2009), it is clear that there is the potential for significant energy savings by reducing the use of artificial lighting. Studies have shown that with the successful use of daylighting and lighting controls, one can provide adequate amounts of light in commercial buildings while reducing the use of electric lights by up to 80% (Bodart and Herde, 2002; Li and Lam, 2003; Ihm et al., 2009). Additionally, careful use of shading strategies may decrease a building's heating and cooling loads (Kolokotroni et al., 2004), thus further reducing the total primary energy consumption of the building.

An additional benefit of daylighting is the potential for improved productivity, learning, and overall health. Although this area of research is still new, early studies and surveys have shown positive effects of daylighting on office employees, such as improved moods, increased productivity and performance, and a decrease in common health problems such as headaches and eye strain (Rashid and Zimring, 2008). Studies have also found benefits for office management, such as decreased rates of employee absenteeism, decreased turnover rates, and increased profits due to improved productivity (Edwards and Torcellini, 2002).

Similar health and performance benefits have been found for students in daylit schools. Although direct causality is not yet known, several studies have shown that students in daylit schools may outperform students in non-daylit schools on standardized tests (Heschong et al., 2002; Plympton et al., 2000). Students in daylit schools may also experience health benefits, such as strengthened immune systems, increased growth, better eyesight, and fewer dental cavities (Edwards and Torcellini, 2002). Research in this area is still ongoing.

There are numerous additional possible benefits of daylight over artificial light for human health and healing. Recent studies indicate that the distinct spectral properties of daylight may control our circadian system, thus influencing our mood and behavior (Webb, 2006). A relationship has been found between the prevailing amount of bright sunlight and the brain's production of serotonin, which has been linked to disorders such as depression and seasonal affective disorder (Lambert et al., 2002). Case studies indicate that patients with these types of illnesses may recover faster in sunny hospital rooms than in dim rooms (Beauchemin and Hays, 1996). Sunlight also aids in our production of vitamin D, which some researchers believe may help protect humans from breast, ovarian, prostate, and colon cancers (Freedman et al., 2002; Lefkowitz and Garland, 1994). Finally, it has been shown that patients recovering from spinal surgeries may experience less stress and require less pain medication if their hospital rooms receive sunlight (Walch et al., 2005).

One final benefit of daylighting is that humans often express a preference for natural light over artificial light. One study of office workers found that 95% of those polled preferred daylight to artificial light, and that those employees seated closer to windows were more satisfied than those seated further from windows (Markus, 1967). Other studies have suggested links between daylighting and job satisfaction or even overall psychological well-being (Boyce et al., 2003b; Edwards and Torcellini, 2002).

In summary, designers have long considered daylight an important element for architectural expression. In recent decades, we have come to understand that daylighting may provide additional benefits, such as reduced energy consumption and improved occupant health and well-being. Detailed summaries of such benefits may be found in (Boyce et al., 2003a; Edwards and Torcellini, 2002; Rashid and Zimring, 2008). Nevertheless, simply providing daylight in a building will not always result in positive results. Daylighting is only as good as its delivery system, so careful design is necessary to ensure that enough light is available and that glare, shadows, and reflections are controlled (Boyce et al., 2003b). Unfortunately, it is often a challenge to create a successfully daylit building. The next section describes current digital tools used by architects and engineers to aid in designing with daylight.

1.2 Digital Tools for Daylighting Design

Successful daylighting schemes are difficult to achieve as architects must balance multiple issues such as energy, visual comfort, aesthetics, space planning, engineering, client need, and cost. To help designers deal with such complex situations, many different tools have been developed for daylighting design and analysis. Traditional tools include scale models, heuristics (rules of thumb or knowledge obtained from previous experience), design guides, and case studies. Such tools remain popular today, especially heuristics, which are more commonly used in the early design stage than other methods (Galasiu and Reinhart, 2008).

Over the past few decades, daylighting simulation software programs have been introduced as an additional tool for daylighting design. Recent surveys have indicated that simulation tools have gained acceptance for both professional (Reinhart and Fitz, 2006) and educational use (Sarawgi, 2006). These computer tools are varied in scope, accuracy, complexity, and intended user. Input may be numerical, geometrical, or both. Outputs range from purely quantitative (for example, calculated lighting metrics in tables or graphs) to purely qualitative (for example, photorealistic rendered images).

Tools developed for use in the early design stages tend to be simpler to learn and faster to use than tools developed for later design stages or for analysis. Such tools typically restrict the geometrical form of the simulated building or offer a finite number of geometry choices, and results are often limited to simple quantitative metrics (Hitchcock and Carroll, 2003; Lehar and Glicksman, 2007; Hviid et al., 2008). These tools seem primarily aimed for those who would like to estimate interior illuminance levels, solar heat gains, and energy consumption due to artificial lights, and they may be more useful for educational purposes than as design tools. Ecotect, a more advanced simulation tool which is still accessible to most designers, allows the user to explore more varied geometries, but displays only quantitative information about the space (Marsh, 2008b). Other tools allow the user to explore more complex geometries and aesthetic effects such as direct sunlight and shadows (Bund and Do, 2005; Google, 2008), providing qualitative information without quantitative data.

For later design phases, designers may use rendering software packages to produce photorealistic images from complex CAD-based models. Tools such as Autodesk's 3ds Max, Autodesk's VIZ, and AGi32 are examples of commercially available software which are popular among design students (Sarawgi, 2006). For professional use, the most commonly used tools are based on Radiance, which has long been the preferred lighting simulation engine for lighting experts despite its complexity (Roy, 2000; Reinhart and Fitz, 2006; Galasiu and Reinhart, 2008). All these tools require a significant time commitment for both the initial modeling and the simulation process itself. Commonly used simulation outputs from these tools include interior illuminance, daylight factor (DF), photorealistic images, interior luminance, electric lighting use, glare indices, and daylight autonomy (DA) (Reinhart and Fitz, 2006).

1.3 The Need for a New Method

While it is clear that simulation-based tools have gained acceptance among design practitioners, such tools have yet to achieve total integration into the design process. Experience and heuristics continue to be used more often than simulation tools during the conceptual and schematic design phases, the periods during which most major design decisions relating to daylighting are made (Galasiu and Reinhart, 2008). These early design phases, however, should be the ones where feedback is received and where the overall design proposal is first assessed (even roughly) against performance goals.

One reason why simulation tends to be used more often for analysis of a near-completed design than for early design exploration may be that most simulation tools generally do not provide designers with the means to easily gauge their specific early design options against performance objectives, or with some kind of feedback about how they might change their designs to meet these goals. Due to this issue, designers may not want to take the risk of wasting time exploring options that do not improve performance. Furthermore, creating models and running simulations can become prohibitively time-consuming if generating a comprehensive understanding of the performance requires too extensive an analysis. This combination of unguided search with time-intensive simulations may make the whole process of integrating daylighting considerations early on too tedious and inefficient for designers.

One method for improving efficiency in design exploration is the use of an intelligent guided design approach. As will be discussed in Chapter 2, optimization algorithms are a common solution to this problem, largely because they have the capabilities necessary to find or generate high performing design solutions. However, these methods generally do not allow for user interaction. As it is highly unlikely for a designer to simply accept a design generated by an optimization algorithm, a better approach would be a more interactive search method, which would accept input from a designer and grant the designer a larger degree of control.

An example of such an approach is a knowledge-based or expert system. An expert system is one in which human expert knowledge about a specific domain is encoded in an algorithm or computer system (Luger, 2004). Such systems will be discussed in further detail in Chapter 2. In the daylighting domain, such a system would function as a virtual lighting consultant, guiding the designer towards design modifications which improve overall daylighting performance. Knowledge-based systems have already been successfully implemented for artificial lighting scenarios (Jung et al., 2003). For daylighting, a few simple expert systems exist. The Leso-DIAL tool provides users with a "qualitative diagnosis" using an expert system based on fuzzy logic rules (Paule and Scartezzini, 1997). The Ecotect Shading Design Wizard is capable of automatically generating an overhang shape which will shade a window for a user-specified period of time (Marsh, 2008a). These systems represent first steps in expert systems for daylighting in design, but they do not allow for a comprehensive understanding of daylighting or an investigation of user-defined performance goals.

This thesis will describe a user-interactive expert system approach which enables such a comprehensive analysis of daylighting. This system will be described in detail in Chapters 4 and 5. The proposed approach includes two climate-based performance metrics, one for illuminance and one for daylighting-specific glare, in order for the designer to have an understanding of both the amount of light and the visual comfort in the space. This method uses a designer's own initial design and performance goals, enabling a search process that is highly specific to the user's design problem. The method will evaluate the performance of the design and create a series of suggestions for design changes which are likely to result in improved performance. Decisions are made using an expert system which is comprised of a pre-calculated database of daylighting-specific information connected to a set of fuzzy daylighting expert rules. Validation of the decision making logic will be described in Chapter 7. Any design decision that the designer chooses to allow will be automatically generated in the original model and the new performance will be calculated. The designer is able to interact with the system during an iterative search process that is intended to be agreeable to the designer and likely to improve the performance of the design.

User interaction in the proposed system is enabled through the use of a highly interactive interface which allows multiple types of input and provides the user with a variety of modes for visualizing both the design and the performance data. This interface has been integrated into the the Lightsolve program, which was developed to meet these criteria and to fully integrate the expert system method within the overall user process (Andersen et al., 2008). The interface and its integration into Lightsolve will be discussed in Chapter 6.

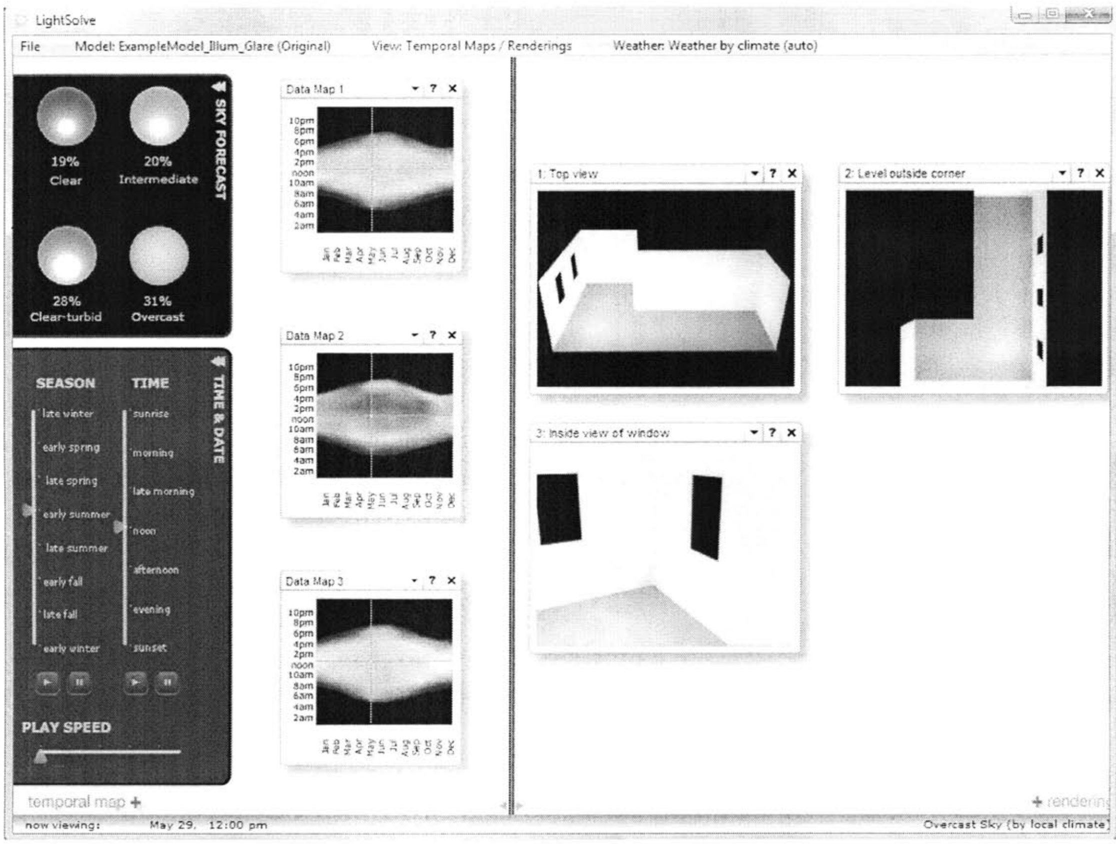


Figure 1.1: The Lightsolve program interface, which displays renderings of views (right side) and performance data in the form of temporal maps (left side).

Lightsolve uses Google SketchUp (Google, 2008), an intuitive modeling software which allows for easy geometrical modeling and material specifications for the initial design input. Goals and constraints are input in a simple and clear interface within SketchUp. An efficient simulation and rendering engine is used within Lightsolve to calculate performance and to create renderings of views. The Lightsolve interface (Figure 1.1) provides a clear visualization of renderings and performance data, and the expert system interface has been embedded within the general Lightsolve program to allow users to interact with the expert system and visualize the aesthetics and performance of their design as it changes during design search process. An analysis of the user interactivity of the expert system process will be given in Chapter 8.

1.4 Thesis Outline

This thesis will describe the development and validation of the expert system for use as a performance-driven search method and as a design tool. Part I consists of an introduction (the present chapter) and literature review. Chapter 2 will present current methods for generating designs based on performance criteria, including optimization methods and expert systems. Chapter 3 will describe the architectural design process and the known methods for introducing performance-based information into the design process.

Part II will focus on the methods used for the development of the expert system. Chapter 4 will describe the population of a database of daylighting-specific information using the Design of Experiments method. Chapter 5 will include an introduction to fuzzy logic and describe the fuzzy logic expert system variables and rules used to make design decisions within the system. Chapter 6 will focus on implementation of the system within the Lightsolve framework and will include descriptions of the expert system interface and the methods used to automate design changes.

Part III will describe the expert system as a search method and as a design tool. Chapter 7 will evaluate the performance and behavior of the expert system decision-making logic based on a series of case studies. The main set of case studies will compare of the performance of designs generated by the expert system to those generated by a genetic algorithm, a popular optimization method. Chapter 8 will discuss the expert system as a user-interactive design tool based on the results of an experiment which studied the use of the expert system during the architectural design process.

Chapter 9 will conclude the thesis with a summary of findings and a description of future work.

Chapter 2

Digital Methods for Performance-Based Building Design

2.1 Introduction

This chapter presents a literature review and state of the art in two different types of digital methods used to improve the performance of architectural design or building systems: optimization methods and expert systems. Although this thesis focuses on the development of an expert system, it is useful to provide a review of optimization methods used within the building design context, as these algorithms have historically been more widely implemented than expert systems. This chapter also summarizes examples of previous expert systems used for building design.

In addition to this chapter, further information about expert systems and genetic algorithms may be found in Chapters 5 and 7, respectively.

2.2 Optimization Methods

A traditional optimization scheme, such as one used in mathematical programming, is an algorithm which finds the minima or the maxima of a given function, typically known as the objective function. The objective function may depend on any number of parameters, and these parameters may either be unconstrained (any value is legal) or may be subject to one or more constraints. The set of constraints will define the search space of the problem, and any combination of parameter values within the defined search space is considered a feasible solution. The optimal solution will be the feasible set of parameters which minimizes (or maximizes) the objective function. A problem will not necessarily have one unique solution. It may have no optimal solutions at all, a finite number of solutions, or an infinite number of solutions, which can be defined as a more specific subset of the search space. A more detailed explanation of optimization techniques can be found in (Papalambros and Wilde, 2000).

The two major categories of optimization algorithms are gradient-based and heuristic search algorithms. A gradient-based optimization algorithm is a "true" optimization algorithm in that its use guarantees that the final solution found will be an optimal one. Examples of gradient-based optimization algorithms are Newton's method, steepest descent, conjugate gradient, and sequential quadratic programming (SQP). These algorithms use information about the gradient of the objective function, either directly or as an approximation. These algorithms are most appropriate when one has a closed-form equation or set of equations at hand.

For problems which involve simulation engines or other scenarios in which such a set of closed-form equations would be difficult or impossible to obtain, heuristic search algorithms are a more appropriate choice. Heuristic search algorithms include genetic algorithms, simulated annealing, particle swarm, ant colonies, tabu search, and many other algorithms which do not require gradient information. In general, such algorithms are based on concepts found in nature (for example, genetic algorithms are based on principles of evolution). These algorithms are considered heuristic because they are not necessarily based on strict mathematical theory and as such, they are not guaranteed to converge to true optimal solutions. However, most of these algorithms do find solutions which are very close to optimal, and they have the additional benefits of being able to effectively search difficult solutions spaces that gradient-based algorithms may struggle with, such as large, multi-modal, and/or non-continuous spaces.

Within each of these two major groups of optimization methods, there are many different individual algorithms, and one generally chooses an algorithm based on the type of function one wishes to optimize and based on certain characteristics of the search space. For example, one may select an algorithm based on whether the problem is linear or non-linear, whether the search space is continuous or discrete (for example, integers only), and so on. For problems involving architectural design or building systems, heuristic algorithms are a popular choice as they are more suitable to the large, multi-modal solution spaces and simulation engines that frequently appear in such problems.

2.2.1 Optimization Methods for Building Performance and Design

In architectural design, optimization may be used in many domains and may include a wide variety of parameters. Performance domains include energy consumption, lighting (both daylighting and artificial lighting), heat transfer, natural ventilation, mechanical systems, materials, structure, and safety (such as evacuation). One might also consider various architectural design domains, such as the floor plan layout or the overall aesthetic of the building. The number of possible parameters can range from a single parameter (for ex. the angle of a blind) to a comprehensive set of parameters which defines an entire building, including geometry, materials, occupant schedule, mechanical systems, lighting and equipment specifications, and general program. Constraints for architectural problems may be based on codes and regulations, programmatic necessities, design intent, client requests, budget, and so on. Constraints may also be based on external characteristics such as weather patterns, site conditions, loads and other structural concerns, or available construction technologies.

While many optimization algorithms exist, they can generally all be classified as either gradient-based or heuristic. Gradient-based methods are often used for problems which have a small, well-defined, convex solution space. In building performance domains, these methods can typically only be used for smaller problems. For example, Park et al. optimized visual comfort by varying a single variable (louver angle) in a double facade system using sequential quadratic programming (SQP), a gradient-based method (2003). However, most problems related to building performance are more complex, requiring the use of many variables and encompassing large, non-linear, and non-convex design spaces. Due to this complexity, many researchers have chosen to use heuristic search methods for building performance optimization.

The genetic algorithm (GA) (Goldberg, 1989) is one of the more commonly used heuristic search techniques and has been applied to many types of architectural problems. A GA is one in which a set of initial feasible solutions is chosen or generated at random. Each is evaluated and those solutions that do not result in good performance are discarded. The remaining feasible solutions are used as "seeds" 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 poor performers are again discarded while the good performers are used as seed values. The cycle is continued until a suitable solution or set of solutions is found. GAs have been successfully used in numerous simulation domains, including single domains such as thermal (Coley and Schukat, 2002; Chen et al., 2008) and daylighting (Wright and Mourshed, 2009; Chutarat, 2001) as well as in multiple domains at once, for example CFD and thermal (Malkawi et al., 2005), natural and mechanical ventilation (Lee, 2007), lighting and thermal (Caldas and Norford, 2002; Caldas, 2008), energy consumption and initial cost (Znouda et al., 2007; Adamski, 2007), and life cycle cost and environmental impact (Wang et al., 2005; Geyer, 2008). GAs have also been used in design domains such as floorplan layout (Michalek et al., 2002; Brintrup et al., 2006).

In addition to genetic algorithms, numerous other heuristic algorithms have been applied towards architectural optimization problems. Examples include shape annealing (Shea et al., 2005), pattern search (Wetter and Polak, 2005), co-evolution (Xiyu et al., 2005), and algorithms based on neural networks (Yeh, 2006), cellular automata (Herr and Kvan, 2007), and cellular analogies (Fischer et al., 2005). The GenOpt system allows users to choose from a wide variety of different optimization algorithms which can be coupled with many different simulation programs, such as EnergyPlus, TRNSYS, SPARK, IDA-ICE, DOE-2, or any user-written program (Wetter, 2001). Additionally, hybrid methods may be used to achieve the advantages of both heuristic and gradient-based methods. For example, Monks et al. used a heuristic method (simulated annealing or SA) as an initial global search and then followed up with a gradient-based method (steepest descent) as a local search for an acoustical design optimization problem (2000). Michalek et al. used a similar approach for architectural layout optimization, combining two methods, one based on genetic algorithms and another which is a hybrid SA/SQP algorithm (2002). Many other possibilities may exist in addition to these strategies which have yet to be studied.

Despite the numerous previous studies in performance-based optimization, most have not considered a goal-driven or user-interactive approach. For example, only a few studies (Caldas and Norford, 2002; Monks et al., 2000) propose tools which allow the user to input specific performance goals for their designs. Likewise, few studies have addressed the issue of user-interactivity or design intent. One of the major roles for an architect in the design and construction process is the architectural design itself, and it is unlikely that an architect would choose a computer-generated design as a final solution, regardless of its optimized performance. Interaction with the tool may increase the chances of the architect actually considering the design as a potential option. Some studies have attempted to address this issue by producing multiple final designs from which the user can choose (Coley and Schukat, 2002; Yeh, 2006; Znouda et al., 2007). While this method will provide the designer with several options instead of one, it does not allow him to truly interact with the system, and there is always the possibility that the designer will not accept any of the options presented to him. Others have implemented interfaces which allow the user to interact with the tool while it is still processing (Monks et al., 2000; Michalek et al., 2002; Malkawi et al., 2005). These systems may allow the user to shift the progress of the optimization process by modifying goals or constraints (for example, Monks et al., 2000), or they may simply allow the user to visualize new designs as they are generated (for example, Malkawi et al., 2005). Such systems represent first steps towards the integration of optimization tools into the design process.

An additional constraint when using optimization algorithms for design purposes is that they can often be considered "black box" algorithms. A "black box" is a system in which the user deals only with the inputs and outputs, while the inner algorithm or logic is "opaque" and hidden from the user. Designers who use such a system to optimize a performance-based design problem will receive an output (an optimized design or set of designs) with little to no explanation of the underlying principles used to generate such outputs.

2.2.2 Examples of Lighting Performance Optimization

Several studies have demonstrated the potential for optimization algorithms to facilitate facade design exploration for daylighting and related performance areas. Several researchers have considered photovoltaic-integrated facade systems and examined the trade-off between facade area used for daylighting and that used for electricity generation (Vartiainen et al., 2000; Charron and Athienitis, 2006). Park et al. considered double facade systems with integrated blinds and found optimal blind angles for several visual comfort metrics (2004). Shea et al. optimized the effect of the glazing type of roof panels on lighting performance and cost (2005). Several studies have optimized window size and placement while considering both daylighting and energy (Caldas and Norford, 2002; Wright and Mourshed, 2009). In the GENE_ARCH system, lighting and energy are optimized in a generative system which can also incorporate an architect's specific aesthetic design intent (Caldas, 2008). Other studies have considered daylighting performance from a visual comfort standpoint. For example, Chutarat's system allowed multiple objectives within the daylighting domain such as illuminance, glare, and direct sunlight (2001), and Torres and Sakamoto's study found facade solutions resulting in high illuminance and minimal glare due to daylighting (2007).

In addition to the studies described above, there also exist related problems in computer graphics which may prove relevant to daylighting optimization. The inverse lighting problem describes a situation in which a user wishes to illuminate an image or a 3d model with certain effects. For example, in this type of problem, the user would use his mouse to highlight the area in the scene which he would like to illuminate (Schoeneman et al., 1993). The problem is considered "inverse" because the user is actually providing the end result, while an algorithm must determine the surrounding conditions which would create such a result in the image or model. Because the user inputs a desired illumination into the program, this type of problem can be considered both user-interactive and goal-driven. Like performance-based optimization, there are multiple ways of solving the inverse lighting problem. Optimization schemes which have been used to attempt to solve this type of problem include least squares (Schoeneman et al., 1993; Do and Gross, 2004), Newton's method for simple unconstrained minimization (Kawai et al., 1993), and genetic algorithms (Tena and Goldberg, 1997). Similar inverse problems which have been studied include a user-guided design of a reflector (Patow et al., 2007) and an automatic light system for rendering 3d models (Shacked and Lischinski, 2001).

2.3 Expert Systems

One potential limitation to most optimization schemes is that the algorithms generally have no relationship to the problem or system that they are trying to optimize. In contrast, a knowledge-based or expert system is a system in which human expert knowledge about the design domain is encoded in an algorithm or computer system (Luger, 2004). While optimization methods originated in the field of mathematics, expert systems are usually considered to be an application of artificial intelligence. Due to these origins, a traditional optimization algorithm is often generic enough to be potentially applicable to a broad range of problems, while an expert system is created for one specific problem. Such systems are typically designed to emulate a human expert and are used in situations where such an expert is needed but not available due to cost, risk, or scarcity.

Expert systems and optimization methods may both potentially be used to improve or optimize a building's design. However, expert systems differ from optimization in several key ways. The most obvious is that expert systems assume that there is a user involved and cannot perform without a certain amount of user input. In general, the user must provide the answers to a series of questions about the problem before the expert system can operate. This user involvement is different from an optimization algorithm, which often performs most efficiently when there is no user input during the optimization process. Another difference between the two methods is that the creation of an expert system requires a great deal of domain-specific knowledge. This means that a human expert must be employed to provide knowledge to populate the expert system and help with decision making.

Knowledge-based or expert systems typically consist of a knowledge base, which contains domain-specific knowledge, and an inference engine, which applies the knowledge to a user-specified problem in order to determine a solution. Rule-based systems are a common type of knowledge-based system. In these systems, the knowledge base is represented in the form of if-then rules. Due to this structure, one final benefit of expert systems is an explanation capability. Because any action or decision made by an expert system is based on logical reasoning using a set of domain-specific rules, an expert system can explain why it chose to act in a certain way based on an initial set of user input. This feature potentially makes expert systems valuable as educational tools. Optimization algorithms, particularly heuristic search methods which function mostly as "black boxes," do not possess this quality.

2.3.1 Expert Systems for Building Performance and Design

There is great potential for expert systems to be used for building design decisions, particularly as the nature of architectural practice is one in which expert consultants are frequently used in specialized domains. Expert systems have been developed for a variety of problems related to architectural and building systems design. These systems have been particularly popular in the areas of mechanical and structural systems engineering.

Early developments of expert systems in the domain of energy and thermal comfort were primarily rule-based and focused on the design of building envelopes and mechanical systems. For example, IDABES was proposed as an expert system used to develop schemes for mechanical systems based on preliminary design goals and constraints (Doheny and Monaghan, 1987). BEADS was developed to make decisions about the building envelope based on a building's climate zone and ASHRAE standards (Fazio et al., 1989). Shaviv et al. developed a system that designs the floorplan and building envelope for passive solar buildings (1996). The SETIS project, a collaboration between a research laboratory and a civil engineering firm, was designed to support decision making about the building envelope and HVAC system design based on the building's climate and specific design goals (Robin et al., 1993). The EKSPRO system was aimed to be a complete system which selected layouts, materials, and mechanical equipment to minimize energy costs due to heating and lighting (Pau and Nielsen, 1990).

In addition to areas of design, rule-based expert systems in the building energy and thermal comfort domain have also been developed for diagnostics and system controls. Several researchers have developed systems to diagnose moisture, dampness, or condensation problems (Matsumoto and Toyoda, 1994; Allwood et al., 1988; Sachdeva, 1985). Du and Jin's system detects and diagnoses faults in VAV systems (2007). Doukas et al. proposed an integrated energy management system which considers occupant preferences and sensor inputs in addition to a building energy knowledge database (2007).

Expert systems have been used frequently in the domain of structural engineering. A common problem in this domain is that of structural frame design, and several types of expert systems have been developed for various framing problems. Example of these problems include preliminary design of tall buildings (Sabouni and Al-Mourad, 1997; Maher, 1987), large spans (Golabchi, 2008), lattice dome design (Lin and Albermani, 2001), and liquid-retaining structural design (Chau and Albermani, 2005). In addition to structural frame design, expert systems have been used for the design of individual structural members, such as cold-formed steel column cross-sectional shapes (Liu et al., 2004). They have also been proposed as a method to help structural engineers develop structural solutions based on early architectural designs (Mora et al., 2006).

Expert systems have been created for more general building design, and development of such tools seems to have peaked in the late 1980s to mid 1990s. Architectural code compliance checking was a popular early expert system type which was developed by many researchers (Oey and Passchier, 1988; Rosenman and Gero, 1985; Dym et al., 1988; Mitusch, 1989; Chun and Lai, 1997). Generative design systems were also popular during this time, and they were based on a variety of different rules sets. Such systems have been proposed based on residential design heuristics (Garijo and de Garrido, 1988), Christopher Alexander's "pattern language" rules for design generation (Gullichsen and Chang, 1985), and general building typology (Lutton, 1995). A more recent study by Mashood et al. has combined an expert system with a genetic algorithm to optimize the layout planning of multi-story buildings (Mashood et al., 2007).

2.3.2 Examples of Expert Systems for Lighting

In the lighting domain, a small number of expert systems have been developed. Light Pen, a system used for artificial lighting design and luminaire selection, allows the user to indicate where he or she would like light to fall in a space and automatically selects one or more options for lighting schemes which may achieve the desired effect (Jung et al., 2003). iPlot is a similar system used to aid theatrical lighting designers select the number and type of luminaires necessary to realize their lighting concept goals (Perelson, 2005). RetroLite is a system designed to aid designers with lighting retrofit problems for energy conservation (Guo et al., 1993).

Expert systems for daylighting also exist, although these systems have remained limited in scope and in capabilities. Shaviv et al.'s system considered shading as a performance

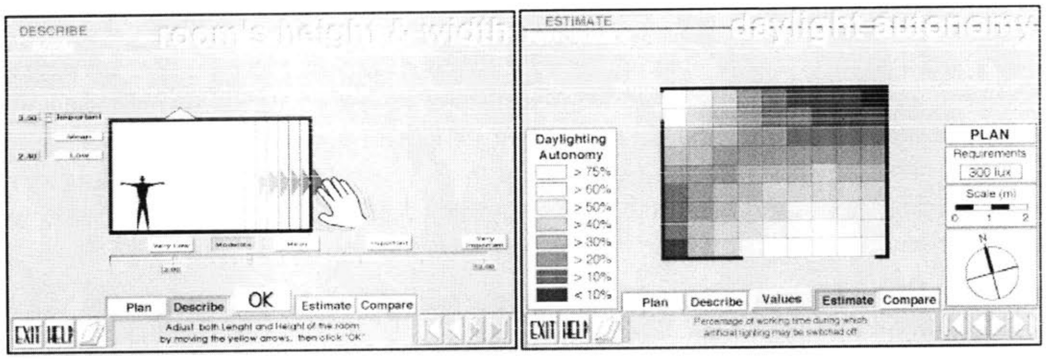


Figure 2.1: Example interfaces from the Leso-DIAL tool (Paule and Scartezzini, 1997)

area in their passive solar expert system (1996). The EKSPRO system considered the daylight factor when determining energy due to electric lighting (Pau and Nielsen, 1990). The NewFacades approach considers energy and visual comfort based on a prescription energy code for hot climates to suggest a range of facade solutions to the designer (Ochoa and Capeluto, 2009).

The Leso-DIAL and DIAL-Europe tools represent the previous work which is the most relevant to the expert system approach described in this thesis. The Leso-DIAL tool uses daylighting performance data with a fuzzy logic rule set to determine if the lighting in a space is satisfactory and to suggest changes to the basic design if it is not (Paule and Scartezzini, 1997). DIAL-Europe, an expansion of Leso-DIAL, includes artificial lighting and overheating modules in addition to daylighting intelligence (de Groot et al., 2003). The DIAL tools thus offer a first attempt to provide design "diagnostics" based on daylighting performance data. However, these tools only consider performance based on the diffuse portion of daylight (skylight), and they impose specific geometry restrictions in terms of model geometries. They do not allow the designer to view renderings of their design and they offer only a limited set of possible design diagnoses.

2.4 Self-Guided Search

While this chapter has focused on formal search and decision-making algorithms, it is worth noting that many designers may use their own self-guided search processes as they work towards improving performance in the early design stages. A common way of searching is by using parametric studies, which allow designers to understand how performance may change when they vary one or more design parameters. Because they are generally completely self-guided, parametric studies may be used with virtually any type of simulation or analysis tool. While they may not enable designers to find optimal solutions, these types of studies provide a simple and easy way to search the solution space and find better performing designs.

Because parametric studies are popular, some tools have begun to incorporate features which allow designers to more easily compare the performances of different designs within the tools themselves. The Decision Desktop is a visualization tool which is connected to two different simulation and evaluation programs and allows for side-by side comparison of the energy use and code compliance of three different designs (Reichard and Papamichael, 2005). The MIT Design Advisor allows users to compare up to four different scenarios at once based on energy, thermal comfort, cost, and daylighting availability metrics (Urban and Glicksman, 2007). The Daylighting Dashboard has been proposed to allow a user to compare different designs as well as variants of the same space (blinds open or closed, passive vs active occupants) for a variety of metrics which consider daylight availability, comfort, and energy use (Reinhart and Wienold, 2011). While these tools do not provide optimization or expert system support, they are valuable in that they do offer a more structured way for designers to consider how different design parameters may influence performance.

2.5 Chapter Summary

This chapter has presented introductions to optimization algorithms and expert systems and has reviewed the states of the art of both of these algorithms for use in domains related to building design. Chapter 3 will present an overview of the architectural design process and the ways in which computer algorithms such as optimization or expert systems may be integrated into it.

Chapter 3

The Architectural Design Process

3.1 Introduction

This chapter presents a review of the traditional architectural design process, a description of the ways in which daylighting considerations are included during design, and a brief outline of trends in digital tools which have affected the design process. The chapter concludes with a summary of key features that support the user-interactive performance-driven daylighting design process described in this thesis.

3.2 The Traditional Design Process

This section includes a review of the traditional architectural design process, including an examination of the nature of design in comparison to that of optimization methods, an outline of the various stages of the design process, and a list of design heuristics used by designers.

3.2.1 The Nature of Design

The architectural design process is inherently similar to a traditional optimization or search process. During the design process, the architect will come up with an initial design or set of designs, based on a set of constraints and goals specific to the current architectural problem. He (or she) will go through an iterative process during which he will refine and develop his design scheme. He will stop when he reaches a final solution which satisfies both the constraints of the project and its goals.

Unlike optimization problems, however, the architectural design process cannot be defined by a formal algorithm. Because it is a process which may vary by individual designer, one can only determine certain typical characteristics of the process based on observation of the process itself and interviews with designers. There have been several large studies in which the architectural design process has been researched (Lawson, 2006; Schon, 1983; Akin, 1986). These studies have shown that the design process and the optimization process differ in several critical ways. For example, Akin has categorized the various search methods used during the design process that are are analogous to optimization algorithms (1986). However, Akin notes that the architect, unlike an optimization algorithm, can never fully explore the entire search space. There are an infinite number of solutions to any design problem, so the architect must always limit his or her search to a small subset of possibilities. While an optimization algorithm will find the best possible solution to the objective function, an architect will attempt to find a solution which merely satisfies all constraints. This mechanism is called "satisficing." A satisficing solution is essentially a feasible solution which is arrived at based on the initial designs the architect chose to develop. Had the architect chosen different early designs, the satisficing solution could thus also be different. Hence, we note that the criteria for the solution of a design problem tend to be less strict than those of an optimization problem, since the final result may be one of an infinite number of possible satisficing solutions.

In addition to the differences in solution criteria, Akin also points out that the entire problem of design is different from that of optimization. Traditional optimization problems are considered "well-defined" because all areas of the problem, including the objective function, parameters, and constraints, are defined by mathematical functions. The goal of the problem is clear: one wishes to minimize or maximize the given objective function within the given search space. In contrast, an architectural design problem is considered "ill-defined" because the problem is almost always too complex to fully define. It is near impossible to define a clear set of specific objective functions because in nearly any given design problem, an architect may wish to meet dozens or even hundreds of goals, many of which may be subjective (for example, achieving a certain aesthetic or ambiance).

The design process may also be influenced by many additional external factors which would never affect a purely mathematical optimization problem. For example, the client may desire a certain aesthetic at first and later shift towards a different aesthetic, or an architect may experience a moment of inspiration mid-way through the design process and completely redefine his or her design scheme. These types of factors will be reflected in both the architect's design process and in the final design. Due to the ill-defined and changing nature of the design problem, it is not possible to fully predict the design process.

3.2.2 Stages of the Design Process

Despite the unpredictable nature of design, the process can still be categorized into a series of general stages, each of which may inform our understanding of the total process. Lawson (2006) has developed a model which divides the process loosely into the following steps, where each category refers to one or more skills typically used during the design process:

1. Formulating - This stage includes the architect's initial analysis and understanding of the problem, including identification of key elements and framing the problem within a larger context.

2. Representing - This stage refers to ways architects choose to communicate their design ideas (whether as drawings, models, or in digital form). It also refers to the designer's skill identifying ideas from representations and in understanding a single design even if it is represented in multiple ways.

3. Moving - This stage involves what many consider "creativity" or "inspiration" in design. During the "Moving" stage, an architect generates design ideas based on the initial problem statement and other relevant information. "Moving" also refers to the development or interpretation of previous designs to inform the current one.

4. Bringing Problems and Solutions Together - In this stage, the architect attempts to create design solutions which will solve the problem at hand. This skill set is not necessarily a design stage but is actually present throughout the design process. The designer is capable of handling problems which occur throughout the design process, not just at the beginning, and he can also choose to solve separate problems on parallel lines of thought before converging to a single solution.

5. Evaluating - This stage refers to both objective and subjective analysis of a potential solution based on the design problem. The architect must not only be able to successfully evaluate his own work, but he also must know when these evaluations will be most fruitful.

6. Reflecting - This final stage refers to the point during the design process in which an architect must determine if his design is actually progressing in the correct way. If necessary, he may have to reformulate his design strategy. This stage also includes a designer's use of his own guiding principles as well as reference material from the architectural community to judge a design.

Although an architect may generally follow these stages in the order presented, it is important to keep in mind that during the design process the architect may in fact continually move back and forth between multiple stages until the point is reached when he or she is satisfied with the final design scheme.

3.2.3 Design Heuristics

We can further our understanding of the design process by examining a set of heuristics used by architects for decision making during the design process. Akin has compiled a comprehensive list of these heuristics based on observation of architects at work (1986). Some heuristics of note include:

- Decision hierarchy - Early decisions may act as constraints in later decisions.
- Most constrained first - Work on the most restrictive or constrained issues first.
- Solution testing - Partial solutions developed based on early constraints must be tested later in the design process if constraints have been modified or added.
- Search in uncertainty - If a piece of information is unavailable, make an assumption and consider it correct until the information become available.
- Sequential processing - If alternative solutions are available, choose one and explore it fully before exploring the others.
- Constraint hierarchy - If some constraints are inclusive of others, satisfy the general constraints before moving on to the more specific ones.
- Least promising first - Work on solutions with potential problems first so that if problems do arise, they will arise quickly and one can immediately discard the solution and move on.

These heuristics can be considered the design process analogy to the various algorithms which can be chosen for a given optimization problem. Although not as well-defined as a formal algorithm, they can provide a logic describing decision-making during the design process.

3.3 Incorporating Daylighting Goals Into the Design Process

Incorporating daylighting performance as a goal into the design process is a difficult task due to the complexity of the design process itself and due to the dynamic nature of daylight. It becomes even more difficult when one considers the many other design constraints that an architect must meet, such as client preferences or initial costs. In *Sunlighting as Formgiver for Architecture*, Lam lists five design objectives which apply to general design and which can be extended to daylighting design (1986):

1. Providing user comfort and delight in the interior environment.
2. Satisfying programmatic needs of users.
3. Minimizing the building energy cost.
4. Optimizing the public architectural image.
5. Minimizing the initial building construction cost.

Lam warns that a balance among all five objectives must be maintained in order to achieve a healthy design, and that those who focus only on daylighting for minimizing building energy costs and who ignore visual comfort and satisfying programmatic needs will end up with an ugly "Solar Shoebox" (Lam, 1986). Although different projects will have different daylighting goals, Lam writes that designs should be judged by the quality, not the quantity, of light. He indicates two major concepts that the designer should consider when assessing a daylighting scheme: visibility and perception. Visibility refers to whether the amount of light is appropriate to the occupant's tasks. Perception refers to the amount of "visual noise", including brightness or contrast, glare, and gloom, and its effect (positive or negative) on the occupant's comfort

Because it is difficult to achieve a successful daylighting scheme, there are numerous types of methods and tools used by architects during the design process to help them create better daylighting schemes. Two major sources are the IESNA's *Lighting Handbook* (Rea, 2000) and the CIBSE's *Code for Lighting* (CIBSE, 2002), which provide useful information such as lists of recommended illuminance ranges for a variety of building and space functions. Heuristics have evolved based on experience or simplified calculations of known metrics. These rules of thumb act as quick checks for a designer in early design stages and may help guide a design towards those forms most likely to produce good daylighting (Reinhart, 2005; Reinhart and LoVerso, 2010). For the designer who requires more comprehensive information, design guides may be used. These guides generally include heuristics along with numerous other tips, including explanation of metrics used to assess performance, design strategies to achieve certain effects, situations to avoid, and case study buildings (O'Connor et al., 1997; Canada, 2002; CHPS, 2006).

In recent years, simulation tools have also gained popularity, as these tools have the capability to predict year-round performance and to create photo-realistic images (Sarawgi, 2006; Reinhart and Fitz, 2006). However, designers continue to use heuristics and experience more often during the early design stages than simulation tools (Galasiu and Reinhart, 2008). It is during these stages that major decisions regarding the building site, room layouts, envelope, and shading should be made, and an architect who would like to incorporate daylighting into his design should make decisions about these elements with lighting already in mind (O'Connor et al., 1997).

This thesis proposes a new method of incorporating daylighting into the early design stages: a generative and performance-based design process guided by an expert system. The next section describes the concepts of performance-based and generative design.

3.4 Generative and Performance-Based Design

Over the past few decades, digital tools have become increasingly widespread in architectural practice and education. Some claim that digital tools are changing the design process itself (Chastain et al., 2002) and that the use of these tools over traditional design methods (drawings and physical models) will allow architects to design in ways which were not previously considered (Marx, 2000).

3.4.1 Generative Design

An extreme example of digital design is that of generative design, where forms are produced based on an algorithm or logic, sometimes with almost no input from the designer. Several studies have attempted to mitigate between the human design process and the generative design process by enabling user interaction in a generative design tool (Chase, 2002; Herr and Kvan, 2007; Talbott, 2007). Chase evaluated a number of generative tools for use by designers and concluded that generative design tools can be used as teaching aids for beginning designers under the following conditions (2005):

- The tool teaches specific design concepts and allows the designer to quickly explore a number of design possibilities.
- The tool uses clear and fast representation.
- The tool uses a classical mode of process, i.e. one that can be understood by a designer and not one that is not easily explained.
- The tool can be used in conjunction with manual methods.

3.4.2 Performance-Based Design

Another idea for the digital design process is that of performance-based design. In this paradigm, "performance" is considered "the desirability of the confluence between form and function in a given context" (Kalay, 1999). Design itself is considered an iterative process of exploration, in which forms are continuously updated and evaluated for performance. An example approach, the Fenestration IDeA (Intelligent Design Assistant), is described as a hypothetical expert system scenario in which a variety of performance areas are considered based on different fenestration choices (Kalay, 1999). In such a system, the IDeA would give advice to the designer on design changes to make if the design were under-performing. In this paradigm, the computer does not generate the design but acts as a "partner" with the designer during the design process. The performance-based design paradigm has been proposed as a way of structuring the design process to enable designers to make better informed decisions in the early design stages (Petersen and Svendsen, 2010). The expert system described in this thesis also takes such an approach, by providing designers with useful feedback about potential design decisions that aim to improve performance and by allowing the designers themselves to choose which design changes to try.

Since the early 1990's, expert systems have been used as a possible support system for a performance-based approach, and many researchers have sought to understand how they could be best incorporated into the design process. Carrara and Novembri proposed KAAD (Knowledge-based Assistant for Architectural Design) as a method that features several key characteristics which allow for successful integration into the design process: an extremely flexible definition of the current design, which can easily be modified at various stages during the process; an ability for the system to integrate user-provided inputs with known or assumed information; and the capability to modify values of various design attributes (1990). Other researchers have focused on effective communication of information, particularly information about the design of the building, as a critical feature of any knowledge-based approach (Rutherford and Maver, 1994; Pohl and Myers, 1994). In his proposal for a general framework for integrated building performance feedback in early design stages, Augenbroe states that an Intelligent Integrated Building Design System (IIBDS) should have two major components: a set of design support tools under complete control of the designer and a system in which the tools are embedded (1992). Petersen and Svendsen propose that an expert system should present to the designer a set of possible design changes and their predicted effect on performance as a means of supporting early design decisions and reducing the number of design iterations necessary to find a good design (2010).

3.5 Features of the Proposed Expert System Design Process

The daylighting expert system proposed in this thesis has been developed to be compatible with the traditional architectural design process based on the principles of performance-based design. The system is intended for early stage design exploration and decision support using daylighting performance criteria. Based our knowledge of the design process and conclusions from the previous work of researchers who have explored performance-based and generative approaches, the proposed system features the following characteristics:

1. Intuitive and robust implementation:

    - The system has been implemented in Google SketchUp (Google, 2008), a popular 3d modeling environment that many designers and consultants already use.
    - The system features an intuitive, interactive interface which allows the designer to view his 3d model, renderings of the interior, and performance metrics all on the same screen (described further in Chapter 6).

2. Ability for the designer to customize his design problem:

    - The designer can start the process with an original 3d model of his or her own design (he is not forced to conform to highly specific geometry rules or accept a default model).

- The designer determines the number, type, and specific details of his or her daylighting goals.
- The designer selects the climate and applicable times of day and year for the system to evaluate.

3. Clear communication of ideas:

   - The system indicates the performance of the design using easy-to-read temporal maps and graphs.
   - The system proposes design changes using regular language (for example, "Make South windows larger").
   - The designer can examine 3d views of his design from multiple angles at all times during the process.

4. Designer control of the process:

   - The designer is allowed to accept or deny any proposed design change.
   - The designer can stop the process at any time if he or she is satisfied with a final design.
   - The designer can stop the process at any time to manually change in his or her 3d model before continuing with the process.

5. Efficient, performance-driven exploration of the design space:

   - The system automatically generates new designs and computes their performance, which allows the designer to explore the design space more efficiently than if he or she had to make each model manually.
   - The system uses an expert system to suggest design changes most likely to improve performance, allowing the designer to search more effectively.

The expert system described in this thesis is not intended to replace traditional design methods, but rather to supplement them by providing additional information to the designer during the design process. The system should help designers discover better performing solutions than they otherwise would during the early design stages. It may also inform designers about how different design elements affect the overall daylighting performance. Such information is valuable because even designers who do not accept the designs generated by the expert system may ultimately incorporate elements which improve daylighting performance into revised designs of their own. Designers may also receive an educational benefit by using the tool and gain intuition about working with daylighting. These phenomena will be demonstrated in the discussion of the user studies in Chapter 8.

3.6 Chapter Summary

This chapter reviewed the nature of architectural design and the various stages that architects go through during the design process. This chapter also discussed the integration of daylighting goals into the design process and described the performance-based design paradigm. The chapter concluded with a summary of critical features of the expert system described in this thesis and the ways in which these features help to integrate such a system into the design process and support a performance-driven approach.

Chapter 4 will describe the first major component of the expert system, a daylighting-specific database which acts as a core source of daylighting knowledge in the expert system algorithm.

Part II

An Expert System for Daylighting Design

Chapter 4

A Daylighting-Specific Database

4.1 Introduction

Many of the expert systems mentioned in Chapter 2 are traditional systems in the sense that they are populated using knowledge from a human expert, and as a result, such systems are restricted in terms of accuracy and complexity. To create an expert system capable of more sophisticated analyses, this thesis uses a daylighting-specific database, or "knowledge base," which has been populated using data from a set of completed daylighting simulations.

This database contains information about the relative effects of various facade conditions on the mean illuminance levels on a workplane and on the probability of glare for views in various directions within a space. Such effects are available for design conditions on facades oriented towards each of the four cardinal directions, for five different zones (four perimeter zones and one core zone), for three different periods of day (morning, midday, and afternoon), and for three different seasons of year (summer, autumn/spring, and winter). For the glare metric, we consider views in each of the four cardinal directions from locations within each of the five zones described.

By using calculated data rather than heuristics to populate the knowledge base, the expert system can consider highly specific goals and multiple sets of goals for the same design, which can differ based on the daily time period(s), season(s), or zone(s) of interest within a space. This type of knowledge base also allows for more logical and accurate comparisons of multiple design options than mere heuristics.

This chapter describes the methodology used to populate the daylighting-specific knowledge database. It includes examples of database results for Boston, Massachusetts (USA) and a brief comparison of database results from two different climates.

4.2 Knowledge Acquisition

The core of an expert system is the knowledge base, a database in which the main domain-related information is encoded. Traditionally, the knowledge acquisition process used to populate a knowledge base involves two people (or groups of people): an expert in the domain and a knowledge engineer. The human expert provides information to the knowledge engineer, who then encodes it as data in the knowledge base.

The most common way of storing such information is in a rule-base, a series of "if-then" statements or a decision tree that roughly reproduces the logic that the human expert would use to solve a problem. However, in daylighting, such knowledge is likely to be limited to only general scenarios. For example, a human expert could predict that increasing the area of a window or raising the window-head-height will increase the illuminance in a space due to daylight. However, it would be difficult or impossible for a human expert to predict whether a change to the aspect-ratio or a horizontal translation of a window would increase glare from a certain viewpoint and at a specific time of day and year. A human expert would also have difficulty quantifying the effects of specific design changes, and such difficulty would be compounded when issues of climate are considered.

For this thesis, a daylighting-specific database was created using a set of completed simulation results. Populating the knowledge base using this approach enables a quantitative comparison of various design changes to the facade and their effects on daylighting performance inside the space. The completed simulations used to create the knowledge base data were selected using the Design of Experiments (DoE) approach (Montgomery, 2004).

DoE is a formal method of experimentation which has previously been used in a variety of building technology applications. For example, DoE has been used to create simple models based on building simulation results in domains such as thermal comfort (Hwang et al., 2009), heating demand prediction (Jaffal et al., 2009), and thermal behavior of adobe walls (Parra-Saldivar and Batty, 2006). The DoE method has also been proposed to support a design methodology for designing low energy buildings (Chlela et al., 2009) and as a means of determining the most significant design parameters to consider when designing a double-skin facade (Seok et al., 2009). The DoE method used for populating the knowledge base developed for this thesis is described in the section 4.2.1.

Three steps were necessary to populate the knowledge base using simulation data. First, a set of models was created based on the facade variables of interest using the DoE methodology. Both the illuminance and glare metrics were then calculated by the simulation engine. Finally, the main effects of each design variable on the chosen daylighting performance metrics were calculated.

In this study, simulations were performed for two locations: Boston, Massachusetts, USA (temperate climate, latitude 42° North) and Siem Reap, Cambodia (tropical climate, latitude 13° North). However, one can obtain data for any other location by simulating the same models with the appropriate weather data and by re-calculating the main effects. In this way, one could build an extended knowledge base which contains information relevant to multiple locations and climates.

4.2.1 Design of Experiments Methodology

Design of Experiments (DoE) is a formal method of experimentation which allows the experimenter to obtain information about how independent factors (variables) affect a given output, relative to each other (Montgomery, 2004). In this thesis, a two-level DoE approach was used. Two-level designs involve the study of two different values, also called levels, of each factor. Results of the DoE method include main effects, two-level interaction effects, and higher level interaction effects.

The knowledge base described in this thesis uses the main effect, which describes the effect of one level of a given variable, averaged across all values of all other variables, on the output. The main effect is calculated as the difference in average response between the two levels of a factor. For a two-level design, the absolute values of the main effects of two different levels of the same factor will be the same; the main effect for one level will be positive and the main effect of the other level will be negative. A positive effect indicates that the output is, on average, increased when the factor is at the specific level tested, while a negative effect indicates that the output is, on average, reduced by the tested value or group of values. The magnitude of the absolute value indicates the magnitude of the effect of that variable on the output. This allows for a useful comparison of all values of all variables on the tested outputs.

Before calculating the main effects, one must first run a set of experiments. These experiment sets may be "full factorial," which require all combinations of factors and levels, or they may be "fractional factorial," which include a smaller number of experiments based on an orthogonal matrix of level combinations. For a given set of variables, there may exist several different fractional factorial schemes, each with a different number of required experiments. Those schemes with a higher number of experiments will have a higher resolution, which refers to the quality of information obtained by the set of experiments, specifically the amount of possible confounding with higher order interaction effects. For example, the main effects of a Resolution III or Resolution IV design may be confounded with two-level or three-level interaction effects, respectively, and all higher-order interaction effects will also be confounded. A Resolution V design results in non-confounded main effects and two-level interaction effects; however, to achieve Resolution V, more experiments must be run.

For the proposed expert system, the chosen study design was a two-level fractional factorial Resolution V design. To achieve this level of resolution, 128 unique models were required, each with a different set of design variables. The levels used for each factor in this DoE scheme are indicated in Figures 4.1 and 4.2. For a given set of 128 models, only one facade orientation at a time was allowed to have windows (the other facades were opaque). By using this design, the database could contain information about how a certain design variable on a facade facing one orientation differed from the same design variables on a different orientation. Details about these design variables will be discussed further in section 4.2.5. To understand the effects of the design variables on each of the four cardinal orientations, a full set of 128 models was created for each orientation. Therefore, a total of 512 simulations were run to populate the complete knowledge base.

The final knowledge base contains the main effect of each level of the ten design variables considered on illuminance and glare. The effects were calculated for facades oriented towards each of the four cardinal directions, for five different zones (four perimeter zones and one core zone), for three different periods of day (morning, mid-day, and afternoon), and for three different seasons of year (summer, autumn/spring, and winter).

4.2.2 Daylighting Simulation Engine

The engine used to simulate the daylighting performance of each DoE model, 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 is the native engine of the Lightsolve program and was chosen because it allows for rapid calculation of the full-year, climate-based daylighting metrics described in section 4.2.3.

To make the whole-year simulation more efficient, the LSV engine divides the year into 56 periods and calculates the illuminance and glare during each time period under four different sky types ranging from overcast to clear using the method described in (Kleindienst et al., 2008). The climate-based metrics are calculated using a weighted sum of the results of each of the four different sky types, weighted by the frequency of each sky condition for a given climate at a given time of year. This metric is a modification of the ASRC-CIE sky model by Perez, which is generally considered to be one of the most accurate sky models available (Perez et al., 1992).

In addition to calculation, the LSV engine also creates color renderings of the interior of spaces. Cutler et al. found that a rendered scene in LSV took approximately 3.3% of the time that it took to complete an analogous "fast rendering" in Radiance (2008). 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).

4.2.3 Illuminance and Glare Daylighting Metrics

The two metrics considered in this thesis were illuminance and glare. These metrics were chosen to enable the system to consider daylighting performance in terms of both visual performance and visual comfort.

Experiment	Window Area	Number of Windows	Window Shape	Vertical Location	Horizontal Location	Distribution	Horizontal Overhang	Vertical Fins	Glass Transmissivity	Glass Type
1	0	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	1	0	0
3	0	0	0	0	0	1	0	1	0	0
4	0	0	0	0	0	1	1	0	1	1
5	0	0	0	0	1	0	0	1	0	1
6	0	0	0	0	1	0	1	0	1	0
7	0	0	0	0	1	1	0	0	1	0
8	0	0	0	0	1	1	1	1	0	1
9	0	0	0	1	0	0	0	1	0	1
10	0	0	0	1	0	0	1	0	1	0
11	0	0	0	1	0	1	0	0	1	0
12	0	0	0	1	0	1	1	1	0	1
13	0	0	0	1	1	0	0	0	1	1
14	0	0	0	1	1	0	1	1	0	0
15	0	0	0	1	1	1	0	1	0	0
16	0	0	0	1	1	1	1	0	1	1
17	0	0	1	0	0	0	0	1	1	0
18	0	0	1	0	0	0	1	0	0	1
19	0	0	1	0	0	1	0	0	0	1
20	0	0	1	0	0	1	1	1	1	0
21	0	0	1	0	1	0	0	0	0	0
22	0	0	1	0	1	0	1	1	1	1
23	0	0	1	0	1	1	0	1	1	1
24	0	0	1	0	1	1	1	0	0	0
25	0	0	1	1	0	0	0	0	0	0
26	0	0	1	1	0	0	1	1	1	1
27	0	0	1	1	0	1	0	1	1	1
28	0	0	1	1	0	1	1	0	0	0
29	0	0	1	1	1	0	0	1	1	0
30	0	0	1	1	1	0	1	0	0	1
31	0	0	1	1	1	1	0	0	0	1
32	0	0	1	1	1	1	1	1	1	0
33	0	1	0	0	0	0	0	1	1	0
34	0	1	0	0	0	0	1	0	0	1
35	0	1	0	0	0	1	0	0	0	1
36	0	1	0	0	0	1	1	1	1	0
37	0	1	0	0	1	0	0	0	0	0
38	0	1	0	0	1	0	1	1	1	1
39	0	1	0	0	1	1	0	1	1	1
40	0	1	0	0	1	1	1	0	0	0
41	0	1	0	1	0	0	0	0	0	0
42	0	1	0	1	0	0	1	1	1	1
43	0	1	0	1	0	1	0	1	1	1
44	0	1	0	1	0	1	1	0	0	0
45	0	1	0	1	1	0	0	1	1	0
46	0	1	0	1	1	0	1	0	0	1
47	0	1	0	1	1	1	0	0	0	1
48	0	1	0	1	1	1	1	1	1	0
49	0	1	1	0	0	0	0	0	1	1
50	0	1	1	0	0	0	1	1	0	0
51	0	1	1	0	0	1	0	1	0	0
52	0	1	1	0	0	1	1	0	1	1
53	0	1	1	0	1	0	0	1	0	1
54	0	1	1	0	1	0	1	0	1	0
55	0	1	1	0	1	1	0	0	1	0
56	0	1	1	0	1	1	1	1	0	1
57	0	1	1	1	0	0	0	1	0	1
58	0	1	1	1	0	0	1	0	1	0
59	0	1	1	1	0	1	0	0	1	0
60	0	1	1	1	0	1	1	1	0	1
61	0	1	1	1	1	0	0	0	1	1
62	0	1	1	1	1	0	1	1	0	0
63	0	1	1	1	1	1	0	1	0	0
64	0	1	1	1	1	1	1	0	1	1

Figure 4.1: Factor levels for the first 64 experiments in the 2-level fractional factorial Resolution V orthogonal array

Experiment	Window Area	Number of Windows	Window Shape	Vertical Location	Horizontal Location	Distribution	Horizontal Overhang	Vertical Fins	Glass Transmissivity	Glass Type
65	1	0	0	0	0	0	0	1	1	1
66	1	0	0	0	0	0	1	0	0	0
67	1	0	0	0	0	1	0	0	0	0
68	1	0	0	0	0	1	1	1	1	1
69	1	0	0	0	1	0	0	0	0	1
70	1	0	0	0	1	0	1	1	1	0
71	1	0	0	0	1	1	0	1	1	0
72	1	0	0	0	1	1	1	0	0	1
73	1	0	0	1	0	0	0	0	0	1
74	1	0	0	1	0	0	1	1	1	0
75	1	0	0	1	0	1	0	1	1	0
76	1	0	0	1	0	1	1	0	0	1
77	1	0	0	1	1	0	0	1	1	1
78	1	0	0	1	1	0	1	0	0	0
79	1	0	0	1	1	1	0	0	0	0
80	1	0	0	1	1	1	1	1	1	1
81	1	0	1	0	0	0	0	0	1	0
82	1	0	1	0	0	0	1	1	0	1
83	1	0	1	0	0	1	0	1	0	1
84	1	0	1	0	0	1	1	0	1	0
85	1	0	1	0	1	0	0	1	0	0
86	1	0	1	0	1	0	1	0	1	1
87	1	0	1	0	1	1	0	0	1	1
88	1	0	1	0	1	1	1	1	0	0
89	1	0	1	1	0	0	0	1	0	0
90	1	0	1	1	0	0	1	0	1	1
91	1	0	1	1	0	1	0	0	1	1
92	1	0	1	1	0	1	1	1	0	0
93	1	0	1	1	1	0	0	0	1	0
94	1	0	1	1	1	0	1	1	0	1
95	1	0	1	1	1	1	0	1	0	1
96	1	0	1	1	1	1	1	0	1	0
97	1	1	0	0	0	0	0	0	1	0
98	1	1	0	0	0	0	1	1	0	1
99	1	1	0	0	0	1	0	1	0	1
100	1	1	0	0	0	1	1	0	1	0
101	1	1	0	0	1	0	0	1	0	0
102	1	1	0	0	1	0	1	0	1	1
103	1	1	0	0	1	1	0	0	1	1
104	1	1	0	0	1	1	1	1	0	0
105	1	1	0	1	0	0	0	1	0	0
106	1	1	0	1	0	0	1	0	1	1
107	1	1	0	1	0	1	0	0	1	1
108	1	1	0	1	0	1	1	1	0	0
109	1	1	0	1	1	0	0	0	1	0
110	1	1	0	1	1	0	1	1	0	1
111	1	1	0	1	1	1	0	1	0	1
112	1	1	0	1	1	1	1	0	1	0
113	1	1	1	0	0	0	0	1	1	1
114	1	1	1	0	0	0	1	0	0	0
115	1	1	1	0	0	1	0	0	0	0
116	1	1	1	0	0	1	1	1	1	1
117	1	1	1	0	1	0	0	0	0	1
118	1	1	1	0	1	0	1	1	1	0
119	1	1	1	0	1	1	0	1	1	0
120	1	1	1	0	1	1	1	0	0	1
121	1	1	1	1	0	0	0	0	0	1
122	1	1	1	1	0	0	1	1	1	0
123	1	1	1	1	0	1	0	1	1	0
124	1	1	1	1	0	1	1	0	0	1
125	1	1	1	1	1	0	0	1	1	1
126	1	1	1	1	1	0	1	0	0	0
127	1	1	1	1	1	1	0	0	0	0
128	1	1	1	1	1	1	1	1	1	1

Figure 4.2: Factor levels for the final 64 experiments in the 2-level fractional factorial Resolution V orthogonal array

Illuminance Metric

The illuminance metric is a climate-based, whole-year metric calculated for each time period as a weighted average of illuminances from each sky type. 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 this thesis (Lee et al., 2009).

For each model in the DoE scheme, the LSV engine was used to calculate the mean climate-based illuminance due to daylight on the five sensor plane zones over the whole year. For our system, results from these 56 periods were averaged into a total of nine general time periods: morning, mid-day, and afternoon for winter, autumn/spring, and summer. The autumn and spring seasons are combined because results from these two seasons in preliminary trials revealed near symmetry. This combination also enabled the size of the knowledge base to be reduced, resulting in greater efficiency of memory and calculation time within the system.

Glare Metric

The glare metric used for this thesis was developed by Kleindienst and Andersen (2009) and is a model-based approximation of Daylighting Glare Probability (DGP). The DGP was originally introduced by Wienold and Christoffersen (2006) and indicates the percent of occupants disturbed by a daylighting glare situation. Like the original DGP and its simplified variant DGPs (Wienold, 2007, 2009), the model-based approximation method (DGPm) is based on the vertical illuminance on each glare sensor plane. While the DGPs metric is solely dependent on vertical illuminance, the DGPm also assumes a user-defined direction of view, and it is dependent on the position of each sensor relative to each glare source as well as the luminance of each of these glare sources. Because this method is specific to daylit situations, windows are currently considered the only glare sources. The model-based approximation method has been found to perform within a 10% error of the DGP over 90% of the time for rectangular models which do not include window frames (Kleindienst and Andersen, 2009). The default model used in this thesis (section 4.2.4) conforms to these modeling limitations.

4.2.4 Default Building Description

The models created for the DoE method all have the same overall massing form, a generic rectangular base model. Each model features a single exterior facade through which daylight can enter the space, based on a set of the 10 design variables described in section 4.2.5. The facade design (windows, shading devices, and glazing type) vary in each model. The base model is a single height space which is 9.1m by 9.1m in area and 3.1m in height (30ft x 30ft x 10ft). The four facades are oriented towards the four cardinal directions. Interior materials are entirely diffuse with reflectances of 80%, 50%, and 20% for the ceiling, walls, and floor, respectively.

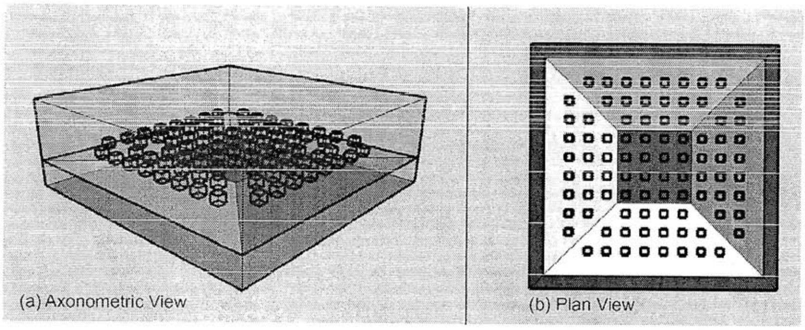


Figure 4.3: (a) Axonometric view and (b) plan view of model with five illuminance sensor plane zones and glare sensors array shown.

Illuminance is measured on a horizontal sensor plane located at a workplane height of 0.9m (3ft) above the floor. The illuminance sensor plane is divided into five zones, one for each perimeter zone and a core zone which is 3.1m by 3.1m (10ft x 10ft) in area (see Figure 4.3). This zone separation was based on the well-known window-head-height heuristic that has recently been validated with annual simulations (Reinhart, 2005): light from a window may penetrate into the space a distance that is up to 1.5 times the window head height for windows with shading devices and up to 2.5 times the window head height for unshaded windows. The zone boundaries occur at depths of 1, 2, and 3 times the window head height for windows located at the maximum height, which roughly correspond to a daylit zone, a drop-off zone, and a deep zone.

The DGPm is measured on vertical sensor planes arranged in arrays of glare sensor "boxes", with each box face oriented towards one of the four cardinal directions, as illustrated in Figure 4.3. Each box face is 0.3m by 0.3m in area (1ft x 1ft). The center of each glare sensor face is located 1.4m (4.5ft) above the floor. The boxes are arrayed in a grid pattern within each zone with a distance of 0.8m (2.5ft) between then. Using the glare boxes, glare is calculated in four different directional views within each of the five zones (20 views in total).

For both metrics, the differentiation between the effect of individual design changes on different zones in the room allows the knowledge base to be more customizable than if information was only obtained in one zone (such as the whole space). Additionally, the four-sided glare sensors enable one to differentiate between different views within the same zone.

4.2.5 Facade Design Variables

Ten different design variables were examined in this study, selected because they are typically early design decisions which may greatly affect daylighting performance. These

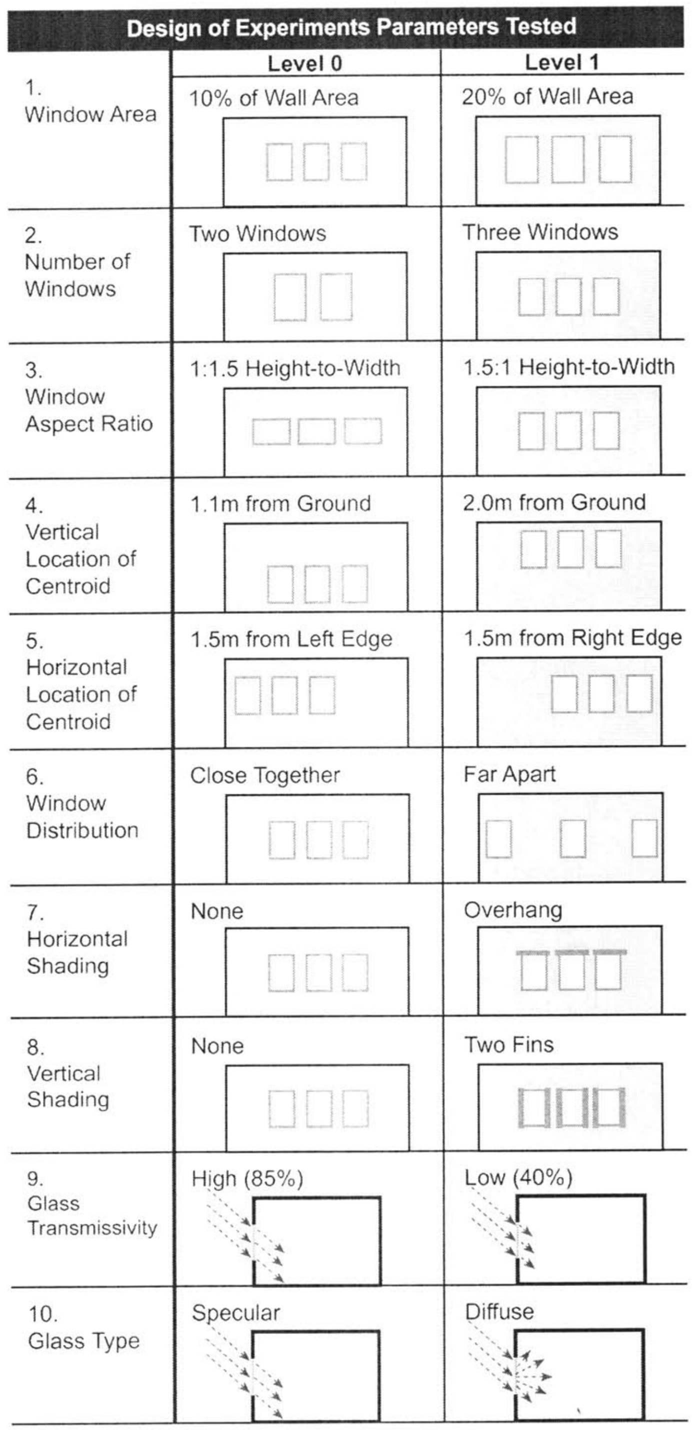


Figure 4.4: Schematic representation of the two levels tested for each design factor in the DoE study

variables include window geometry, location, and distribution on the facade, shading device geometry, and glazing material properties. Two levels of each variable are illustrated in Figure 4.4.

Geometry-based window variables consisted of: total window area, window aspect ratio (narrow or wide), vertical location of windows on the facade, horizontal location on the facade, number of windows, and window distribution on the facade. The "number of windows" variable does not affect the total area; for the larger number of windows, each individual window area is decreased so that the total area remains the same. The window distribution variable refers to the distance between each window. Shading device variables consist of: existence of a horizontal overhang and existence of vertical fins. The horizontal overhangs are dimensioned to block direct sunlight on the equinox at noon in the location considered (in this case, Boston, MA). The vertical fins have the same length as the overhangs.

The material variables considered are glazing transmissivity and type. The glazing transmissivities correspond to clear glass and tinted glass, while the types correspond to specular and diffuse (translucent) transmission.

The values examined for each design factor are also indicated in Figure 4.4: a high value and a low value, as required for a two-level DoE study (see section 4.2.1).

4.3 Design of Experiments Results

The knowledge base holds the main effects results calculated after completing the set of simulations described in section 4.2. The knowledge base consists of two large databases, one for illuminance and one for glare, whose basic structure is represented in Figure 4.5. The illuminance database is a 45x80 matrix containing the main effects for eighty design conditions total (two different values of ten variables on each of four facades) on the mean illuminance specific to three seasons, three periods of day, and five zones. The glare database is a 180x80 matrix containing the main effects of 80 design conditions on the approximated model-based DGP during the three seasons and three periods of day. For each of the five zones, glare information is available for four different views (the cardinal directions), obtained from an array of glare sensors distributed across each zone.

In order to use the information in the knowledge base in a design setting, it is necessary to interpret the main effects data by considering both the magnitude and sign for each effect. The meaning of the sign of a main effect is intuitive and indicates whether a result (either illuminance or glare) will go up (positive) or down (negative). For example, highly positive main effects in the illuminance database indicate that the corresponding design condition will result in a higher mean illuminance on the relevant work plane relative to other design conditions, while highly negative values indicate that the corresponding design condition will result in a lower mean illuminance on the work plane relative to other design conditions. In general, unless they have highly specific aesthetic lighting

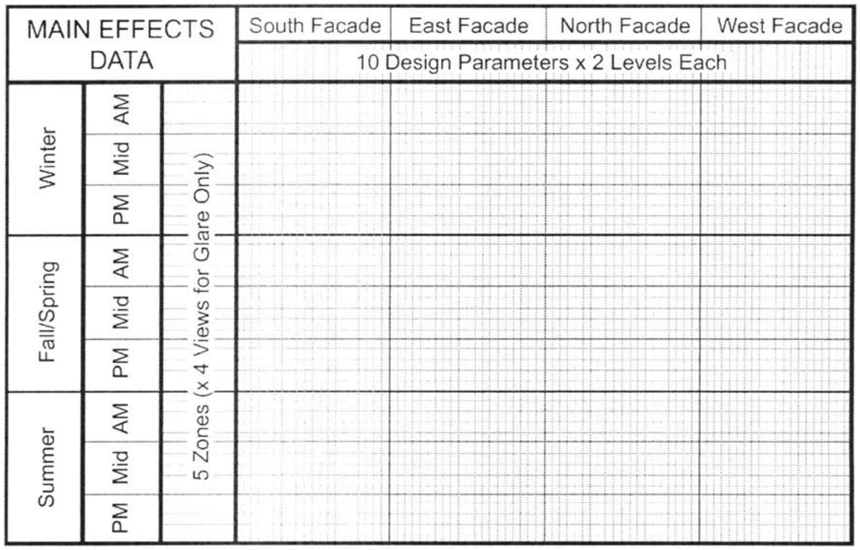


Figure 4.5: Schematic diagram of main effects databases

effects already in mind, most designers will be interested in facade designs that have a positive value for illuminance and a negative value for glare, as these elements will result in higher illuminance with lower glare.

When comparing two main effects of the same sign, the design change with the higher magnitude is more likely to result in a higher mean illuminance if both are positive or a lower mean illuminance if both are negative. For example, if the main effects for both "large window area" and "high transmissivity glass" are positive but the value for "large window area" is higher, then one is likely to see a greater increase in illuminance in the space if one increases the window area than if one increases the glass transmissivity. Likewise, highly positive main effects in the glare database indicate that the corresponding design condition will result in a higher glare probability from the relevant viewpoint relative to other design conditions, and so on.

The actual value of each main effect refers to the difference you can expect between a situation with one particular parameter value and the average of all situations. For example, if the effect of large windows on the south facade on a south perimeter zone at a certain time of day and year is 500, one can predict the zone will receive 500 more lux with large rather than small windows. However, these values are only truly relevant for building designs which have the same general characteristics as the test model described in section 4.2.4. For use within the expert system, it is the relative difference between main effects for different design parameters that should be considered. The relative difference between design parameters should indicate whether one design change is more likely to have an effect on the building performance than another and in what direction, but it will not allow us to predict what that actual magnitude of the change will be.

It is important to note that main effects are calculated over all combinations of design variables in the DoE set. This characteristic is powerful because one can apply the information within it to a wide variety of designs. One can consider the experiments as a collection of case study designs and the knowledge base itself as an aggregate of those case studies. In this way, the knowledge base was designed to be customizable to specific design situations. This means that one can pull out the information most relevant to a specific design scenario while ignoring the rest of the data. For example, if the design were a school room with a single south-facing facade, one would consider only south-facade information during the school schedule (morning and mid-day, autumn through spring). Specific zones and views can also be identified for further customization.

This customization is one important way in which the knowledge base results are used within the expert system. Within the system, a customized database of information most relevant to the user's design is created. This database is formed using the combination of views, zones, facades, times of day, and seasons most relevant to the user's input and initial building design.

The second important way that the knowledge base is used in the expert system is to help create a ranked list of design changes to be proposed by the system before a design iteration. The suggested design changes are determined by first comparing the performance of the current design to the user's goals, then by using the customized knowledge base within a fuzzy logic rule set to select and rank those design changes that are most likely to affect the performance in a positive way. The logic and algorithms used to create the customized database and to select design changes are described further in Chapter 5.

The next sections contain example results from the illuminance and glare databases.

4.4 Illuminance-Based Analysis

The illuminance database contains information about how changing the selected design parameters will affect the mean illuminance on sensor plane in each of the five zones, for the three different times of day and three different seasons. As mentioned in section 4.3, the data can be customized based on specific scenarios. In this section, a selection of example customizations are shown based on the main effects database created for Boston, MA. These examples demonstrate the variety of information that is available within the database.

Three examples are shown with different sets of zones, times, and seasons. They range from general to highly specific. In each example, the main effects for all four different facades are shown on the same chart. Positive main effects indicate that the given design condition is likely to increase the illuminance in the zone(s) considered, and negative main effects indicate that the given design condition is likely to decrease the illuminance. The magnitude of the main effect indicates the magnitude of the expected illuminance effect.

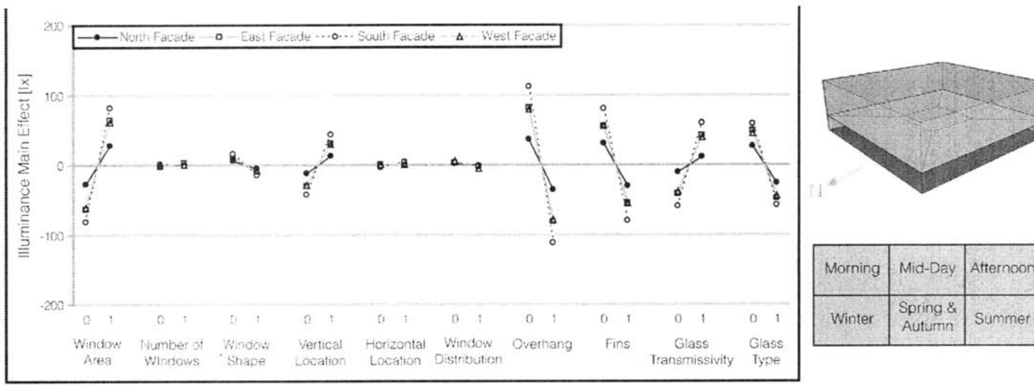


Figure 4.6: Illuminance main effects averaged over all zones, all times, and all seasons

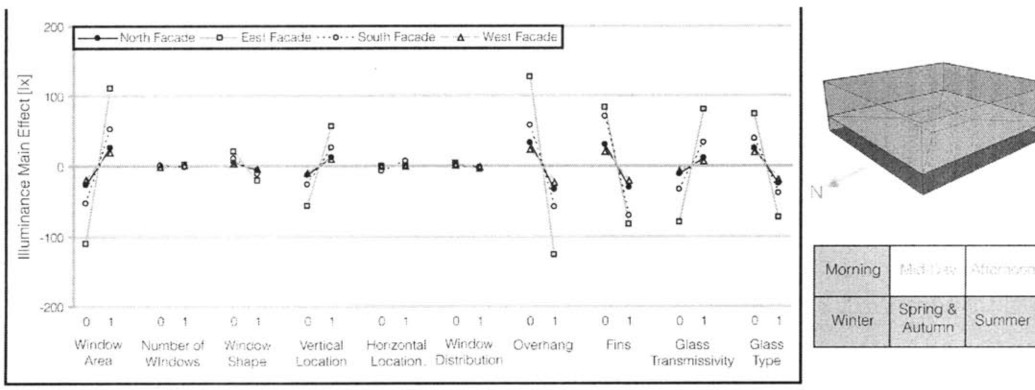


Figure 4.7: Illuminance main effects averaged for AM times, averaged over all zones and all seasons

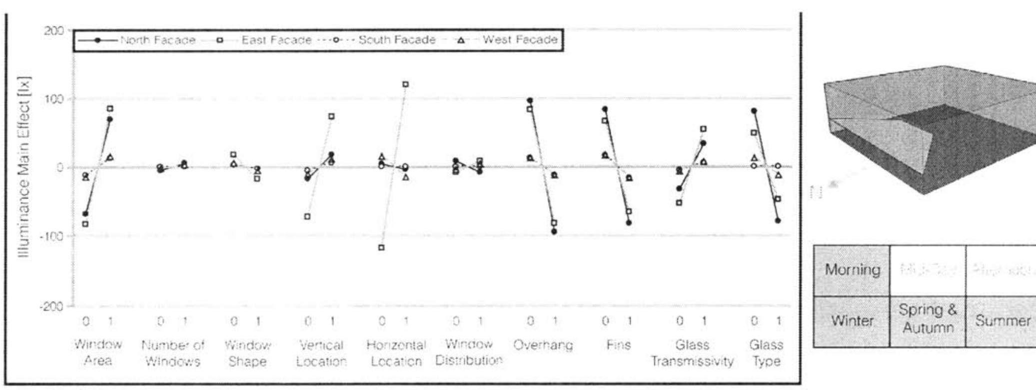


Figure 4.8: Illuminance main effects for north zone and AM times, averaged over all seasons

The first example shows the average main effects for all zones, all times, and all seasons. The second example shows the average main effects for all zones and seasons, but only during morning times. The third example shows the main effects for only the north zone during morning times, averaged across all seasons.

Example 1: Main effects averaged over all conditions

This example, shown in Figure 4.6, represents the most general application of the illuminance database, as it indicates the main effects of the various design parameters when averaged over all possible cases: all five zones, three times of day, and three seasons. From Figure 4.6, it is clear that in general, the design parameters likely to have the largest overall effect on illuminance are the window area, shading devices, and glass type and transmissivity. Geometric properties of the windows such as shape, distribution, and location on the facade have almost no effect on illuminance.

This example also demonstrates the differences in main effects among windows on each of the four cardinal facades. From the chart, we see that for a given level of a given design parameter, the main effects for each facade orientation have the same sign (positive or negative). However, the magnitude of the main effects vary based on orientation. In general, windows on the south facade have the greatest effect on illuminance, while north-facing windows have the least effect. The east and west orientations have roughly the same magnitude of effects in all cases. These results are expected given that this database is for a location in the Northern Hemisphere and that the effects from the whole year were averaged.

Example 2: Main effects for morning only

This example, shown in Figure 4.7, represents a more specific customization of the database which considers only information about the morning, averaged over all zones and all seasons. In this example, we see somewhat different results than the whole-year scenario in Example 1. The design parameters which result in the highest magnitude of effects are the same as the previous example, but we note a large difference between the effects for the different facade orientations. In this scenario, windows on the east facade have a much larger magnitude of effect than in the previous case, where the south-facing windows had the largest effects. These results seem reasonable since this case considers a morning-only scenario, which means that the only direct sunlight considered will be coming from the east.

Example 3: Main effects for north zone, morning only

This example, shown in Figure 4.8, demonstrates a highly specific customization of the database results. In this scenario, only the north zone is considered for morning times, averaged over all seasons. Here we find that for the five major design parameters (based on window area, shading devices, and glazing type), both the north and east facade orientations have large effects on illuminance while the west- and south-facing windows have relatively little effect. These results are expected, given that the north windows are most likely to affect the north zone and the east windows will have a large effect during the morning.

This example is dramatically different from the previous two examples in that the vertical and horizontal location of windows, two design factors which had little effect on illuminance in the more general cases, are shown to have a very large effect in this more specific example. These effects indicate that illuminance in the north zone will increase dramatically when the east-facing windows are shifted horizontally on the facade towards north, or when the east-facing windows are higher on the facade. Both of these design actions allow direct morning sunlight from the east or south-east to penetrate further into the north zone.

4.5 Glare Risk Evaluation

The glare database contains information about how changing the selected design parameters will affect the mean DGPm for views towards each of the four cardinal direction in each of the five zones during the three different times of day and three different seasons. Similarly to the illuminance database, the glare database can be customized so as to be relevant to a variety of situations ranging from general to highly specific. In this section, three examples are shown based on the main effects database created for Boston, MA.

When using the glare database, it is important to recall that a positive main effect indicates that the given design condition is likely to increase the probability of glare seen based on the view(s) considered, and a negative main effect indicates that the given design condition is likely to decrease the probability of glare. As before, the magnitude of the main effect indicates the magnitude of the expected effect on glare probability.

Three examples are shown, each for views towards the south. As with the illuminance examples, the glare database examples range from general to more specific. The first example indicates the average main effects for south-facing views averaged over all zones, times, and seasons. The second example shows the average main effects for south-facing views during the summer months, averaged over all zones and all times of day. The third example shows the main effects for south-facing views from the west zone during evenings in the summer months.

Example 1: Main effects for south views averaged over all conditions

This example, shown in Figure 4.9, demonstrates the most general customization of the glare database. It indicates the main effects for south-facing views averaged over all zones, seasons, and times of day. We note that for all design parameters, the north-facing windows had no effect on glare, which is reasonable given that these results are for views towards south. Likewise, some of the largest main effects are seen for windows on the south facade. We note that the glass type on the south, west, and east facing windows has a large effect on glare probability; for all three orientations, the use of translucent glass is likely to reduce the probability of glare. Lowering the transmissivity on the south facing windows and adding fins to the east and west windows are also shown to decrease glare

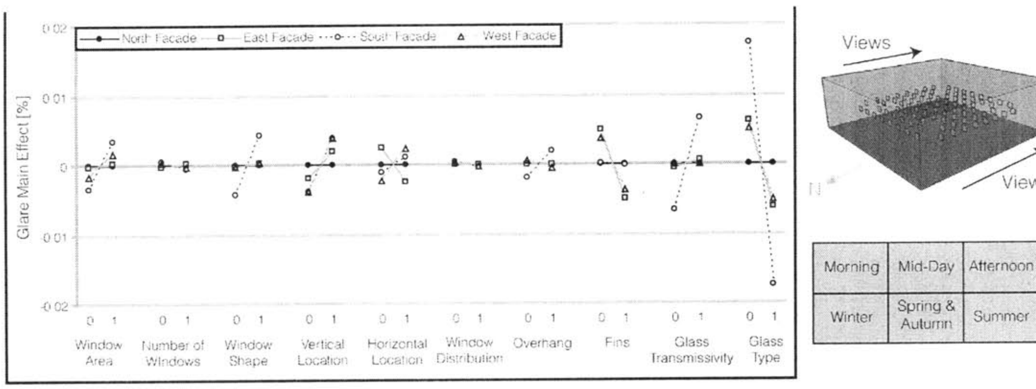


Figure 4.9: Glare main effects for south views averaged over all zones, times, and seasons

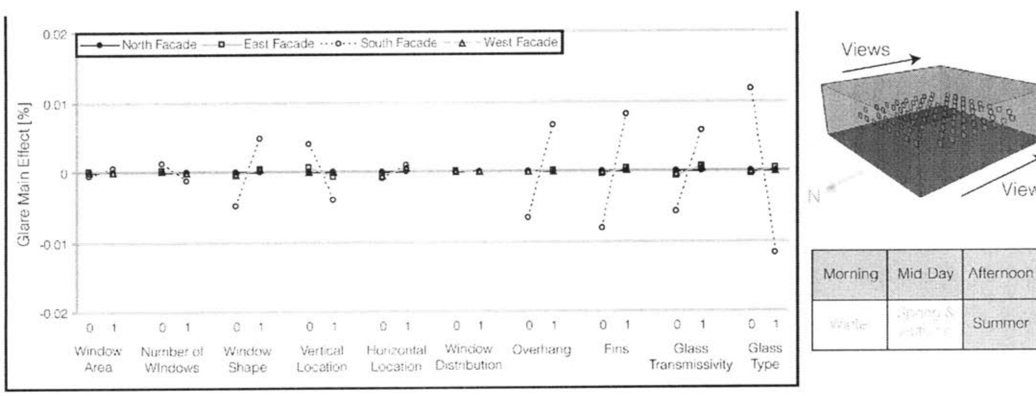


Figure 4.10: Glare main effects for south views and summer season, averaged over all zones and times

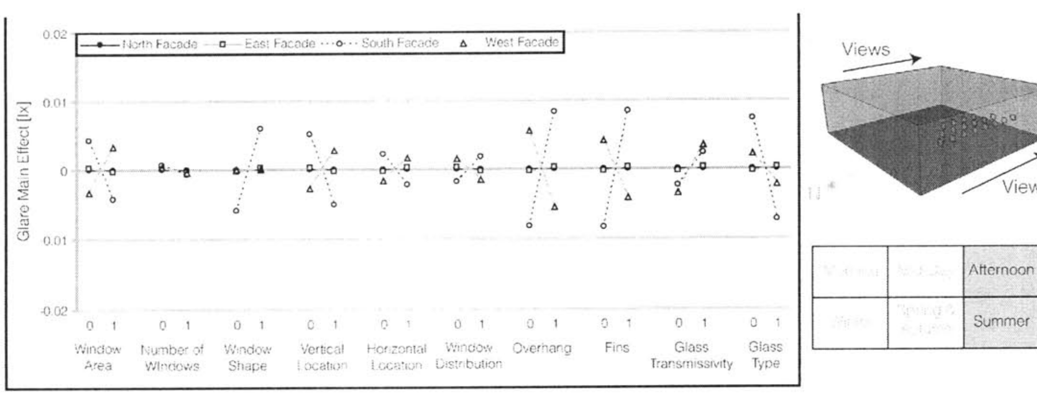


Figure 4.11: Glare main effects for south views in the west zone, summer evenings only

probability. Other design parameters which may have an effect on glare are window area, window shape, vertical location, and horizontal location.

Example 2: Main effects for south views in the summer

In this example, shown in Figure 4.10, south views during the summer months were considered, averaged over all zones and all times of day. This example is striking because the design parameters of the north, east, and west windows have almost no effect on glare in this specific scenario. Instead, only a few parameters on the south facade windows are shown to affect glare. These parameters are window shape (aspect ratio), vertical location, shading devices, and glass transmissivity and type. As expected, lower glass transmissivity and translucent glass will reduce the probability of glare. A less intuitive result seen in this example is that adding shading devices to the south facade is likely to increase glare for a view towards south. This result is likely due to the shading devices reducing the vertical illuminance at the sensor plane while not dramatically decreasing the luminance at the window itself, thus creating a greater amount of contrast in the view direction than the unshaded case.

Example 3: Main effects for south views from the west zone, during summer evenings

The final glare example, shown in Figure 4.11, is highly specific. This example indicates the main effects for glare for views towards south from the west zone, during only evenings in the summer. As expected, the two facade orientations which most effect glare in this case are south and west. However, this example is interesting because for several design parameters, the west and south orientations display opposite effects on glare. For example, adding an overhang or fins to the west facade is likely to reduce the probability of glare, while adding shading devices to the south facade is likely to increase it. Window area and vertical location display a similar pattern. Changes to the glazing transmissivity and type have the same effect for windows on both orientations, as expected.

4.6 Influence of Climate

Because the knowledge base is populated using simulation data based on one specific location, the results should be most relevant to that location, which means that, in order to consider additional locations, new knowledge bases must be created. The previous examples were all created using data from a knowledge base for Boston, MA, USA. In this section, some example results are shown from two different databases: one for Boston, MA, and one for Siem Reap, Cambodia. The daylighting conditions in these locations are different from one another, as Boston has a latitude of 42 degrees N and a continental climate while Siem Reap has a latitude of 13 degrees N and a tropical climate. The two climates also differ based on sky type, which is largely overcast in Cambodia during the monsoon season (from May through October), while it is generally intermediate or turbid during those months in Boston.

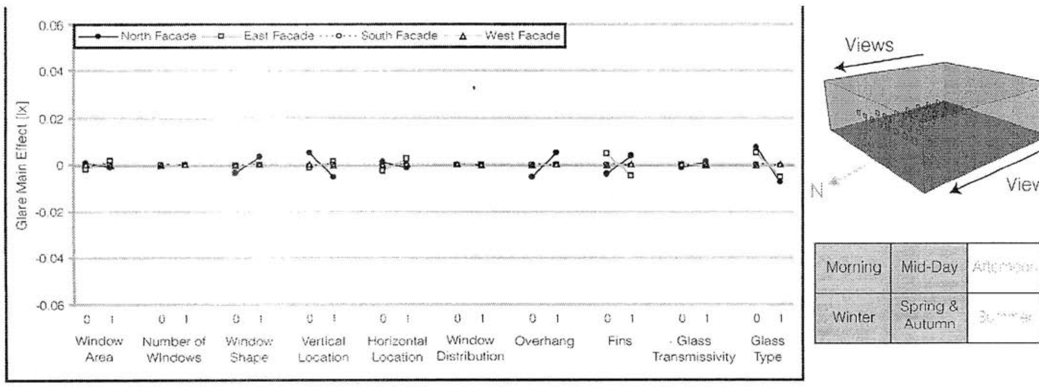


Figure 4.12: Main effects for glare for north views from the north and core zones during a school year schedule in Boston, MA

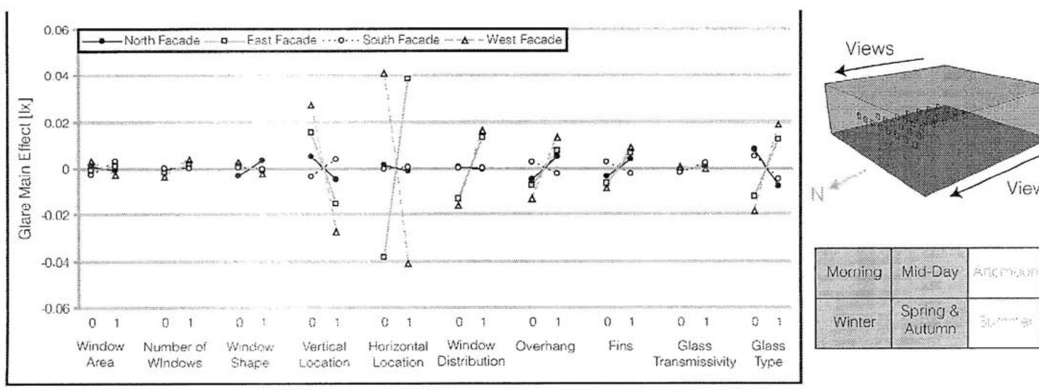


Figure 4.13: Main effects for glare for north views from the north and core zones during a school year schedule in Siem Reap, Cambodia

To illustrate the difference between the knowledge base results for these two locations, an example set of data is shown in Figures 4.12 and 4.13. These figures indicate main effects for glare for north views from the north and core zones during a school year schedule (morning through mid-day, autumn through spring).

One obvious difference between the two charts is that the magnitude of effects is much greater in general for the Cambodia data. This is partially indicative of direct sun from the north, which occurs far more often in Siem Reap than in Boston, due to Cambodia's proximity to the equator. Another large difference is the influence of east and west facing windows on glare in the north direction, which is much larger for Siem Reap than for Boston. This is likely due to the difference in solar altitude between the two locations, which results in a stronger amount of light in Siem Reap during the fall, winter, and spring months. This example shows how different the daylighting conditions may be for the same

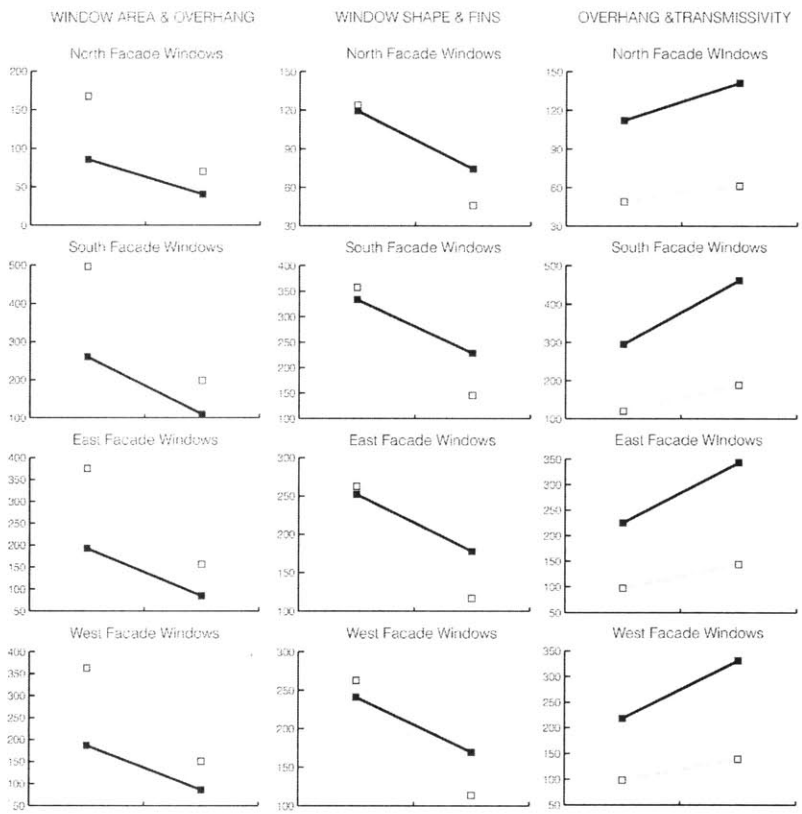


Figure 4.14: Examples of major two-way interaction effects for illuminance, averaged over all zones, seasons, and times of day, in Boston, MA

building if it is sited in a different latitude and climate. This example also demonstrates the need for climate-specific knowledge bases which are accurate for specific locations.

4.7 Interaction Effects

Two-way interaction effects are similar to main effects, except instead of providing an understanding of how a single variable affects illuminance or glare, they indicate how the interaction between two different design parameters affects performance. Although interaction effects are not included in the current version of the daylighting knowledge base, the DoE scheme used in this thesis has a sufficiently high resolution such that they could be included in future work. Interaction effects are important because they can indicate to a designer a variety of combinations of design conditions that may have a large effect on daylighting performance. Like the main effects, the interaction effects can be calculated for situations which range from general to specific.

In this section, one example of interaction effects is demonstrated: the two-way interaction effects for illuminance, averaged over all zones, seasons, and times of day, for Boston, MA. Figure 4.14 shows interaction plots for three of the design parameter pairs which have the largest interaction effects for windows on all four facade orientations. The first two interactions, window area/overhangs and window shape/fins, serve as reminders that the sizing of shading devices relative to window geometry can greatly affect daylighting performance by limiting direct sunlight. The third interaction, glass transmissivity and translucency, indicates that care must be taken when choosing a glass type, as using a translucent glazing will have a different effect depending on whether the transmissivity is high or low.

4.8 A DoE-Based Design Guide

Although the daylighting knowledge base has been created specifically for use within the expert system described in this thesis, this database may also be used as a stand-alone resource. This section describes a simulation-based, climate-specific design guide, created using the knowledge-base results. To generate such a guide, one must first determine the specific set of zones, views, seasons, and times of day most relevant to the given project and pull out the corresponding data from the knowledge-base. The main effects and interaction effects should then be sorted according to the desired performance, such as high illuminance or low glare. Those design parameters or pairs of design parameters which are known to result in the desired performance can then be presented to the designer. One can also consider both illuminance and glare at the same time, and sort the design parameters based on both metrics.

An example guide is shown in Figure 4.15. This guide is for Boston, MA and considers views looking towards north from the north and core zones as well as illuminance in the north and core zones for a school schedule (morning to mid-day, autumn to spring).

The example guide has been created to suggest design conditions which result in high illuminance and low glare at the same time. It includes individual design conditions (for example, "wide windows") as well as combinations of design conditions. The design suggestions are grouped by facade orientation so the designer can understand how different design conditions may be appropriate for one facade orientation and not another. The layout of the guide aims to simplify the knowledge-base results so that a designer can quickly identify design characteristics which will help him to achieve the desired performance. The single design elements that are likely to have the desired performance effect are located closest to each facade. Lines connect these elements with others to indicate pairs of design characteristics that, combined, should result in good performance. The data is presented in a clear way, much like traditional design guides, but, as the guide is based on simulation results, it is also applicable to more specific situations than simple heuristics.

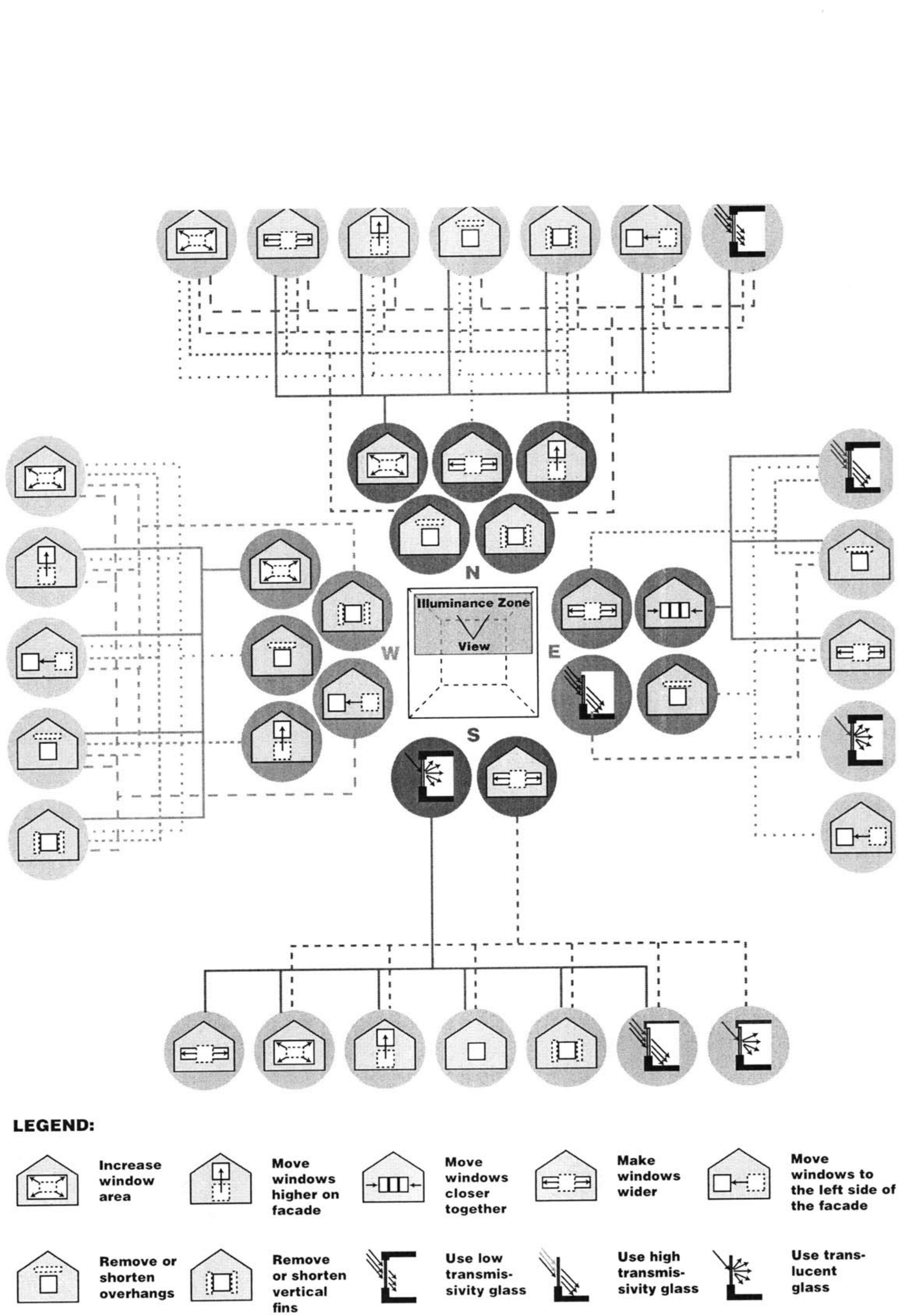


Figure 4.15: Example design guide for Boston, MA with cardinal facade orientations considered (specific for views looking towards north from the north and core zones as well as illuminance in the north and core zones for a school schedule)

4.9 Chapter Summary

This chapter discussed the generation of a daylighting-specific database, or "knowledge base," using the Design of Experiments methodology. This knowledge base will act as the core set of knowledge for use in the expert system.

The daylighting knowledge base contains information about the relative effects of ten different facade parameters on the mean illuminance levels on a workplane and on the probability of glare for views from within the space. The effects are available for design conditions on facades oriented towards each of the four cardinal directions, for five different zones (four perimeter zones and one core zone), three different periods of day (morning, mid-day, and afternoon), and three different seasons of year (summer, autumn/spring, and winter). For the glare metric, views were considered in each of the four cardinal directions from locations within each of the five zones described.

This chapter described a set of examples of illuminance and glare main effects results for Boston, MA. The results indicate how the main effects for these metrics differ based on the level of specificity desired and how the knowledge base can be used for design scenarios which range from general to highly specific. This chapter also illustrated the difference between the main effects data for two different locations, Boston and Siem Reap, Cambodia, which demonstrated the need for climate-specific knowledge bases.

Chapter 5 will discuss the logic by which the knowledge base is incorporated into and utilized by the expert system.

Chapter 5

A Fuzzy Rule-Based Expert System

5.1 Introduction

The goal of an expert system is to emulate the logic and decision making capabilities of a human expert when given a problem in a certain domain. The expert system described in this thesis has been designed to act as a daylighting consultant, one which can guide a designer towards design decisions which will help improve the daylighting performance of a space based on the designer's own goals for illuminance and glare.

The proposed daylighting expert system contains two major components: the daylighting-specific database described in Chapter 4 and a set of fuzzy rules which act as the decision making logic within the system. Fuzzy logic is a type of logic which is used to describe ideas or values which are not as black and white as the concepts used in classical logic. This type of logic is commonly incorporated into expert systems so as to better represent the way that humans (as opposed to computers) think and solve problems.

This chapter describes the expert system as a whole, including the various inputs into the system, the fuzzy logic variables and rule set, and the means by which the daylighting database is incorporated into the process.

5.2 Fuzzy Logic

This section provides a brief overview of fuzzy logic and fuzzy rule-based expert systems.

5.2.1 Introduction to Fuzzy Logic

Fuzzy logic was first developed in the 1930s by philosophers who wished to explore concepts of "vagueness" and extend classical logic from having only two values (0 and 1, or

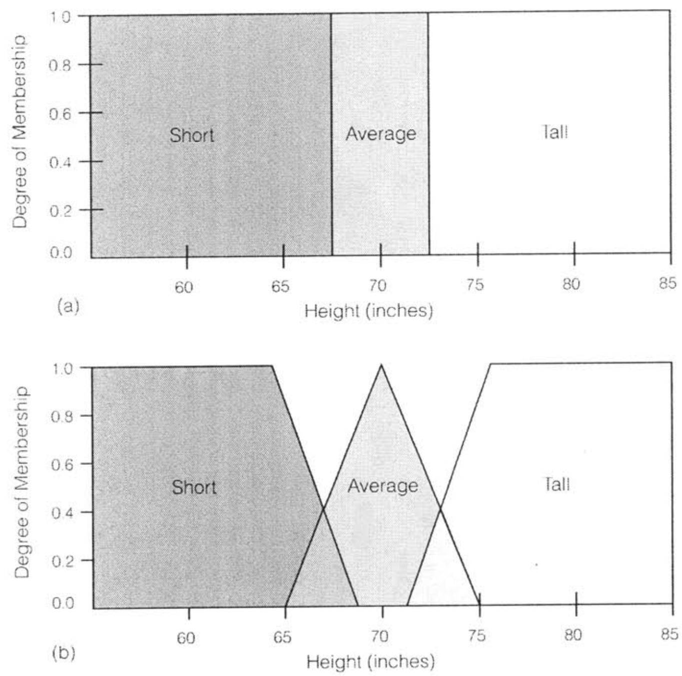


Figure 5.1: Example crisp (a) and fuzzy (b) sets for short, average, and tall men.

"true" and "false") to a continuous set of values between 0 and 1 (Lukasiewicz, 1930; Black, 1937). It was re-discovered in 1965 by Zadeh, who expanded it into a formal system for mathematical logic (1965).

The main idea of fuzzy logic is to represent vague or non-perfect membership within a set. Using classical or "crisp" logic, an element may either belong to a set or not, so the degree of membership in a set can only be either 0 or 1. Using fuzzy logic, an element may belong to multiple sets, and the degree of membership in a set can be any real number between 0 and 1. The use of fuzzy sets thus allows the representation of partial membership in a set and of simultaneous membership in seemingly conflicting sets. Fuzzy logic enable those who use them to think in "shades of gray", as opposed to the "black and white" of classical logic.

A common example of fuzzy logic is that of categorizing men based on their heights (Figure 5.1, adapted from Negnevitsky, 2005). Using crisp logic, a man who is 5′7″ is considered "average" and a man who is 6′2″ is considered "tall." Using fuzzy logic, a man who is 5′7″ has a degree of membership of 0.3 for "short" and 0.4 for "average." A man who is 6′2″ is considered 0.2 "average" and 0.7 "tall." Using the fuzzy sets, one can begin to understand the degree of "short," "average," or "tall" in a more intuitive and human way than using the strict crisp sets. Additionally, because membership is not mutually exclusive, the degrees of membership do not need to sum to 1 as would be necessary using classical logic.

Because fuzzy logic has been developed to be a formal logic system, many of the same operations can be done using fuzzy sets as with crisp sets; for example, complements, intersections, and unions. Both types of sets also have the same properties, such as commutativity, associativity, distributivity, identity, and transitivity. A good overview of fuzzy logic theory can be found in (Gaines, 1976).

5.2.2 Fuzzy Rule-Based Systems

Rule-based expert systems are a common type of expert system in which decision making logic is written as a series of if-then statements. An example rule-based expert system is one which chooses the appropriate accessories for someone to wear or carry based on the weather. Below are two example rules which might exist in such a system:

IF sky is cloudy

AND predicted_weather is rain

THEN necessary_accessory is umbrella

IF sky is clear

AND predicted_weather is sun

THEN necessary_accessory is sunglasses

A fuzzy rule-based system works in a similar way, except that some or all of the variables are fuzzy instead of crisp. In the example above, the weather could be somewhat cloudy and somewhat clear. In general, a fuzzy rule-based system uses the following steps (Negnevitsky, 2005):

1. Fuzzification - In this step, crisp inputs are converted to fuzzy variables based on fuzzy logic similar to that explained in section 5.2.1. For example, a crisp input might be the amount of cloud cover in the sky, and the result of the fuzzification would be the degree of membership in each of the categories "cloudy" and "clear."

2. Rule evaluation - In this step, the fuzzified inputs from step 1 are applied as the antecedents of the fuzzy rules. Just as each of the antecedents is fuzzy in these rules, the rule outputs will also be fuzzy values. For example, in the rule set above, it might be possible for the outputs to be: necessary_accessory is 0.8 umbrella and 0.1 sunglasses.

3. Aggregation of the rule outputs - In this step, the outputs of all the rules are combined into a single fuzzy set.

4. Defuzzification - In this step, the fuzzy results are defuzzified so that the final output is a crisp value. For example, the final output from the example rule set above might be: "Definitely bring an umbrella. You may want to consider bringing sunglasses."

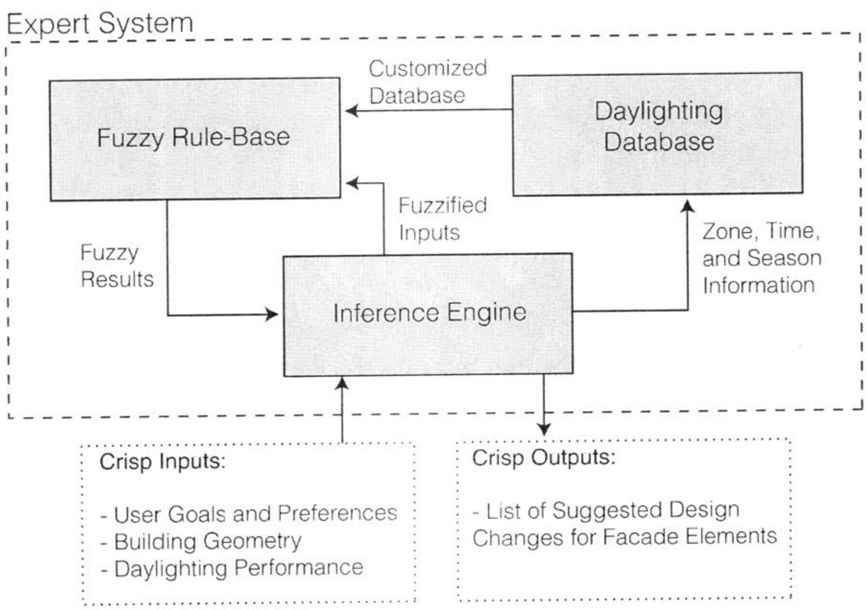


Figure 5.2: Schematic diagram of the expert system structure and information flow

In addition to the set of fuzzy rules, an important component in a fuzzy expert system is an inference engine. The inference engine acts as the "brain" of the expert system by connecting the various components. For example, the inference engine is what determines which rules to fire based on available inputs.

In the system described in this thesis, the inference engine connects the fuzzy rule-base and the daylighting database. Additionally, it handles inputs from the user interface, the 3d modeler, and the simulation engine.

5.3 Expert System Structure

The general structure of the expert system is indicated in Figure 5.2. The general process is described below:

1. The user creates an initial 3d model of a design and specifies performance goals and preferences.
2. Daylighting performance for the current model is calculated using the LSV engine described in section 4.2.2.
3. The inference engine "fuzzifies" the crisp inputs into fuzzy variables. These variables refer to user goals and preferences as well as current daylighting performance for the design.

4. The inference engine creates a customized database using the general daylighting database described in Chapter 4. The customized database contains only the information most relevant to the current design.

5. Fuzzy rules are fired using the customized database and fuzzy variables. The results are a set of suggested design changes that the system will propose to the user in order to improve performance.

6. Results are presented to the user in the user interface.

7. The user selects a design change to make, and new 3d models are created automatically.

8. Go to step 2.

Steps 1-5 will be discussed in further detail in this chapter. Steps 6 and 7, which refer to the user interface and system automation, will be discussed in Chapter 6.

5.4 General Description of Assumptions and Logic

This section provides a brief general overview of some of the major assumptions and logic that the expert system uses to decide which design changes are most likely to improve performance of a given design. Further details of the expert system logic are provided in sections 5.5, 5.6, and 5.7.

Selecting Which Windows to Target

The expert system assumes that the design changes made to the facade closest to a given sensor will affect the sensor more than design changes made to facades further from that sensor. Similarly, the expert system assumes that on a given facade, some windows will be closer to a sensor plane than other windows, and changes to those windows will have a greater effect on the sensor plane than other windows on the same facade. The expert system uses these assumptions to help determine which windows should be changed during a given design iteration.

Dealing with Multiple Performance Goals

If there are multiple sensors within a model, the expert system will attempt to find design changes which are likely to improve the performance of all sensors at once. However, in situations where the user's goals are conflicting, the expert system will choose to improve one sensor at a time, perhaps at the expense of performance of another sensor. In these scenarios, the expert system uses the following logic: goals which the user has specified as high priority take precedence over lower priority goals, and sensor planes which have the lowest current performance have priority over sensor planes with good current performance.

Dealing with Illuminance Goal Ranges

Illuminance goals are based on ranges, which means that the user is allowed to specify both lower and upper bounds. As a result, an illuminance sensor plane may see illuminance that is too low, within range, or too high. Dealing with a sensor plane which sees illuminance that is both too high and too low at the same time is a complicated problem. The expert system chooses to deal with this problem in two ways: it determines if other sensors would benefit more from moving towards higher illuminance or lower, and it takes into account whether the amount of illuminance which is too high is greater than or smaller than the amount of illuminance which is too low.

Determining an Appropriate Magnitude of Change

A problem similar to that of dealing with illuminance goal ranges is selecting an appropriate magnitude of change. For example, the system may wish to increase the illuminance on a sensor plane a small amount so as to reach the user's goal range; however, an increase which is too large will result in decreased performance due to the illuminance being too high on the sensor plane. The expert system deals with this issue by determining whether a change should be "small" or "large," and by selecting design changes from the daylighting knowledge base which are deemed most appropriate for that magnitude. It then creates design changes in three increments and allows the user to select the version he or she prefers. For example, if the design change were to increase the overhang length on a group of windows, the system would test out three different models, each with a different overhang length, and the user would choose one of the three models to accept based on the resultant performance and aesthetics.

5.5 Required Expert System Inputs

This section describes the crisp inputs that the expert system requires. The crisp inputs help the system establish the context of the design problem by providing the system with the information it needs to understand about the initial design, the performance goals, the performance of the initial design based on those goals, and the user's preferences. These inputs include those which are provided directly by the user and those which are calculated by the simulation engine or within the inference engine.

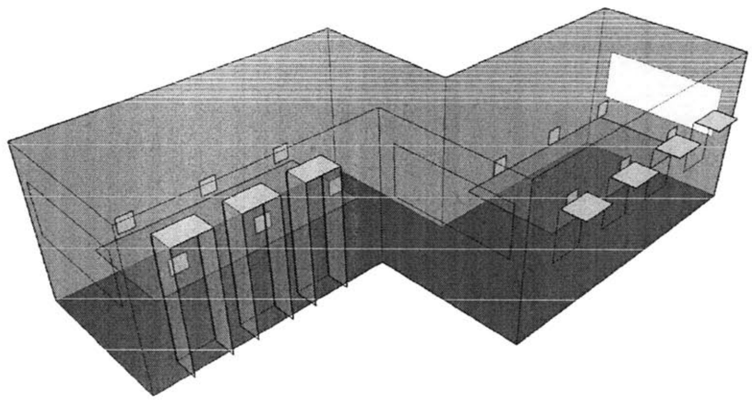


Figure 5.3: Example 3d model which meets expert system modeling criteria. Horizontal illuminance sensor planes are colored red, vertical glare sensor planes are colored yellow.

5.5.1 User Inputs

Before the expert system process begins, a large number of user inputs are required from the user. The major user input is a 3d model, which must be created in Google SketchUp. Simple pop-up interfaces within Google SketchUp allow the user to input the remaining information, including performance goals, priorities, and other preferences.

3d Model

One innovation within the expert system is to allow users to specify their design by creating a 3d massing model in Google SketchUp instead of requiring them to define their base model using a text-based approach or choose from a set of default options. This user-defined massing model should indicate the general form of the space and all desired opaque material properties, i.e. wall, floor, and ceiling reflectances, as well as glazing types and shading device geometry.

Within the 3d model, the user must also specify 2d sensor planes on which either illuminance or glare will be calculated. The user may elect to have any number of illuminance and/or glare sensors and goals. The sensor planes may be any size and they may be oriented vertically or horizontally. For glare goals, the user may elect to model an array of small sensor planes (like those in the default model described in section 4.2.4, Figure 4.3). If all glare sensors in an array are modeled using the same material name, they will be considered as a single "glare sensor group."In order for the system to correctly interpret the massing model, the model should conform to a few basic guidelines which enable the model to be understood by the Lightsolve system. These guidelines allow the system to differentiate opaque from transparent materials, materials location outside the interior

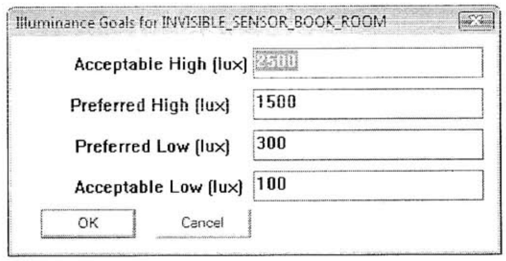


Figure 5.4: Interface for illuminance goal range inputs.

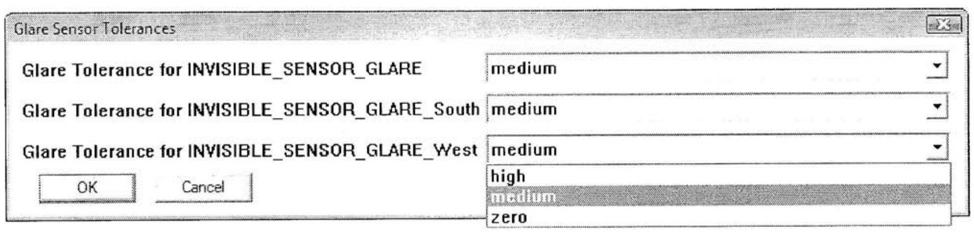


Figure 5.5: Interface for glare threshold goal inputs.

space (such as shading devices), and special surfaces used as sensors for illuminance or glare. The guidelines are described in detail in Appendix A.

One important requirement for models is that all facades that are modeled with windows must be parallel or orthogonal to the cardinal axes in SketchUp. The expert system logic also does not currently take internal walls into account. Figure 5.3 shows an example model in SketchUp that meets the expert system modeling criteria.

Performance Goals

For each sensor plane within the 3d model, the user must specify performance goals. For each illuminance sensor plane, the user must specify a desired illuminance goal range in lux, including the actual desired range and a buffer zone of acceptable values (Figure 5.4). For example, the user may desire the illuminance of a given sensor plane to fall between 400 lux and 1200 lux, but he or she may also accept illuminances as low as 200 lux and as high as 1500 lux. For each glare sensor or glare sensor group, the user must choose a glare tolerance. The glare tolerance options are "zero" (which means that no glare is tolerated), "medium," and "high" (which means that a high amount of glare is allowed). The DGP values corresponding to these tolerance levels are described further in section 5.5.2. The user can input these goals into a simple interface in Google SketchUp (Figure 5.5).

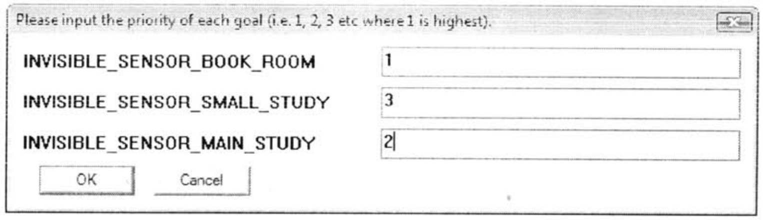


Figure 5.6: Interface for goal priority inputs.

Figure 5.7: Interface for selecting times and seasons of interest.

Goal Priorities

In addition to the performance goals, the user may also specify a priority level for each individual goal. The user may choose to set different priorities for all goals, the same priority for all goals, or a combination. Priorities are input using a simple ranking system: "1" indicates the highest priority goal(s), "2" is the second highest priority goal(s), and so on. The user may specify values in any range from 1 to n, where n is the total number of sensors included in the model. The user priority interface is shown in Figure 5.6.

Window Uniformity Scheme

One of the functionalities of the expert system is to automate changes to facade elements on the user's original model. Because the user may wish to maintain a certain aesthetic, he or she must select one choice from from the following window uniformity options from a pull-down menu in SketchUp:

- All windows in the model should look the same.
- All windows on a facade should look the same.
- Windows can look different from other windows on the same facade.

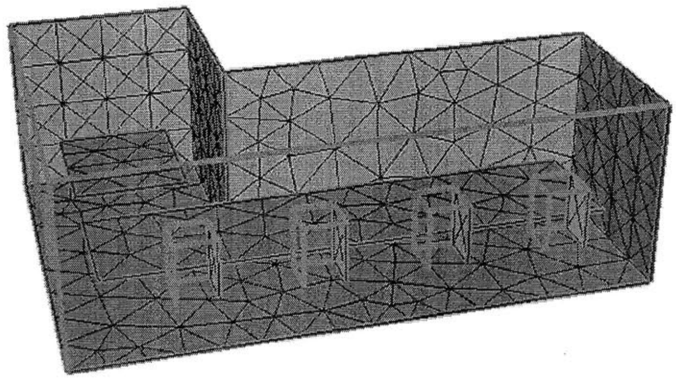


Figure 5.8: Example triangulated 3d model. The goal-based performance is calculated individually for each triangle on a sensor plane.

Times of Interest

The user is allowed to select which seasons and times of day he or she is interested in. The user is allowed choose from: winter, fall/spring, summer, morning, mid-day, and afternoon (Figure 5.7). For example, a designer who is working on a school design may choose only to consider fall, winter, and spring during morning and mid-day. Selecting all seasons and times of day will allow the system to consider performance during all daylit hours over the whole year. The expert system will only consider performance during those times of day and year which correspond to the user's selections.

Weather and Location

A final critical input is the location of the design and its corresponding weather file. The location information includes latitude, longitude, and timezone. The weather file is a Lightsolve-native file with a .wea extension. If an appropriate .wea file is not available, an .epw (EnergyPlus) file can be automatically converted into a .wea file from within the Lightsolve system.

5.5.2 Calculated Data

Once the user has input the required information, the system must calculate some additional crisp inputs. The goal-based performance is calculated using the LSV engine (described in section 4.2.2). The building data model and geometric information are determined by the inference engine.

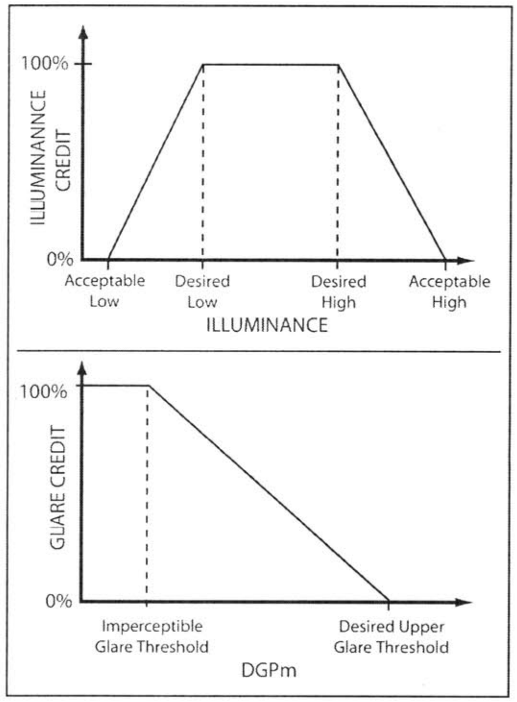


Figure 5.9: Goal-based performance metrics for illuminance and glare.

Goal-Based Daylighting Performance

Goal-based performance for all glare and illuminance sensors are calculated using the 3d model, the location and weather information, the performance goals (illuminance ranges and goal thresholds), and the times of interest.

As mentioned in section 5.5.1, the goal-based illuminance metric requires the user to input four illuminance values: acceptable low, desired low, desired high, and acceptable high. To calculate the goal-based illuminance, the LSV engine first triangulates each sensor into small patches (Figure 5.8), then calculates the climate-based illuminance (section 4.2.3) on each patch over 56 time periods which represent a whole year. For a single patch, the goal-based illuminance metric is defined as the percentage of the user's times and seasons of interest during which daylight provides an illuminance within the user's specified range. The final goal-based illuminance for a sensor plane 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 5.9). 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.

Similarly, the DGPm (section 4.2.3) is calculated on each glare sensor over 56 time periods which represent a whole year. To evaluate glare risks, this thesis uses the glare thresholds described by Wienold (2009), where any value below 0.33 (imperceptible glare)

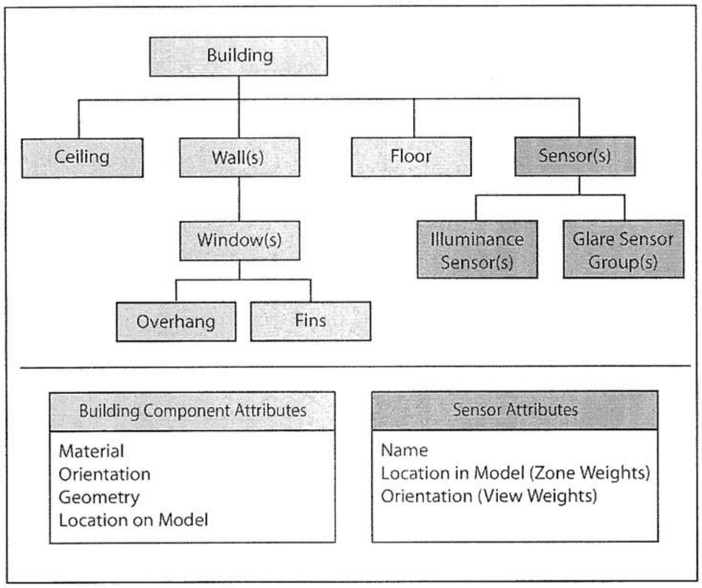


Figure 5.10: Schematic structure for simple building data model generated by the expert system.

is considered a "no glare" situation and given a glare credit of 100%. The user chooses from three tolerances: "zero", which corresponds to an upper glare threshold value of 0.37; "medium", which corresponds to a threshold value of 0.42; and "high", which corresponds to a threshold value of 0.53. Any calculated glare value above the upper threshold is given a glare credit of 0% (Figure 5.9). These glare credits are averaged across all glare sensors in each glare sensor group within the model. A value of 100% indicates that the specified view direction is unlikely to see glare due to daylighting.

Simple Building Data Model

In addition to performance, it is necessary for the expert system to understand the geometry and materials of the design. To accommodate this, a simple 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 5.10. Each building element object contains information about its location, geometry, orientation, and material. The building data model allows the expert system to understand which walls have windows, how large those windows are and where they are oriented relative to each other, as well the shading devices and glazing associated with each window.

In the proposed approach, the user creates a 3d model in SketchUp as an initial input and a simple building data model is automatically created by the system. During the expert system process, the building data model is also updated each time a design change is made to the facade. The logic for the automatic model population is defined in detail in section 6.4.1. Identification of each building element occurs using a series of logic statements regarding the geometry and material of each modeled component. Element attributes are then determined using information available from SketchUp about each face. The logic assumes that the model conforms to a few basic guidelines (Appendix A). The use of the building data model is necessary because SketchUp is a geometric modeling tool and not a building information model (BIM).

Once the building data model has been populated, a few pieces of critical information about the building geometry can then be determined by the inference engine. This information helps the system understand where the sensors, facades, and windows are located in relation to each other. These metrics are described in the next three sections.

Perimeter Zone Weights

Each sensor in the model is assigned a zone weight for each of four perimeter zones (corresponding to the four cardinal orientations) and for the core zone. Perimeter zones are considered to be the area within one wall height away from each facade. Perimeter zones may overlap, and the sensor weights take these overlappings into account. The weights represent how much of the total area of the sensor is located within each zone, and the weights from all five zones sum to 1. Figure 5.11 shows the perimeter and core zones for an example floorplan with one illuminance sensor. In this example, the zone weights for the sensor are: 0.09 North, 0.11 East, 0.17 South, 0.27 West, and 0.36 Core.

The perimeter zone weights are calculated at the beginning of the expert system process and will remain constant regardless of changes to the facade design that occur during the expert system process. These weights are then used to create the customized daylighting databases described in section 5.5.3 and to rank possible design actions during the expert system decision making process.

Window-Sensor Weights

It is also important for the system to understand which windows are most likely to affect the various sensor planes within the system. Within the building data model, each window object is assigned a sensor weight for each sensor or sensor group in the model. The window-sensor weights represent a rough estimate of how much of the total area of the sensor plane is affected by a given window. In order to calculate the weights, the width of each window is projected orthogonally back into the space a total distance of two times the window-head-height, and the area of the projection which falls onto each sensor plane is calculated. The window-sensor weight for a given pair of one window and one sensor plane is calculated as the ratio of the projected area over a maximum possible projected area, defined as the window width times the sensor length in the orthogonal direction. The values that the window-sensor weights can take therefore range from 0 to 1.

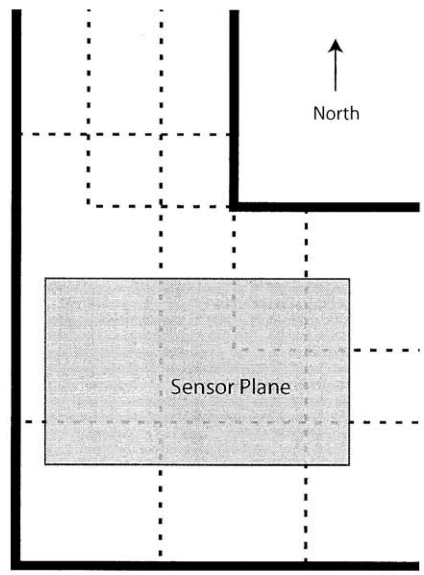


Figure 5.11: Example floorplan with one sensor plane and perimeter zone weights indicated.

An example with three windows is shown in Figure 5.12, where window-sensor weights for each window are: (a) 0.67, (b) 0.13, and (c) 0.69. Based on the weights, the system can assume that windows (a) and (c) will both affect the sensor plane more than window (b).

Unlike the perimeter zone weights, the window-sensor weights may change during the expert system process if window geometries are modified. These weights must be recalculated and updated after each iteration of the expert system process. These weights are used to determine the window groups described in the next section, and they are only used by the system when a user allows the expert system to make changes to individual windows on a facade instead of requiring a uniform facade aesthetic to be maintained.

Window Groups

In addition to determining which design changes to propose to the user, the expert system must also decide which windows the change should be applied to. This choice is in part determined by the user's window uniformity selection: if the user selects "All windows in the model should look the same" or "All windows on a facade should look the same," then the window groups will always consist of either all windows or all windows on each facade, respectively.

However, if the user selects "Windows can look different from other windows on the same facade," then the expert system may elect to choose only a subset of windows on a given

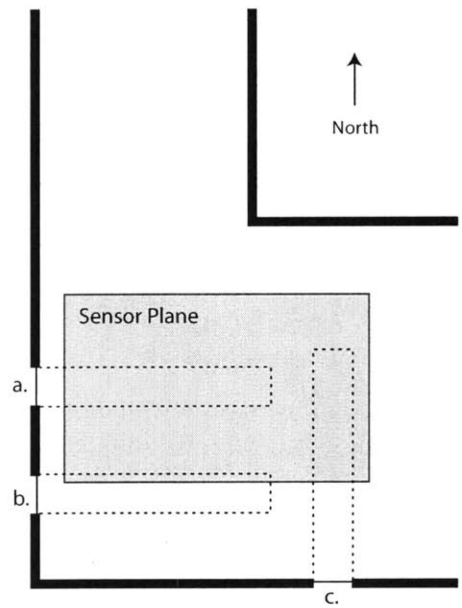


Figure 5.12: Example floorplan with three windows and one sensor. Windows (a) and (c) have large sensor weights and window (b) has a small sensor weight.

facade to be modified. The determination of window groups is based on the window-sensor weights. If, for a given window, the window-sensor weight is greater than 0.15, that window is added into the window group for the appropriate sensor. This means that only those windows which are located close to the sensor will be included. This threshold value was chosen based on trial and error in early studies. For example, in Figure 5.12, the only west facing window that would be selected would be window (a), as window (b) has a window-sensor weight which is too small to qualify. Because the window groups are based on the window-sensor weights, they must be updated after each design change during the expert system process.

5.5.3 Customized Daylighting Database

An important component of the expert system is the ability to create customized daylighting databases. A customized database features a subset of the information contained in the full daylighting database described in Chapter 4 which includes only that data which is most relevant to the user's current design scenario. In order to create the customized database, only the data corresponding to the seasons and times of interest given by the user are taken from the original database.

This subset is further customized for each individual sensor based on the zones in which each sensor is located. The knowledge base data contains different information for each of the five zones (four perimeter zones and one core zone) and for four views from within each zone. For each sensor plane, the data for each zone is multiplied by the sensor's corresponding perimeter zone weight. Additionally, only the relevant views are included for glare sensors. The weighted data is added together to create a single database which is more relevant to that particular sensor plane than to sensor planes located in other zones or for other views within the same model. Therefore, for designs with multiple goals, multiple customized databases are created.

5.6 Expert System Fuzzy Variables

This section describes the fuzzy variables used within the expert system. Fuzzy or "linguistic" variables such as those demonstrated in Figure 5.1 allow an expert system to understand the degree of membership in a set in a less strict way than crisp variables, which can only take the values "true" or "false."

5.6.1 Fuzzy Variables Calculated Before Rule Firing

This section describes the fuzzy sets that are created using the crisp inputs described in section 5.5.

userPriority

The userPriority set is based on the user-input sensor priorities, described in section 5.5.1. The user is allowed to input integer values from 1 to n, where n is the total number of sensors. After the user enters the integer values, they are normalized using the following equation:

$p = 1 - [P/(1 + P_{max})]$

where p is the normalized value, P is the user-input value, and $Pmax$ is the largest value of P input by the user. For example, if a user enters values [1, 2, 3] for 3 sensors, the normalized values are calculated to be [0.75, 0.5, 0.25]. If the user enters values [1, 2, 2] for 3 sensors, the normalized values are calculated to be [0.667, 0.333, 0.333]. The higher normalized values indicate higher priority.

Once the values have been normalized, the fuzzy variables highPriority and lowPriority are calculated for each sensor. If the highPriority variable has a high value (approaching 1.0), then the given sensor has a high priority. If the lowPriority variable has a high value, then the given sensor has a low priority. The membership functions are indicated in Figure 5.13, with an example scenario in which the normalized priority value is 0.6, which results in a 0.6 degree of membership for high priority and 0.4 for low priority.

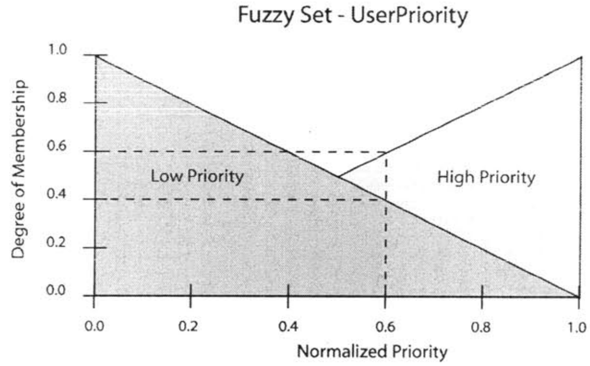


Figure 5.13: Fuzzy set for user-input priorities.

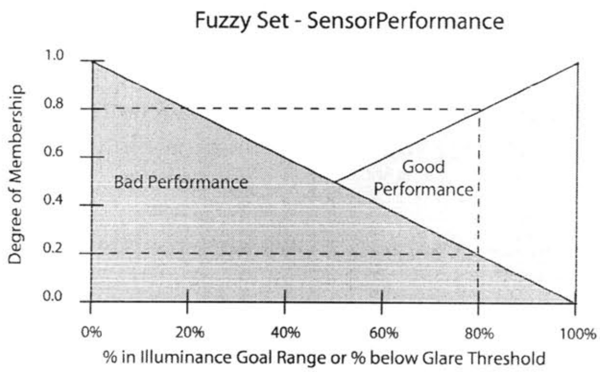


Figure 5.14: Fuzzy set for sensor performance.

sensorPerformance

The sensorPerformance set is based on the current performance of the sensor and is a general set which is applicable to both illuminance and glare sensors. The crisp input necessary to calculate this set is either the average percent in goal range for an illuminance sensor plane or the average percent below the glare threshold for a glare sensor plane, over all the times and seasons of interest, as calculated by the LSV engine. Two fuzzy variables, goodPerformance and badPerformance, are calculated. The sets are indicated in Figure 5.14. The example value in Figure 5.14 indicates a sensor plane which is 80% in the goal range, which results in degrees of membership of 0.8 for good performance and 0.2 for bad performance.

illuminancePerformance

The illuminancePerformance set is based on the current performance of an illuminance sensor plane. This set uses two crisp inputs from the LSV engine: the percentage of the

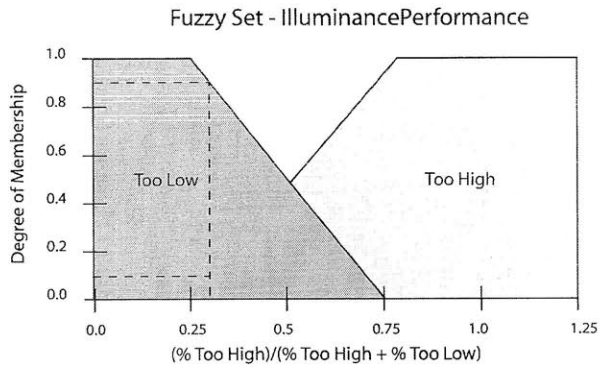


Figure 5.15: Fuzzy set for illuminance performance.

illuminance sensor plane area that sees illuminance that is higher than the goal range, and the percentage of the illuminance sensor plane area that sees illuminance that is lower than the goal range. These values are both averaged over the desired times and seasons. Before obtaining the fuzzy outputs, the crisp inputs are combined into a ratio:

$$Ratio = \%_{high} / (\%_{high} + \%_{low})$$

The ratio is then used to calculate two fuzzy variables, tooHigh and tooLow, as indicated in Figure 5.15. The example in Figure 5.15 demonstrates a situation in which the calculated ratio is 0.3, which results in degrees of membership of 0.9 for too low and 0.1 for too high.

distanceFromGoal

This set is also based on the current performance of an illuminance sensor plane and requires the same two crisp inputs as the illuminancePerformance set: percent too high and percent too low on an illuminance sensor plane. This set qualifies whether the sensor is close to achieving the desired performance goal or far from it. The four variables calculated are: tooHigh_Far, tooHigh_Close, tooLow_Far, tooLow_Close. Figure 5.16 demonstrates an example in which the % too high on a sensor plane is 12%, which results in degrees of membership of 0.8 for far from goal and 0.2 for close to goal.

glarePerformance

Similarly to the illuminancePerformance set, the glarePerformance set is calculated for glare sensor planes and uses inputs from the LSV engine. In this case, the system requires the percentage of glare sensors which see glare that is higher than the desired glare threshold, averaged over the desired times and seasons. One fuzzy variable, tooHigh, is determined, as indicated in Figure 5.17. Figure 5.17 includes with an example case in which 40% glare has been found to be above the glare threshold, which results in a degree of membership of 0.8 glare too high.

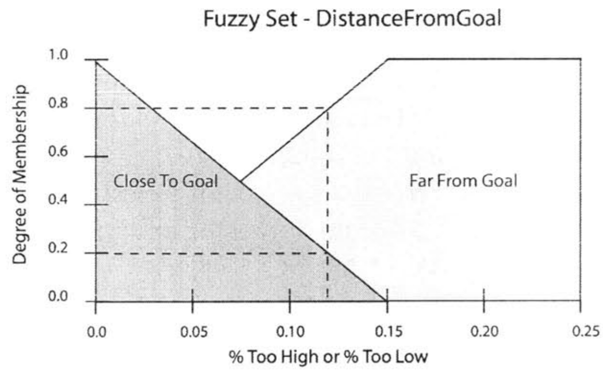


Figure 5.16: Fuzzy set for distance from goal.

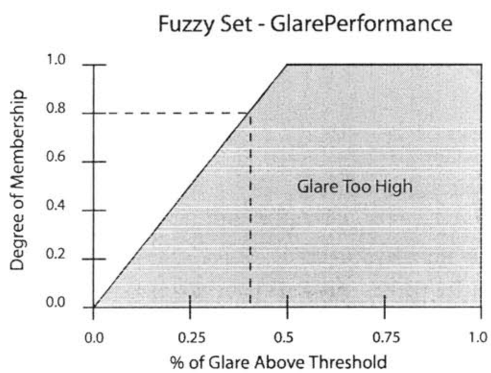


Figure 5.17: Fuzzy set for glare performance.

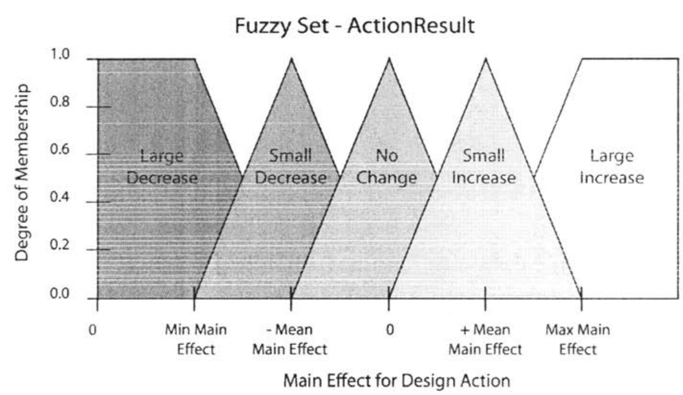


Figure 5.18: Fuzzy set for action result.

actionResult

The actionResult set is used for both illuminance and glare sensor planes. It is used to determine whether a given design action is likely to increase illuminance or glare, decrease illuminance or glare, or do nothing. The necessary crisp inputs are all obtained from the customized knowledge base described in section 5.5.3. These inputs include the main effect of the given design action, the values of the design actions with the highest and lowest main effects, and the mean of the absolute values of all main effects in the customized knowledge base.

The fuzzy variables calculated are: largeIncrease, smallIncrease, noChange, smallDecrease, largeDecrease. These values are calculated as indicated in Figure 5.18.

5.6.2 Fuzzy Variables Calculated During Rule Firing

A number of fuzzy variables are calculated during the process of rule firing. These variables are described below, and the logic by which they are determined is described in section 5.7.

sensorPriority

This set describes the general priority of each sensor, based on a combination of user priorities and current performance. In general, sensors which have high user priority and low performance are given the highest overall priority. The fuzzy variables in this set are highPriority and lowPriority. These variables are calculated in Rule Base 1.

desiredChange

This set describes the desired performance change(s) that will be most beneficial to the overall performance of the design, based on the current performance and the user's goals. The fuzzy variables in this set include: "Increase illuminance by a large amount," "Increase illuminance by a small amount," "Decrease illuminance by a large amount," "Decrease illuminance by a small amount," and "Decrease glare." These variables are calculated in Rule Base 2.

actionPrediction

This set describes whether a design action is predicted to result in good performance for a given sensor, based on the desired performance change necessary to reach the user's goal and the performance change likely to result from that particular design action. The fuzzy variable associated with this set is goodForSensor, which is calculated in Rule Base 3.

5.7 Fuzzy Rules

This section describes the fuzzy rule sets used to determine which design changes to suggest to a user based on a given design scenario. A diagram of the inputs and outputs for these rules is shown in Figure 5.19.

To evaluate these rules, the expert system uses the Zadeh operators (Siler and Buckley, 2005), which are defined as follows:

- x AND y = min(x, y)
- x OR y = max(x, y)

It is important to remember that these rules use fuzzy rather than classical logic, which means that the resultant values will necessarily not be equal to 0 or 1 (true or false). Each rule will be fired once, and each fuzzy variable will be given a value between 0 and 1. For example, in Rule Base 1, the fuzzy set SensorPriority may contain values that are non-zero for both High and Low.

5.7.1 Rule Base 1: Determine Sensor Priorities

The first set of rules determines priorities for each sensor based on a combination of the user priorities and the current performance of each sensor. The rule set is fired once for each sensor plane. The rules are:

1. IF SensorPerformance is Bad AND UserPriority is High, THEN SensorPriority is High.
2. IF SensorPerformance is Good AND UserPriority is Low, THEN SensorPriority is Low.

5.7.2 Rule Base 2: Determine Desired Change

The second set of rules determines the change(s) in illuminance or glare that the system should aim for to improve performance based on the user's goals. This rule set helps to determine whether the system should attempt to increase illuminance, decrease illuminance, and/or decrease glare. It also attempts to determine the magnitude of change that will be most helpful for overall performance. Because illuminance goals may include a range of desired values, it is possible that an illuminance sensor will have multiple positive values (for example a single sensor may have positive values for "Increase illuminance by a large amount", "Increase illuminance by a small amount," and "Decrease illuminance by a small amount"). The use of fuzzy variables allows such a condition to be correctly understood by the system.

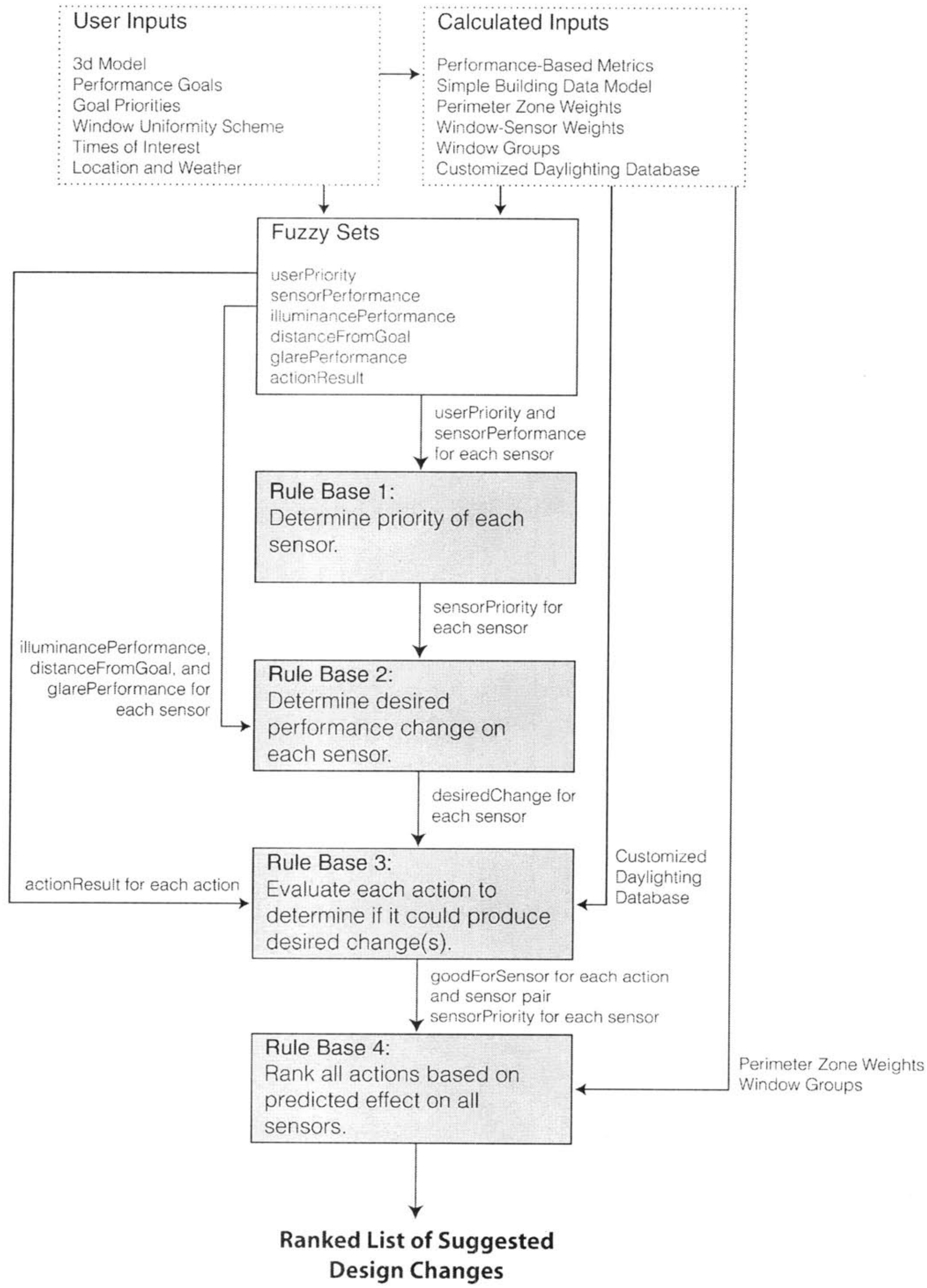


Figure 5.19: Schematic diagram of the fuzzy rule sets and their inputs and outputs.

This rule set is fired once for each sensor plane. The rules in this rule base are:

1. IF SensorPriority is High AND SensorType is Illuminance AND IlluminancePerformance is TooLow:
   (a) IF distanceFromGoal_Low is Far, THEN DesiredChange is "Increase illuminance by a large amount."
   (b) IF distanceFromGoal_Low is Close, THEN DesiredChange is "Increase illuminance by a small amount."
2. IF SensorPriority is High AND SensorType is Illuminance AND IlluminancePerformance is TooHigh:
   (a) IF distanceFromGoal_High is Far, THEN DesiredChange is "Decrease illuminance by a large amount."
   (b) IF distanceFromGoal_High is Close, THEN DesiredChange is "Decrease illuminance by a small amount."
3. IF SensorPriority is High AND SensorType is Glare AND GlarePerformance is TooHigh, THEN DesiredChange is "Decrease glare."

5.7.3 Rule Base 3: Evaluate Actions

The third set of rules determines whether a given action is likely to produce the desired performance change (calculated in Rule Base 2). These rules are fired once for each potential design change and for each sensor plane.

The rules in this rule base are:

1. IF SensorType is Illuminance:
   (a) IF DesiredChange is "Increase illuminance by a large amount" AND ActionResult is LargeIncrease, THEN actionPrediction is GoodForSensor.
   (b) IF DesiredChange is "Increase illuminance by a small amount" AND ActionResult is SmallIncrease, THEN actionPrediction is GoodForSensor.
   (c) IF DesiredChange is "Decrease illuminance by a large amount" AND ActionResult is LargeDecrease, THEN actionPrediction is GoodForSensor.
   (d) IF DesiredChange is "Decrease illuminance by a small amount" AND ActionResult is SmallDecrease, THEN actionPrediction is GoodForSensor.

2. IF SensorType is Glare:

    (a) IF DesiredChange is "Decrease glare" AND ActionResult is LargeDecrease, THEN actionPrediction is GoodForSensor.

If the sensor is a glare sensor, the value GoodForSensor is used as is. If the sensor is an illuminance sensor, the value of GoodForSensor is calculated using Sugeno-style inference (Sugeno, 1985). Using this method, the output of each fuzzy rule from 1a through 1d is represented by a single value. These values are then added together to represent a weighted approximation of how the proposed change will affect the sensor in all possible ways.

5.7.4 Rule Base 4: Rank All Actions

The final rule base allows the system to rank all the actions based on the predicted effects of each action, the sensor priorities, the window groups, and the sensor weights. The general algorithm used is:

- For all possible design actions:
    - For all sensors:
        * Calculate for the design action and sensor: actionRank[sensor] = (sensorPriority) x (GoodForSensor) x (sensorWeight for relevant facade)
    - Find the total actionRank value when summed over all sensors.

The final step is to sort the set of design actions from highest to lowest actionRank. The first design actions in the list will be those actions most likely to produce positive performance results in the current design, while those actions at the end of the list are likely to decrease overall performance.

5.8 An Example Scenario

This section demonstrates an example design scenario with expert system results. The model used in this example is shown in Figure 5.20. There are two illuminance sensor planes, each with different goals.

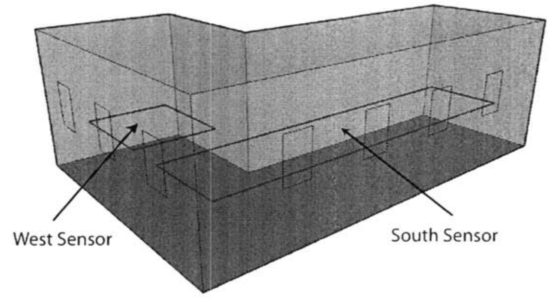


Figure 5.20: Example model with two illuminance sensors.

The major inputs are:

- Performance Goals:
  - West Sensor - Illuminance range: Maximum 800 lux desired, 1000 lux acceptable; minimum 200 lux desired, 0 lux acceptable.
  - South Sensor - Illuminance range: No maximum; minimum 500 lux desired, 400 lux acceptable.
- Goal Priorities:
  - West Sensor - 1 (higher priority)
  - South Sensor - 2 (lower priority)
- Window Uniformity Scheme: Windows can look different from other windows on the same facade.
- Times of Interest: Full year
- Location: Boston, MA
- Performance calculated by the simulation engine:
  - West Sensor - 40% in range, 59% too high, 1% too low
  - South Sensor - 90% in range, 0% too high, 10% too low

Due to the location of sensors within the space, the customized daylighting database for the west sensor contains only information about the west zone. The customized database for the south sensor contains information about the south and west zones (with the south zone information weighted more heavily).

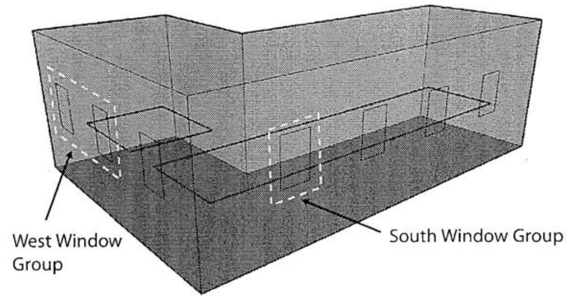


Figure 5.21: Example model with window groups highlighted.

Based on the goals, current performance, and geometry of this scenario, the expert system proposes to change only those windows indicated in Figure 5.21. The first eight proposed changes are calculated to be (in order of likelihood to improve performance):

1. Add an overhang to the west window group.
2. Decrease the area of the west window group.
3. Add fins to the west window group.
4. Make the west window group glazing more translucent.
5. Make the west window group glazing less transmissive.
6. Add an overhang to the south window group.
7. Decrease the area of the south window group.
8. Add fins to the south window group.

These design changes all focus on decreasing the illuminance on the West sensor in order to improve its performance.

5.9 Chapter Summary

This chapter discussed the logic used by the expert system in order to determine which design changes to suggest to a user based on a given design scenario and a given set of performance criteria. The expert system uses fuzzy logic instead of classical logic in order to better emulate the human thought process.

This chapter outlined the various inputs used by the system in order to make decisions, including user goals and preferences, model and sensor geometry, and a customized database of daylighting-specific information created from the knowledge base described in Chapter 4. Fuzzy variables and rule sets are used in order to deal with multiple sets of goals and different levels of goal priorities. The output of the fuzzy rule set is a ranked list of possible design changes which are likely to improve the daylighting performance of the current design. In situations where a user allows the system to create non-uniform facades, the system also selects a subset of windows on each facade which are most likely to affect the higher priority sensors.

The fuzzy expert system has been designed to be a flexible system which has the capabilities to deal with a wide variety of design scenarios, including different types of building footprints and various combinations of illuminance and glare goals, including possibly conflicting goals. The system has also been designed to allow for user interaction so as to be compatible with the architectural design process. A performance evaluation of the expert system as an algorithm which can improve performance in the absence of user interactivity will be discussed in Chapter 7. Assessment of the system of as a design tool will be described in Chapter 8.

Chapter 6 will discuss the implementation of the expert system using SketchUp and will demonstrate the expert system user interface.

Chapter 6

System Implementation

6.1 Introduction

One of the main goals of this thesis was to develop a daylighting expert system which could be fully integrated into the architectural design process. To enable this integration, it was necessary to create a functional tool which allows users to interact with the expert system during the design process. This chapter describes the overall system structure of the tool and the user interface which connects the user to the system.

This chapter also describes several key features which enable the user to efficiently and effectively design with the tool. One important feature of the system is to allow the designer to analyze his or her own initial design using a 3d model that he or she created (instead of a generic box model). Another important feature is that if the user elects to make a design change during the expert system process, the system will automatically make the appropriate change to the 3d model. To implement these two features, a simple building data model was developed which allows the expert system to correctly interpret the geometry of the user's model. This chapter describes the building data model structure, the logic for its population, and the logic for automated design changes.

6.2 System Structure

The expert system tool was developed as a part of the original Lightsolve project (Andersen et al., 2008) and has been integrated into the Lightsolve system. The general system structure is shown in Figure 6.1. Google SketchUp (Google, 2008) is used as the 3d modeler, and the embedded Ruby application programming interface (API) within SketchUp is used to create the pop-up interfaces which allow the user to enter the initial inputs

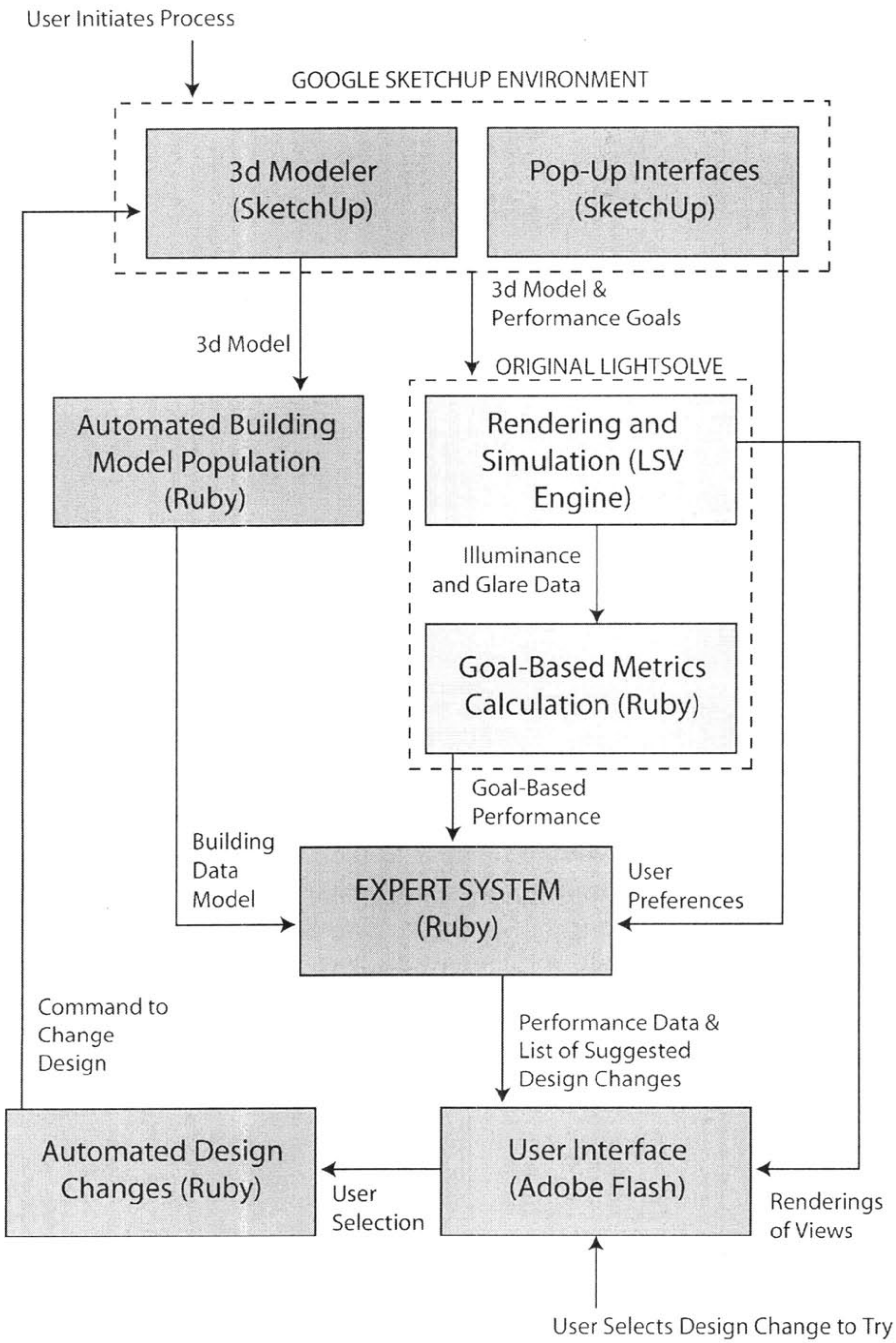


Figure 6.1: Schematic diagram of expert system structure and process

and to perform the major processes and calculations. The LSV simulation engine (section 4.2.2) is a stand-alone executable which is called directly from within the SketchUp/Ruby environment. The user interface has been implemented using Adobe Flash and communicates with SketchUp by sending commands through the SketchUp Bridge (labs.plugins.ro, 2006).

The expert system process is initiated from within the Google SketchUp modeling environment, where the user creates an initial 3d model and specifies his or her goals and preferences using the pop-up interfaces shown in section 5.5.1. The 3d model is used by the LSV simulation engine to create renderings of views from within the model and to calculate full-year, climate based illuminance and glare (DGPm) on all sensor planes. The 3d model is also used to populate a simple building data model (described in section 6.4). The goal-based performance metrics are then calculated using information about the user's goals and data from the LSV engine.

The goal-based performance data, the building data model, and the user's preferences are all used as input into the expert system (described in section 5.5.1). Once the expert system has determined a list of possible design changes to suggest to the user, the interface displays this information, along with performance data and renderings of views. Using the interface, the user selects a design change to make to his or her design. The system then automatically makes three different models, each representing a different magnitude of the selected design change (for example, the window area may increase by 10%, 20%, and 30%). The logic regarding the calculation of these magnitudes of change will be discussed further in section 6.4.2. Each of these three models is then simulated and the expert system is used to produce a new list of suggested design changes for all three models. This information is then displayed in the user interface, where the user will first select one of the three models to accept before selecting the next design change. The process will continue until the user's goals are met or until the user ends the process.

6.3 User Interface

The user interface for the expert system is shown in Figures 6.2, 6.3, and 6.4. There are four major components to the interface:

- **Views of the current design:** In this window, the user can scroll through four different views of the current design (southeast, southwest, northwest, and northeast views). The images of the views indicate the user's current design choice and are updated to display any design changes that occur during the process.

- **Temporal maps of all illuminance and glare sensor planes:** In this window, the user can scroll through all available temporal maps. Like the views of the current design, the temporal maps are updated at each design step to display current performance data.

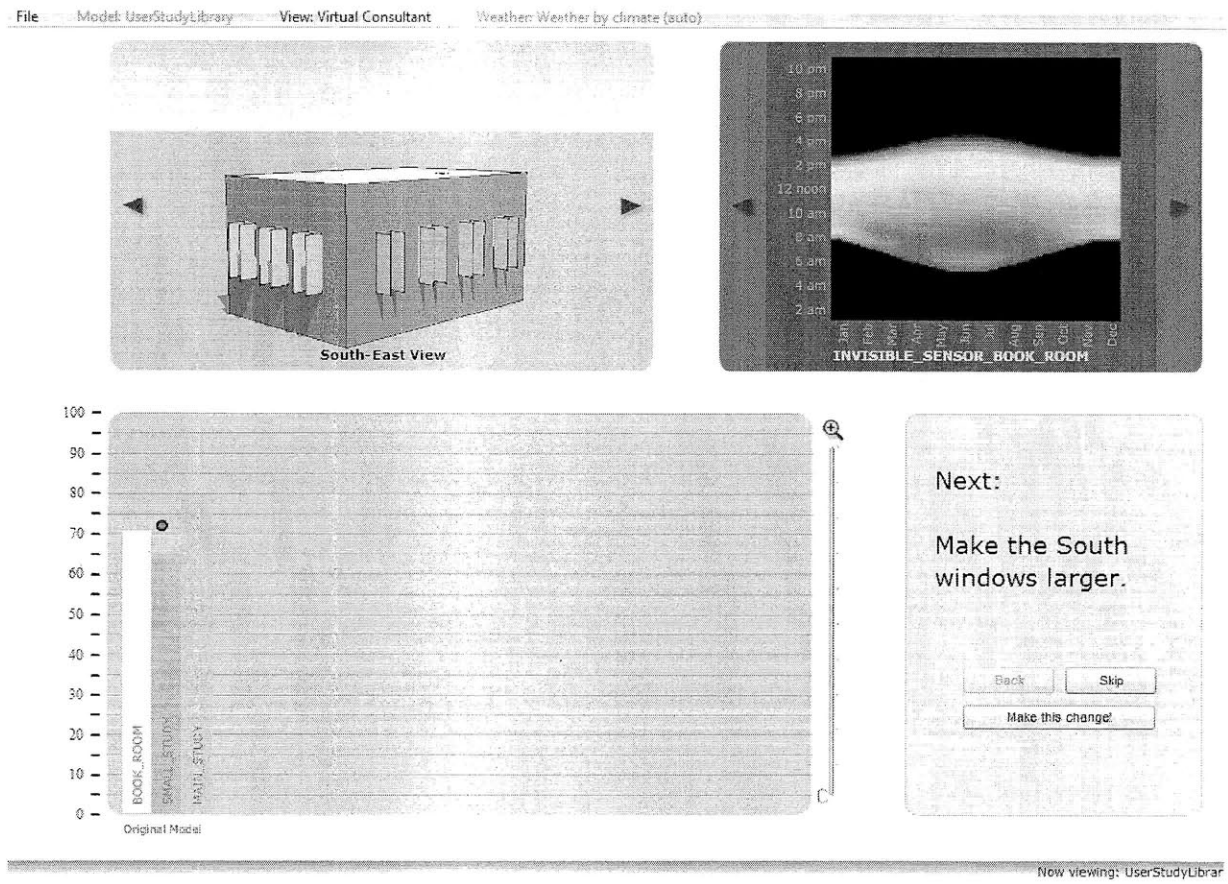


Figure 6.2: Expert system user interface after first run. The user may choose to make the first suggested design change ("Make the South windows larger") or to skip to a different suggested design change.

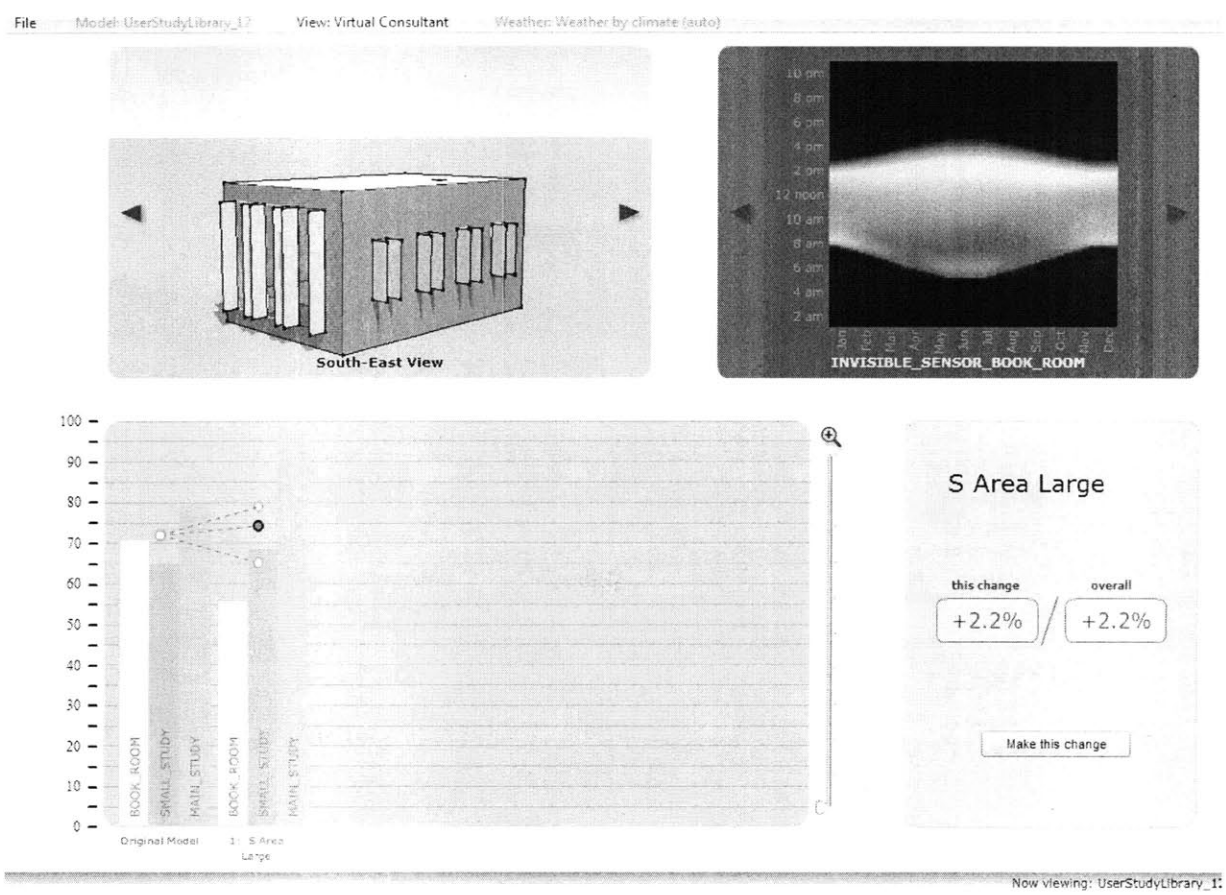


Figure 6.3: Expert system user interface after one design step (south windows were increased). The user must choose one of the three different magnitudes of change indicated in the interactive graph.

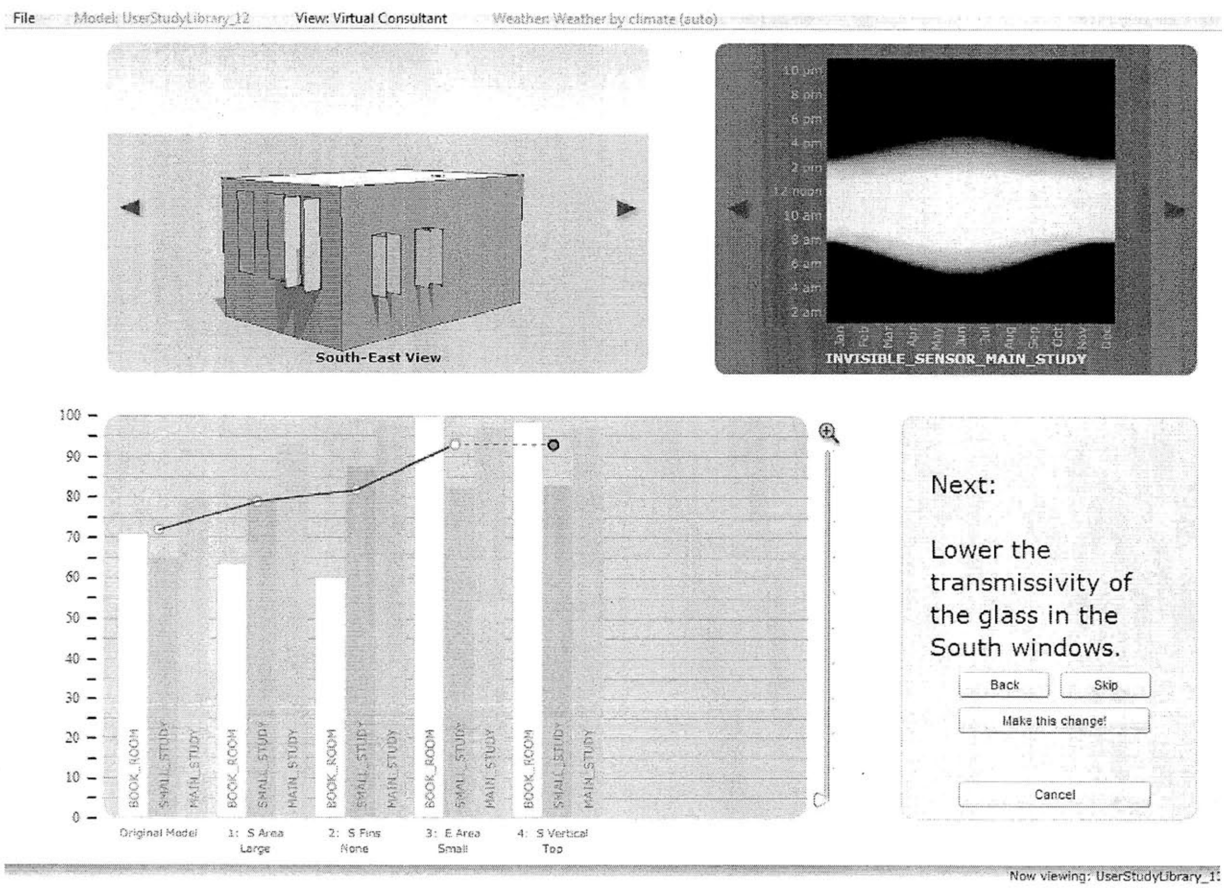


Figure 6.4: Expert system user interface after four design steps. The progression of designs explored during the process is shown in the interactive graph.

- **Suggested design changes:** This window displays one design change at a time, in order of highest rank (most likely to improve performance) to lowest rank (Figure 6.2, lower right section of interface). The user can choose to skip ahead to select design changes that appear later in the list, and they can go back to select an earlier one. Once the user selects a design change, that change will automatically be applied to the model in SketchUp for three different magnitudes (for example, three different overhang lengths will be tried). In situations where the user has allowed the expert system to change individual windows on a facade (instead of keeping all windows uniform), the edges of those windows that are to be changed are highlighted in yellow (Figure 6.2).

- **An interactive graph indicating the performance of the design over all design iterations:** This window allows the user to view the performance of each sensor plane (averaged over all relevant times and seasons) in a bar as well as the average of all sensor plane performances together as a dot. After the user selects to make a design change, the results for the three different magnitude of change are shown in this window (Figure 6.3). The user will select which of the three magnitudes he or she prefers by clicking on the dot which refers to the preferred design (at this point, that dot becomes filled in). As the process continues, this graph will display the progression of design changes selected by the user along with the performance of each new design (Figure 6.4). At any point during the process, if the user selects one design iteration in this window (by clicking on the appropriate dot), the rest of the interface will automatically update to display the information specific to that design. The user can go backwards to a previous design iteration simply by selecting the dot that refers to that design.

The user interface supports the design process described in section 3.5 by providing the user with clear views of the current design and by allowing the designer to browse performance information in an intuitive way. The user is able to see the progression of his or her design using the interactive graph and can quickly move between previous iterations of the design and the current version. The suggested design changes are specified in regular language that the designer can easily understand. The automatic generation of design changes along with the intelligent design suggestions allow the user to efficiently explore the design space and identify designs with improved performance.

6.4 Building Data Model

The simple building data model used by the expert system (section 5.5.2) allows the system to understand the geometry and materials of the user's design. Using this model, the system knows which shading devices belong to which windows, what type of glass is used for a given window, how many windows are on a given facade, and so on. The building data model is not only important to the expert system logic itself, but it also enables the expert system tool to determine the design boundary conditions and make automated design changes. The following sections describe the logic used to populating the building data model and for automating the design changes.

6.4.1 Logic for Automated Model Population

The expert system process begins with a user-defined 3d model. The user may model almost any initial design, as long as the model meets the guidelines described in Appendix A. 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 called "GLASS_[name]", it is a window.
2. If the face is opaque and called "EXTERNAL_[name]", 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_[name]".
   (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 is called "SENSOR_[name]", it is a sensor plane.
   (a) If it is called "GLARE_SENSOR_[name]", it is a glare sensor.
   (b) Else, it is an illuminance sensor.

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, consider each window and if two vertices of the overhang are located 1 inch [0.025 m] 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. A given window may have at most one overhang and two fins.

3. Assigning glare sensors to sensor groups: Cycle through all glare sensors. Any glare sensors that have the same material name are assigned to a glare sensor group.

Once the building data model has been populated, the system will also calculate the sensor perimeter zones, window-sensor weights, and window groups as described in section 5.5.2. A new building model is created every time the original model changes, either manually by the user or automated by the system.

6.4.2 Logic for Automated Design Changes

Each time a user selects a design change from the expert system interface, the system will automatically produce three new models, each with a different magnitude of change. A critical set of calculations during this part of the process is the determination of the maximum change allowed, given the geometry of the original model. The system assumes that the general massing model will stay the same while the facade elements may change, so it is important to ensure that a given design change does not result in windows are located outside the facade boundaries or windows that overlap each other.

The system also determines if a given design change can legally be applied to a model. If a design change cannot be applied to the design, that design change will not be displayed in the user interface. For example, if the design change is "move windows to the south end of the facade," but the windows are already located as far south as is possible given the geometry of that facade, then the design change will be removed from the list that is displayed to the user.

The determination of the maximum change allowed is calculated based on the window groups described in section 5.5.2. The system assumes that a given design change will be made to all windows in a window group. Windows that are on the same facade as a window group but not included in that group are considered as boundary conditions.

The following sections describe the logic used to calculate the maximum change allowed and the three different magnitudes of change for all available design change options.

Aperture Location on Facade

Four different design changes involving translation are possible: moving windows higher, lower, right, and left on the facade. Shading devices associated with the window group are translated as well.

To determine the three different magnitudes of change, the system first calculates the maximum possible translation. The maximum translation is that which results in the window group being located 2 inches [0.05m] from either the facade boundary or another set of windows. The maximum translation is then divided by three. For example, if it is possible to move a window group 3 ft higher on the facade, the system will create three models with 1ft, 2ft, and 3ft translation.

Window Distribution

Two changes are available based on window distribution: moving windows in a window group closer together or further apart. Shading devices associated with the windows are moved as well. The system recognizes that windows may not be spaced uniformly on the facade, so it attempts to retain the original characteristics of the window spacing by maintaining the same relative distance between each set of adjacent windows.

To determine the highest magnitude of change, the system determines the maximum possible change in distribution. The maximum change in distribution is that which results in the windows in the window group being located 2 inches [0.051 m] from each other (if they are moving closer together) or two inches from the facade boundary or another set of windows. The maximum translation is then divided by three.

Aperture Aspect Ratio

The two available aspect ratio (height-to-width ratio) design changes are: increasing the ratio (narrower windows) or decreasing the ratio (wider windows). This change involves changing the aspect ratio of all windows in the window group while keeping the original area of each window the same. Shading devices are also scaled appropriately.

The maximum possible change is that which allows all windows to fit on the facade without overlapping each other and without moving beyond the facade boundary conditions. The maximum change is divided by three to determine the three different magnitudes of change. For example, if it is possible to make a window 60% wider, the system will create three models with increasing wideness of 20%, 40%, and 60%.

Total Number of Apertures

Changing the total number of apertures involves either increasing or decreasing the number of windows in a window group while keeping the same total area of the group and the same aspect ratio of each individual window. This change assumes that all windows are the same general size and shape. Shading devices are also removed or added along with the windows.

For this change, the system assumes that it will add or subtract one window per design change. For example, if the window group begins with 2 windows and the design change is to increase the number of windows, the system will create three models with 3, 4, and 5 windows respectively.

Total Area of Glazing

Two options are available for changing the total window area: increasing the area and decreasing the area. This change assumes that the area of each window in the window group will change while the aspect ratio of each individual window will stay the same. Shading devices are also scaled appropriately.

The system calculates the maximum amount that it can increase windows based on the largest window that it can create without having the windows overlap or move outside the facade boundary. The system assumes that the maximum amount it can decrease the area of all windows is 100%, i.e. the final magnitude of change will be to remove the windows completely. The three magnitudes of change are calculated by dividing the maximum amount of change by 3. Therefore, if the system is decreasing the area of a window group, the three magnitudes of change will always be 33%, 66%, and 100% smaller. If the system is increasing the area of a window group, and the maximum amount is calculated to be 30%, it will create three models with 10%, 20%, and 30% increases in area.

Glazing Type

Four possible design changes are available which involve glazing type: increase transmissivity, decrease transmissivity, make more translucent (as opposed to transparent), or make less translucent. All window glazing is defined using a percentage of "regular" (specular) transmissivity and a percentage of diffuse (translucent) transmissivity.

The maximum possible total transmissivity is allowed to be 90% and the minimum allowable transmissivity is 5%. The maximum allowable percentage of regular transmissivity is 100% (completely clear transparent glass) and the maximum allowable percentage of diffuse transmissivity is 100% (completely translucent). For example, if the current glazing is 60% clear glass and the design change is "increase transmissivity", the three new models will have glass with 70%, 80%, and 90% transmissivity, all clear glass. If the current glazing is 60% regular glass and the design change is "increase translucency", the three new models will have glass with a total transmissivity is 60%, but the ratio of diffuse to specular transmission will be 0.33, 0.66, and 1.0.

Shading Devices

Four possible design changes involve shading devices: increasing or decreasing the length of horizontal overhangs and increasing or decreasing the length of vertical fins. In the current version of the system, the maximum possible shading device length that the system will create is 4ft [1.2m]; however, it is possible that this value could be adjusted based on the height of the space or a user-defined limit. The minimum possible length is 0ft, i.e. the shading device is removed completely. The three different magnitudes of change are determined by subtracting the current length form the maximum (or minimum) length and dividing by 3. For example, if the current overhang length is 1ft and the design change is "increase overhang length," then the new models will have overhangs will lengths of 2ft, 3ft, and 4ft respectively.

6.5 Chapter Summary

This chapter discussed the implementation of the expert system as a functional design tool with an intuitive user interface. The expert system tool and its interface have been designed to be compatible with the architectural design process and to support an efficient and intelligent performance-driven search. The tool includes many key functionalities, including the ability for the designer to model his or her own design (instead of relying on a default model), the ability for the designer to interact with the system by choosing which design changes to try and which magnitude of change to accept, and the ability for the designer to clearly understand the performance of his or her design over a series of design iterations. A study which demonstrates the user compatibility of the tool has been conducted and will be described in Chapter 8.

This chapter also discussed the simple building data model used by the expert system and the logic used to automatically populate the data model from a user's original 3d model. The building data model is a critical component of the system as it contains information about the user's geometry and materials which is used by the expert system rule base to help it determine which design changes to suggest to the user. This information is also used by the system to determine the maximum possible design change that it can make to a given model and to calculate the three different magnitudes of change that the system will attempt.

Chapter 7 will describe the validation of the expert system performance when compared to that of a genetic algorithm for a variety of case studies.

Part III

Results

Chapter 7

Expert System Performance Evaluation

7.1 Introduction

The main function of the expert system described in this thesis is to effectively guide a user towards improved daylighting performance of an original design. It is of critical importance that users have confidence in the advice given to them by the system, so a high level of performance is requisite. Although the expert system differs from a traditional optimization algorithm due to its domain-specific and user-interactive nature, it should be capable of performing similarly to an optimization algorithm in a best case scenario.

In order to assess the behavior of the expert system, a series of case studies were completed which compare the performance of designs found using the expert system to high-performing benchmark designs generated using a genetic algorithm, a popular optimization method. The case studies include problems with a variety of initial conditions and goal criteria, ranging from a single performance goal for a simple box model to multiple performance goals in more complex models.

Two additional short studies were completed to demonstrate the relationship between the expert system behavior and the facade design or constraints imposed by a user. The first of these two studies demonstrates how performance may be related to the specificity of the initial facade designed by the user. The second study considers how performance may be affected by the window uniformity scheme selected by the designer.

The goal of the case studies described in this chapter is to examine a variety of different scenarios and to determine the strengths and limitations of the expert system as an algorithm which can effectively work towards finding improved performance. For all case studies, the user-interactive element of the expert system was removed to allow for fair and consistent comparisons of results. The performance of the expert system when user-interactivity is allowed will be discussed in Chapter 8.

This chapter describes the genetic algorithm which was implemented for the major set of case studies and present the case study results in order of increasing levels of design problem complexity. This chapter also presents an analysis of the performance and behavior of the expert system based on the various case studies considered. The chapter concludes with a general discussion of the strengths and limitations of the expert system as a performance-driven algorithm.

7.2 Comparison Study: Expert System vs. a Genetic Algorithm

This section presents the major set of case studies used to evaluate the performance of the expert system. In each case study, the performance of a "best case" design generated by the expert system is compared to a "best case" design generated by an optimization algorithm. Due to the generic nature of the expert system, which allows the user to input almost any initial design and set of performance goals, it is difficult to precisely determine the overall performance of the expert system algorithm. The purpose of this set of case studies is thus to demonstrate the behavior of the expert system over a selection of problems with a range of initial design geometries and performance goals.

For this study, it was necessary to choose an optimization algorithm which would be appropriate for comparison with the expert system. The genetic algorithm (GA) was chosen for a number of reasons, particularly because the GA has been previously used for a wide variety of building-related problems (see section 2.2 for examples). Additionally, the GA is one which is similar to the expert system as it is a generic algorithm, one which can be applied to a large assortment of initial designs and performance goal scenarios without needing to be customized. Finally, the GA is one which is known to be consistent, reliable, and does often not get trapped in local minima or maxima. While the GA may not always find the global minima or maxim, it will always provide a very good benchmark.

The GA is an algorithm which works by imitating the process of natural evolution. During the GA process, a set of initial solutions (a population) is chosen or generated at random. Each member is evaluated for "fitness" (performance) and members that result in good performance are used as "parents" for a new generation. Parent members are combined using a genetic operator called crossover to create a new generation of "child" members which have characteristics of the parent members. Since this new generation is based on the best performing solutions in the previous solutions, it is assumed that some members of the new generation will perform better. Once evaluated, again 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 or until a predetermined number of generations have been completed.

Multi-objective GAs work in a similar way except that in these cases, one might consider two or more objectives which are conflicting. In such cases, increasing the fitness of one objective may decrease the fitness of another, which means that a single optimal solution for all objectives may not exist. Instead, it is traditional to try to find the Pareto front, which is the set of all solutions in the solution space that are non-dominated, or Pareto-optimal. If for a given solution, one can find another one within the solution space that is better for both objectives, that solution is considered strongly dominated. Pareto-optimal or non-dominated solutions are those which are not dominated by any others within the solution space.

7.2.1 A Micro-Genetic Algorithm

GAs typically require large population sizes and numbers of generations to converge, particularly for multi-objective problems where the desired result is not a single solution but a set of Pareto-optimal solutions. The GA used for the case studies described in this chapter was a micro-genetic algorithm (micro-GA), a GA which uses a very small population size when compared to a classical GA. This small population size reduces the computational time necessary to simulate each generation, which means that several generations of the micro-GA can be run using the same number of simulations as a single generation of a classical GA. Micro-GAs have also been shown in some studies to require fewer total function evaluations than a classical GA to converge to the near-optimal region (Krishnakumar, 1989; Carroll, 1996). For this thesis, the micro-GA was selected because the LSV engine requires a relatively large amount of time necessary for to complete one climate-based, full year daylighting simulation (1-5 minutes on the author's computer).

Micro-GAs have been successfully implemented for building performance optimization based on building energy criteria, lighting, and thermal behavior (Caldas and Norford, 2002; Caldas, 2008). The micro-GA system implemented for this thesis can be used for both single- and multi-objective problems. The single-objective problem considers illuminance or glare only, while the multi-objective problem considers two conflicting illuminance goals at once or both illuminance and glare.

7.2.1.1 Single Objective Micro-GA

Single-objective optimization problems aim to optimize a single function, i.e. those which have a single performance goal. Problems which have multiple non-conflicting goals may also be modeled as single-objective problems by averaging or adding the performance of all goals into a single function. The single-objective micro-GA used in this thesis is the original algorithm as described by Krishnakumar (1989), encoded using binary strings. The process is as follows:

1. Generate a random population of five 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.

A micro-GA differs from a traditional GA in several ways. The most obvious is the small population size, which tends to cause the micro-GA population to reach bitwise convergence within only a few generations. At this point, the micro-GA resets the population by creating a new random population. It is of note that this algorithm does not use mutation, as it is assumed that enough diversity will be maintained in the population through the generation of new random strings upon bitwise convergence, which is likely to occur numerous times during the optimization process.

For the single-objective case studies described in this thesis, fitness is defined as the goal-based illuminance for a single sensor plane or the goal-based glare for a glare sensor array. For problems which have multiple goals that are not considered to be conflicting, fitness is defined as the average goal-based illuminance or glare over multiple sensor planes. An "optimal" solution will be one in which fitness is found to be 100%, which indicates that the performance goals are met over 100% of the sensor plane area and over 100% of the time during a year. For a simple problem, it may be possible that multiple solutions within the solution space meet the goal criteria (an example of this phenomenon is described in section 7.2.4).

7.2.1.2 Multi-Objective Micro-GA

For problems in which multiple conflicting performance goals are considered, a multi-objective algorithm is required. The micro-GA has previously been successfully used for multi-objective problems by including external memory which stores non-dominated solutions generated over the course of the process (Coello and Pulido, 2001). For this study, the algorithm used is similar to that described for single-objective problems (Krishnakumar, 1989), with the addition of an external memory similar to that described by Coello and Pulido (2001). A binary Pareto fitness ranking is used, and 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. A pseudo-Pareto front is approximated by those solutions contained within the external memory after a certain number of generations. The multi-objective process is essentially the same as the single-objective process except it works towards finding non-dominated solutions instead of working towards a single solution with the highest fitness.

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, as this study aims only to evaluate the performance of the expert system and not to find true Pareto fronts, it is assumed that an approximated Pareto front will still provide an adequate level of accuracy for comparison purposes.

7.2.2 Micro-GA System Description

The micro-GA was implemented into the Lightsolve system so that results from both the micro-GA and the expert system could be directly comparable. The micro-GA was created to have the same or similar features to the expert system in the following areas:

- 3d Modeling - Both systems use Google SketchUp to model the geometry and materials of the design.
- Performance Goals - Both systems require illuminance and glare sensors to be modeled within the 3d model and require the user to input performance goals in the interfaces described in section 5.5.1. Calculation of the goal-based performance metrics is the same for both systems.
- Simulation Engine - Both systems use the Lightsolve Viewer (LSV) simulation and rendering engine.
- Design Variables - Both systems consider the same ten variables. The expert system allows two directions of change for each variable (for example, the system might suggest to increase or decrease the area of windows). The micro-GA system allows a range of possible values as indicated in Table 7.1. The full set of values is encoded into a string of 30 bits for each separate facade considered.

7.2.2.1 Micro-GA System Process

The process of using the micro-GA system is as follows:

1. A massing model is created with sensor planes modeled. This massing model stays the same through the process while the facades are generated by the system. The initial massing model should not have any windows modeled.
2. The performance goals are input using the interfaces shown in section 5.5.1.

Façade Parameter	Min	Max	Step
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%
** Absolute values of max and min for these parameters will depend on user-defined geometry*			

Table 7.1: Design parameters used in the micro-GA system

3. The system generates a random initial population of binary strings.
4. 3d model representations of each binary string are created using the logic described in section 7.2.2.2.
5. The population is simulated using the LSV engine. Goal-based performance metrics are calculated for each member of the population.
6. The micro-GA uses the results of the previous generation and their binary strings to generate a new population of binary strings.
7. Return to step 4 and repeat until the fixed number of generations is completed.

7.2.2.2 Micro-GA Model Generation

The micro-GA system created for this thesis automatically generates 3d model representations of the binary strings created during the GA process, i.e. it creates and saves new SketchUp models for all population members. Like the automated design changes described for the expert system in section 6.4.2, the micro-GA system uses a simple building data model based on the original massing model in order to understand the dimensions of each building element (section 6.4).

The generated models are created using the following process (steps 1-5 are demonstrated for an example facade in Figure 7.1):

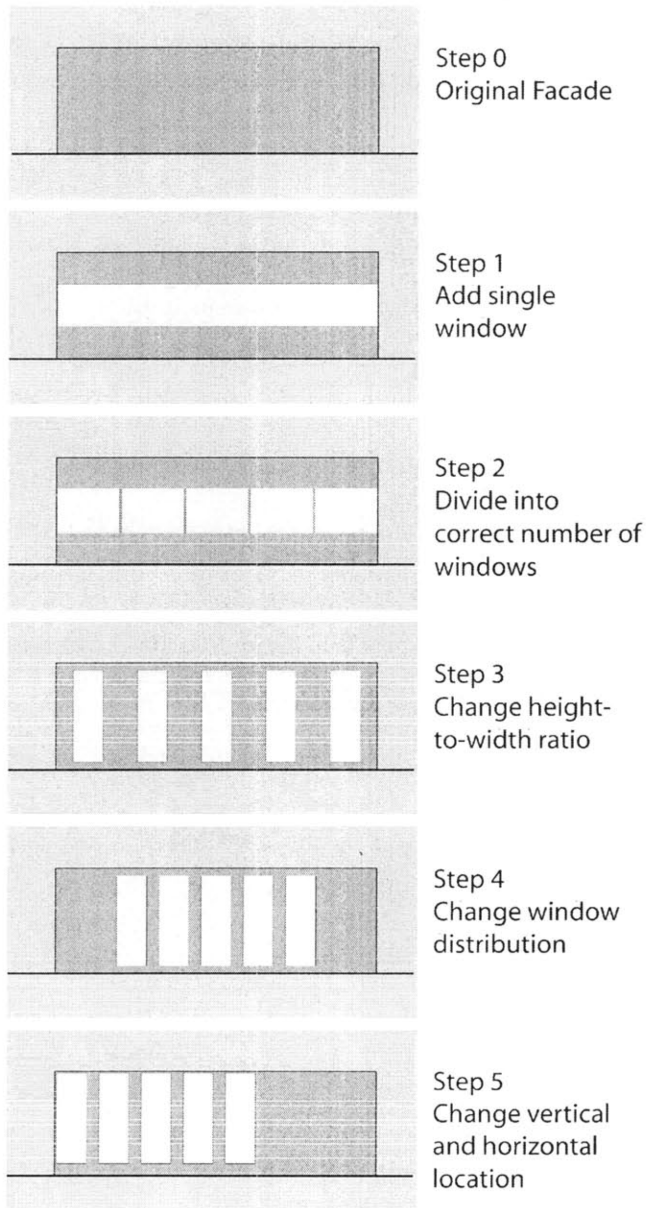


Figure 7.1: First five steps used to generate models in the micro-GA system

1. Add a single window of the given window-to-wall ratio (window area) to the facade using the width 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 [0.05 m]). 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 geometric 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. This limitation is the same as that of the expert system.

7.2.3 Case Study Procedures

A set of study procedures was developed to more appropriately compare results from the expert system to the GA, given their differences in algorithm type. While a GA is one that generates designs, the expert system always assumes that an initial design is given and suggests design changes based on the current design. The following procedure was used for each algorithm type:

Micro-GA

An initial massing model with no windows was used to create a new model of each design generated by the micro-GA. The algorithm was run either until a perfect solution was found or for ten generations before stopping. If a perfect solution was not found, the best design was considered that with the highest performance found over all generations.

Expert system

An initial model was created with generic rectangular windows. To avoid starting out with a design whose performance was very poor or very good, all initial models were designed to have mediocre performance (between 40% and 60%). For these case studies, a "perfect user" scenario was used. The "perfect user" was defined as one who would select the

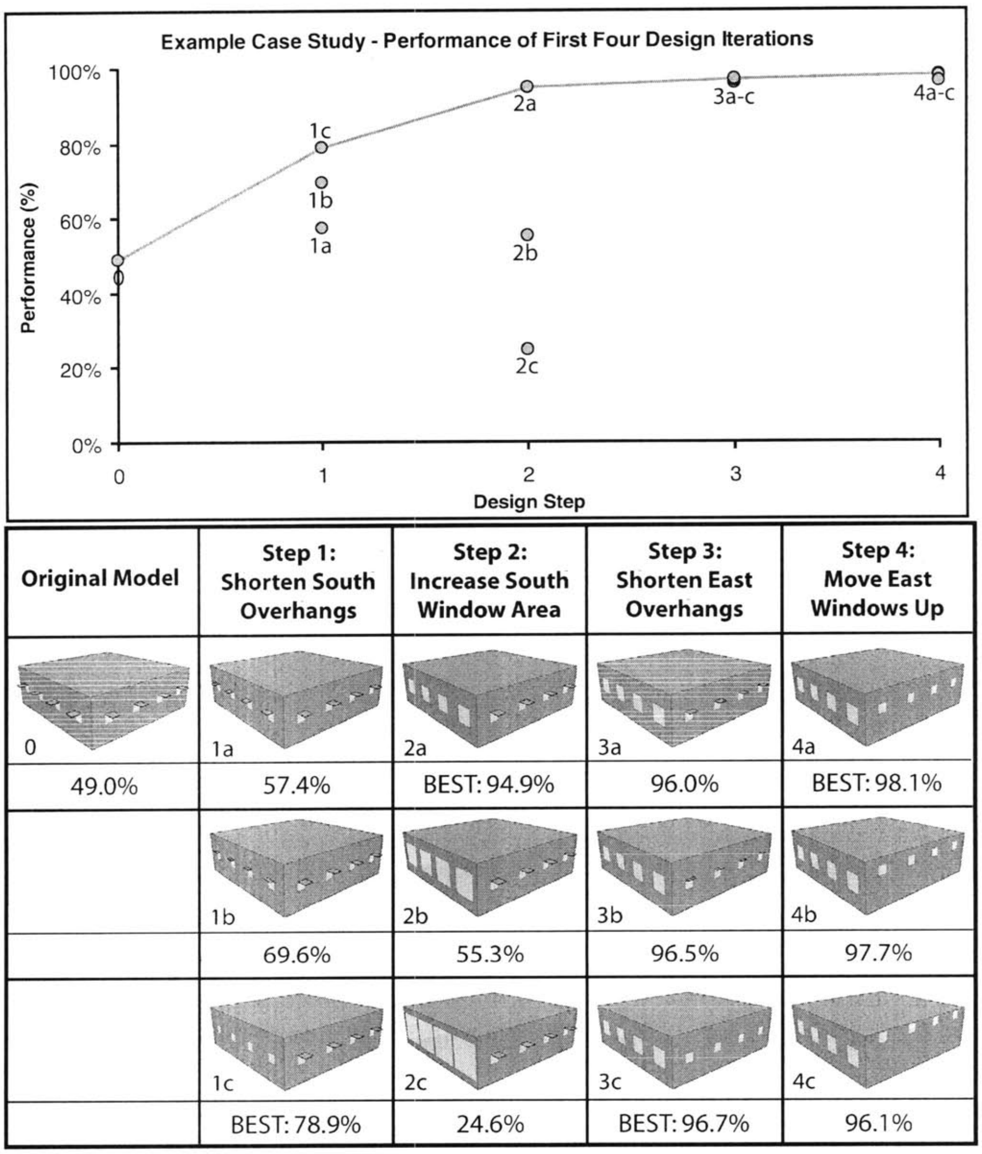
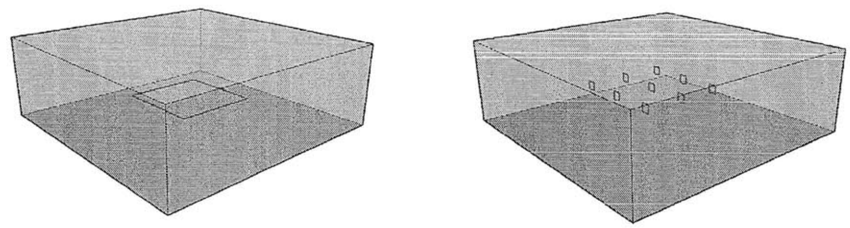


Figure 7.2: The performance and designs of the first four design steps of an example problem. The performance goal considered is a wide illuminance range (described further as case study #4). The "perfect user" selections are 1c, 2a, 3c, and 4a.



(a) Massing Model with Illuminance Sensor (b) Massing Model with Glare Sensors

Figure 7.3: Box massing model for case studies 1 through 5 with (a) illuminance sensor in the core zone or (b) glare sensors in the core zone with a view towards south

first suggested design change at each iteration and the best performing magnitude of each design change. The "perfect user" scenario was also one in which the process was allowed to continue even if performance decreased after a given design iteration. The algorithm was run until a perfect solution was found or for ten design iterations before stopping. As with the GA study, if a perfect solution was not found, the best design was considered that with the highest performance found over all completed iterations. The first four steps of an example case study (which will be discussed further as case study #4) are indicated in Figure 7.2, along with the performance and look of each generated model. In this example, for the "perfect user" scenario, the models chosen at each step were: 1c, 2a, 3c, 4a. After four steps, the best performing design found is 4a.

For the case studies which involved multiple conflicting goals, it was not possible to select a "best" performing design from either the GA or the expert system. In these cases, an approximate Pareto front was created by the multi-objective GA to demonstrate the range of possible designs and their performances for each of the conflicting goals. The design produced by the expert system were compared with those along the approximated Pareto front.

For all case studies, the window uniformity scheme selected for the expert system was "uniform facades." This selection ensured that the designs generated by the expert system would be comparable to those generated by the micro-GA, as the micro-GA was implemented such that all windows on the same facade would have the same properties (area, aspect ratio, glazing type, and shading devices). A short study which considers the effect of allowing the expert system to create non-uniform facades is described in section 7.3.2.

7.2.4 Single Performance Goal Case Studies

This section describes five simple case studies which each have a single performance goal. Each of these five case studies considers a box model with the same dimensions as the base model used to create the knowledge base results (described in section 4.2.4). For

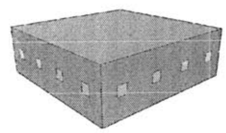
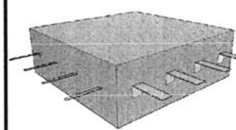
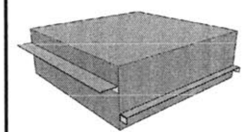
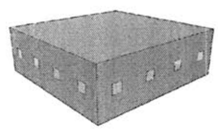
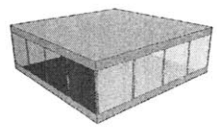
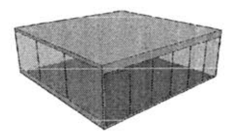
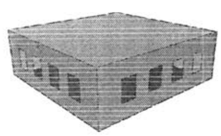
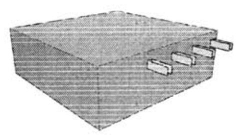
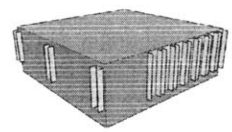
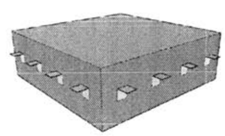
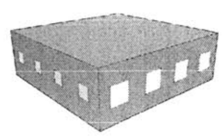
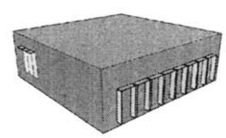
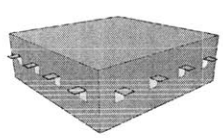
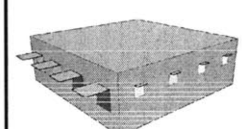
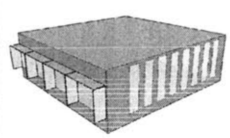
Performance Goal	**Expert System Starter Design**	**Expert System Best Design**	**Micro-GA Best Design**
Case Study #1 Maximum Illuminance Only	 50.5% In Range	 100.0% In Range	 100.0% In Range
Case Study #2 Minimum Illuminance Only	 49.5% In Range	 100.0% In Range	 99.4% In Range
Case Study #3 Maximum Glare Threshold	 60.1% In Range	 100.0% In Range	 98.0% In Range
Case Study #4 Wide Illuminance Goal Range	 49.0% In Range	 99.8% In Range	 99.9% In Range
Case Study #5 Narrow Illuminance Goal Range	 49.2% In Range	 94.4% In Range	 97.3% In Range

Figure 7.4: Comparison of best performing final designs from the expert system and micro-GA for case studies #1 through #5

the illuminance case studies, a single illuminance sensor plane was modeled in the core zone of the space (Figure 7.3a). For the glare case study, an array of glare sensors was modeled in the core zone of the space with a view towards south (Figure 7.3b). The first three case studies consider a single threshold (a maximum or minimum allowable value for illuminance or glare). The final two case studies consider illuminance ranges (where the allowable values include both maximum and minimum values). All case studies were considered for Boston, MA.

The first two case studies were considered validation studies for the micro-GA implementation. In these studies, three different GA runs were completed, resulting in three different final solutions. These two studies demonstrated that the micro-GA was performing as expected. In subsequent studies, only a single GA run was completed.

The results of the first five case studies are shown in Figure 7.4. This chart indicates the performance and the design of three models in each case study: the initial design used by the expert system (a generic design with mediocre performance), the best design found by the expert system, and the best design generated by the micro-GA.

7.2.4.1 Case Study #1: Maximum Only Illuminance Goal

The first case study had a single performance goal for illuminance, where the maximum illuminance values were 200 lux (desired) and 400 lux (acceptable); no minimum values were specified. All seasons and periods of day were considered. To verify that at least one design within the solution space was able to meet the desired performance goal, the author tested a small number of models manually. Although the author did not examine the entire solution space, more than one "perfect" solution were found. The designs which were found to meet the goal criteria all shared the characteristics of small window areas with large shading devices. This case study was considered the most basic because many designs within the search space meet the illuminance goal range (including a box with no windows).

Because this study was one of two validation studies for the micro-GA implementation, three micro-GA runs were completed. The micro-GA was highly successful at determining solutions to this problem, finding a "perfect" solution on each of the three separate runs, each time within 10 generations (Figure 7.5). The best performing designs from each of the three runs are shown in Figure 7.6. Because the solution space was known to be highly multi-modal, it is not surprising that the three solutions found all met the goal criteria yet all had different forms. The different final solutions are due mainly to the fact that each run began with a different random initial population, although the probabilistic nature of the algorithm contributes to this diversity as well. It is possible that additional runs would have produced even more varied results, although it is likely that any additional solutions would feature similar characteristics of small window area with large shading devices.

For the comparison case study, the expert system started with a generic design with a performance of 50.5% within the desired illuminance range, averaged over all times and

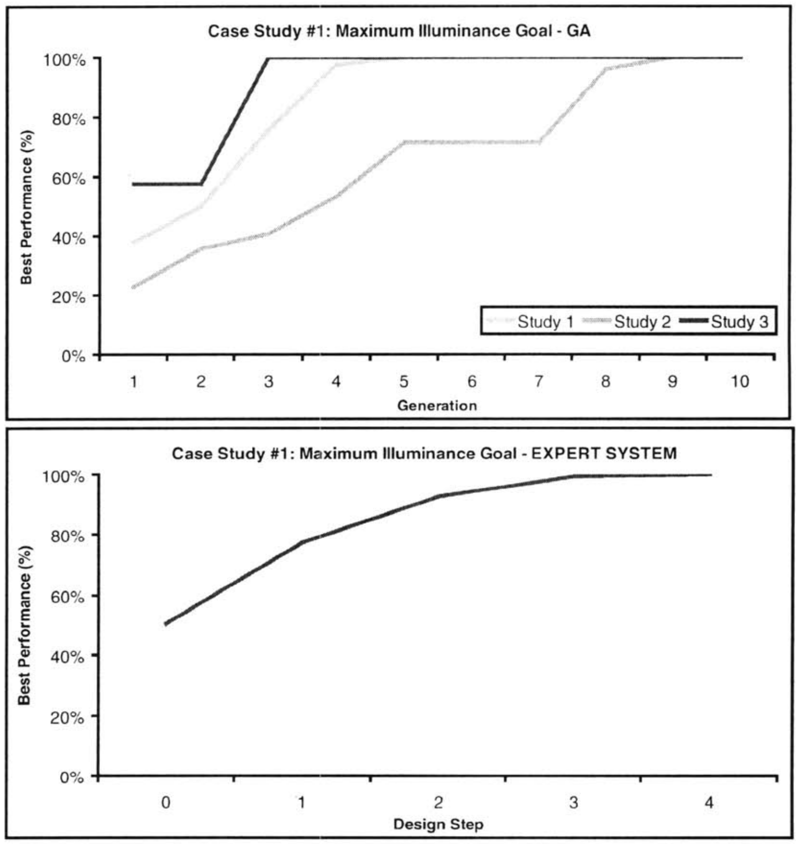


Figure 7.5: Case study #1 - Maximum illuminance goal: Performance for the micro-GA and expert system

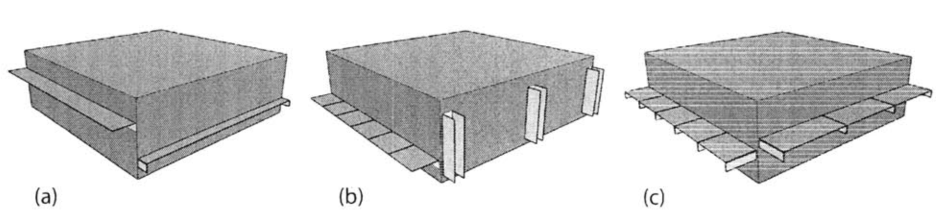


Figure 7.6: Case Study #1 - Maximum only illuminance goal: Best performing designs from three micro-GA runs

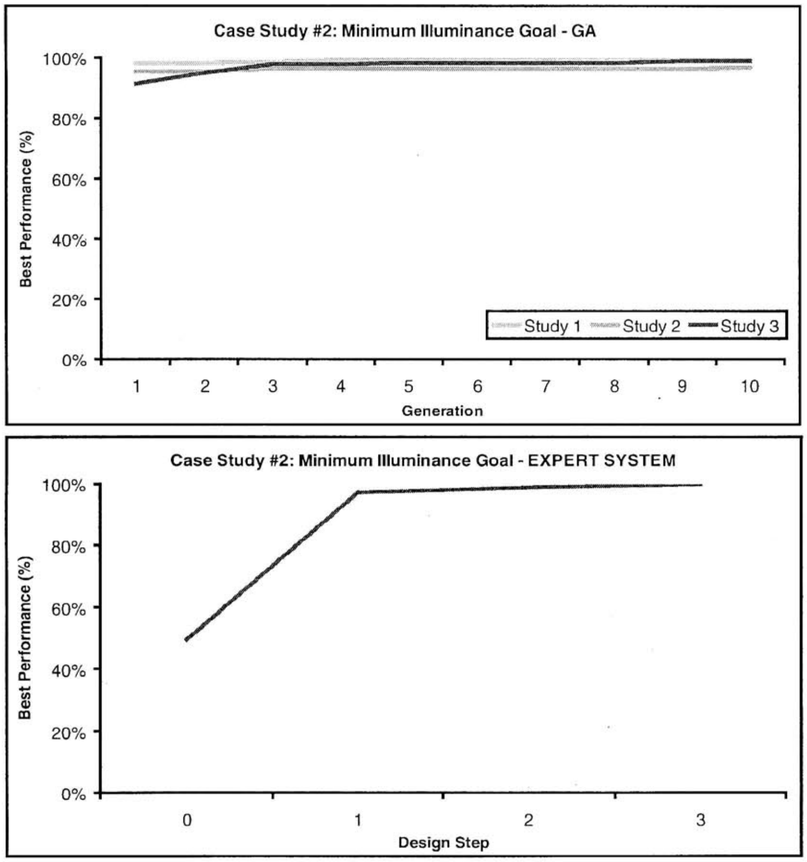


Figure 7.7: Case study #2 - Minimum illuminance goal: Performance for the micro-GA and expert system

seasons. From this initial design, the expert system was able to find a "perfect" solution very efficiently, within only 4 design iterations. As indicated in Figure 7.5, the performance of the expert system design improved at each design iteration, i.e. the expert system never proposed design changes which made the performance worse in this case study.

The initial and final designs of the expert system are shown in Figure 7.4. It is clear from this chart that the design generated by the expert system resembles its starting point design more than it resembles the design generated by the micro-GA. However, it is clear that both the expert system and the micro-GA found perfect designs for the maximum illuminance goal scenario which matched the known solutions, with large shading devices and a small window area.

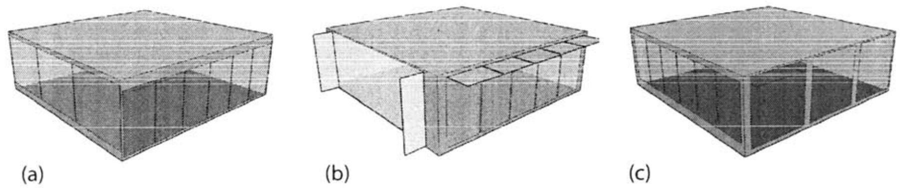


Figure 7.8: Case study #2 - Minimum only illuminance goal: Best performing designs from three micro-GA runs

7.2.4.2 Case Study #2: Minimum Only Illuminance Goal

This case study used the same model and sensor plane as the previous study, but the goal in this problem was to obtain a desired 400 lux minimum (200 lux minimum accepted) with no maximum values. Again, all seasons and periods of day were considered. As with the previous case study, the author manually searched a section of the solution space before running the GA to verify that at least one design which met the goal criteria existed. In the small subset of models examined, the author was only able to find one "perfect" solution, although it is possible that more solutions exist. In this case, the known solution featured a large window with a high glass transmissivity and no shading devices. This case study was considered more difficult than the previous one because there were fewer known solutions within the search space.

In this case study, the micro-GA came very close to finding the known solution after ten generations but never found one in which the illuminance goals were met over 100% of the sensor plane area and over the whole year (Figure 7.7). The most successful of the three final solutions generated has the same features as the known solution. All three solutions feature very large window area (Figure 7.8); however, it is interesting to note that the second solution got "stuck" in a search space that only included shaded designs, so the final solution in this case is the worst performer of the three. It is likely that this solution would have become "unstuck" if the algorithm had been allowed to continue running for more generations, which would have introduced new random solutions into the population upon convergence. Nevertheless, all three final designs found good designs after only a few generations, indicating that the micro-GA was again successful at efficiently converging on good designs.

For the comparison case study, the expert system again began with a generic design of mediocre performance. The initial design had a performance of 49.5% within the illuminance goal range, averaged over all times and seasons. As with the first case study, the expert system was able to find a "perfect" solution very efficiently, within only three design iterations (Figure 7.7), and it was able to suggest design changes which resulted in improved performance at every design iteration. The final design generated by the expert system has the same characteristics as the known solution and as the best performing micro-GA design (Figure 7.4).

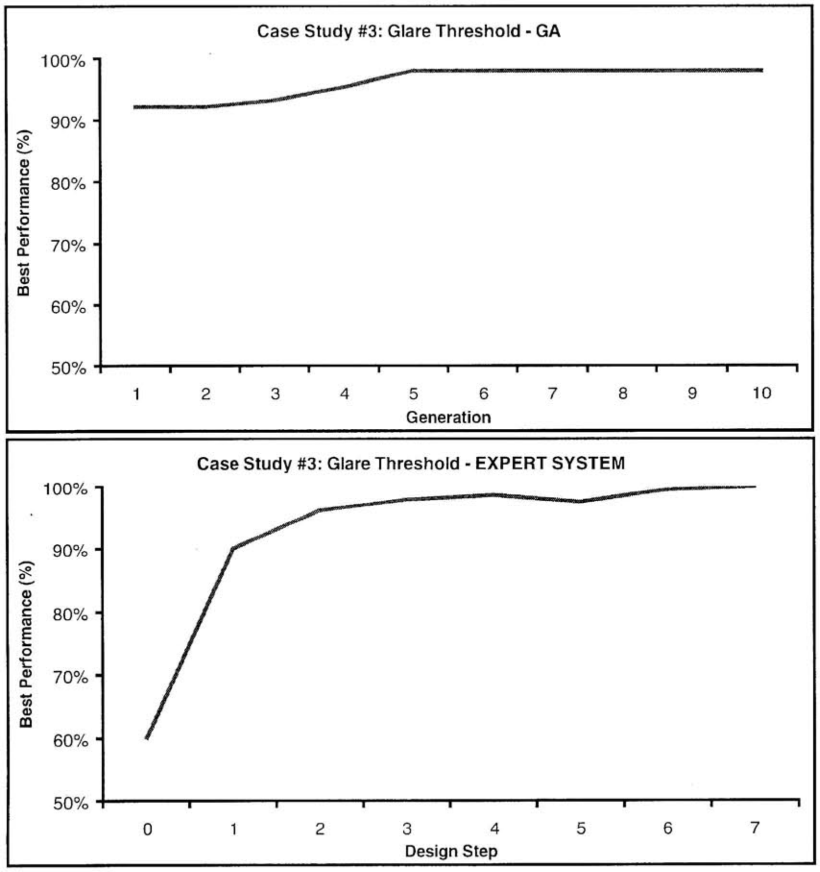


Figure 7.9: Case study #3 - Glare threshold goal: Performance for the micro-GA and expert system

7.2.4.3 Case Study #3: Glare Only Illuminance Goal

In this case study, the same box model was used as with the previous two studies, but the performance metric considered was glare instead of illuminance. An array of glare sensors was modeled in the core zone of the model facing south (Figure 7.3b) and the performance goal for this glare sensor group was to keep glare below the "medium" threshold, i.e. below the value of DGP that corresponds to perceptible glare.

In this case study, the micro-GA generated a design that was 98% in range after ten generations were run. The expert system found a solution that was 100% in range after seven design iterations. Both final designs are shown in Figure 7.4. While the designs generated by the expert system and the micro-GA look very different from each other, they do share some similar characteristics in that they both have small windows with shading devices, and windows clustered towards the north end of the east-facing facade.

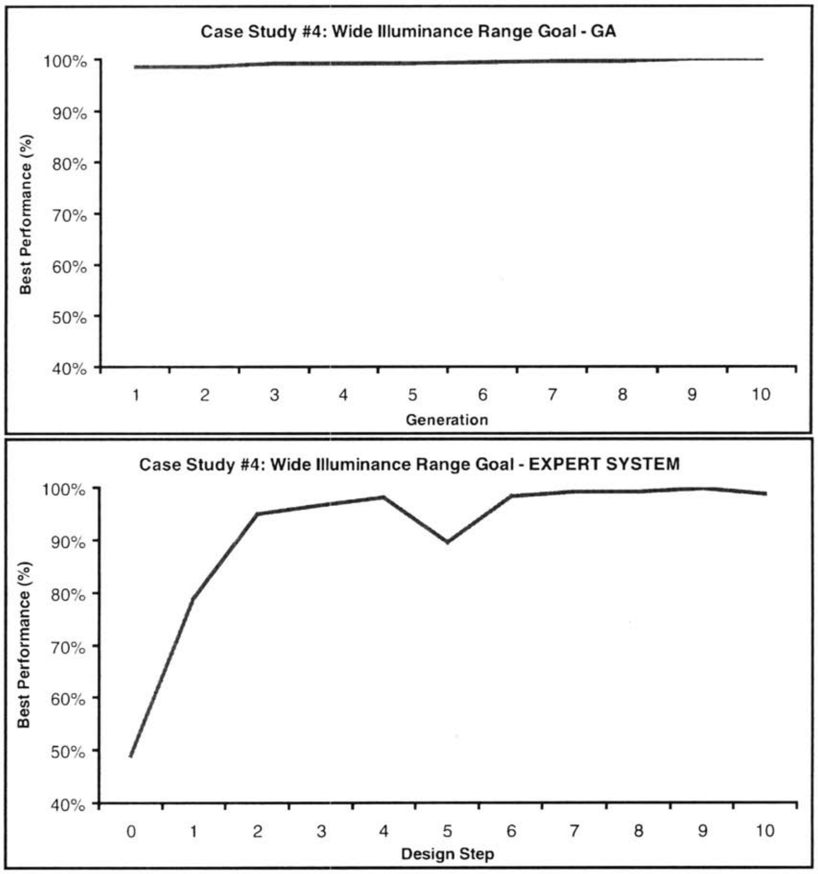


Figure 7.10: Case study #4 - Wide illuminance range goal: Performance for the micro-GA and expert system

The progression of performance for both algorithms is shown in Figure 7.9. In this case study, the expert system did suggest design changes that resulted in slightly lower performance at design iteration #5. However, in general, the performance improved almost continuously through the expert system process.

7.2.4.4 Case Study #4: Wide Range Illuminance Goal

The fourth case study considers a single performance goal with an illuminance range: 300 lux minimum preferred (100 lux accepted) and 1500 lux maximum preferred (2500 lux accepted). This is a more difficult goal to achieve than the previous illuminance case studies, in which only a maximum or minimum illuminance value were considered as a performance goal. However, this is still a relatively simple problem to solve, given that the range of acceptable illuminance values is fairly wide. To make this case study slightly more specific, only the school schedule was considered (morning through mid-day, autumn through spring).

In this case study, it is interesting to note that the micro-GA generated an extremely good design in the very first generation (98.5% in range for the times and seasons considered). After ten generations, the micro-GA algorithm was able to find a solution that was essentially "perfect" (99.9% in range). This design is shown in Figure 7.4. The expert system was also able to find a near-perfect solution (99.8% in range) after ten design iterations. Both final designs feature smaller windows on the south facade and larger windows on the east facade, and both designs have small or no shading devices on either facade.

In this case study, it is interesting to note the progression of performance of the expert system (Figure 7.10), as performance takes a significant drop after the fifth design iteration, although the system is ultimately able to correct this error. This drop is a result of the system working within an illuminance range, instead of towards a maximum or minimum: the system determined that the illuminance on the sensor plane was too low, so it proposed a change to increase illuminance, but the resultant increase pushed the illuminance values outside the desired performance range. This behavior of the expert system is more apparent for strict illuminance ranges (such as that considered in the next case study).

7.2.4.5 Case Study #5: Narrow Range Illuminance Goal

In the previous example, a wide illuminance goal range was considered. In this case study, the performance goal is a narrow illuminance range: 300 lux minimum preferred (100 lux accepted) and 800 lux maximum preferred (1200 lux accepted). Because the illuminance range is more strict than the previous case study, the problem is more difficult to solve. Like the previous case study, a school schedule was considered for this problem.

In this case study, the micro-GA was able to find a design with excellent performance (97.3% in range, for the times and seasons considered) after 10 generations (Figure 7.11). The expert system was also able to find a design with very good performance (94.4% in range), but as indicated in Figure 7.11, the expert system behavior exhibited a see-saw effect as it tried to work within the strict illuminance range. The system did not necessarily always suggest changes which improved the performance of the design, but it was able to correct itself successfully when the performance became too low. The final designs produced by the two algorithms both have large shading devices on the south facing windows; however, the east facades are visually very different. This difference may be the cause of the 3% difference in performance between the final designs found by the two systems.

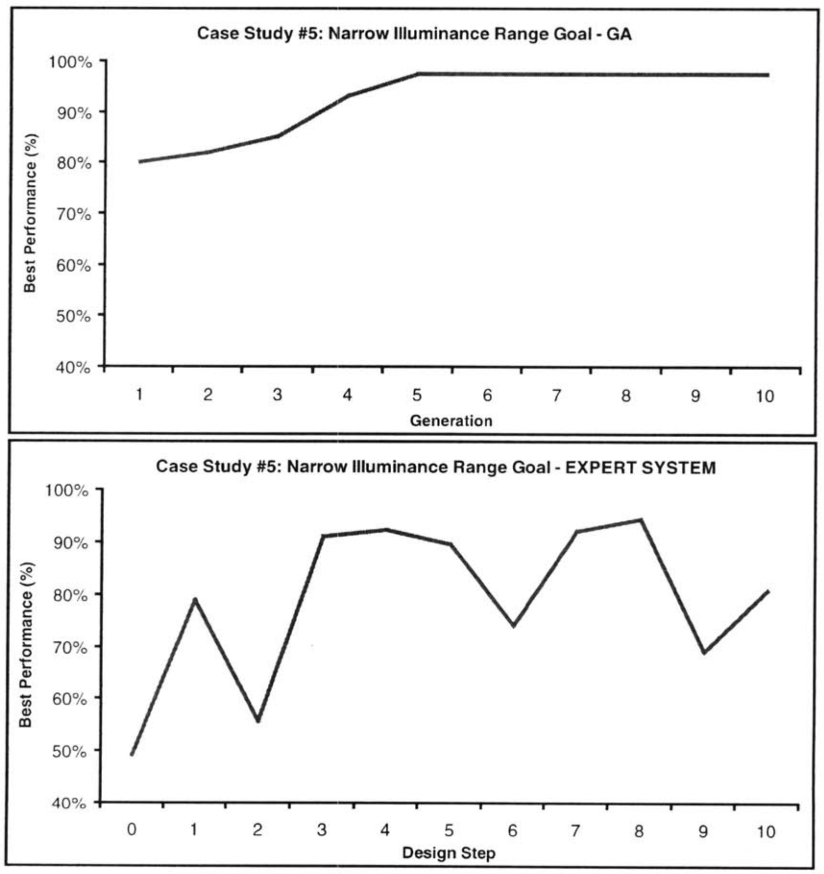


Figure 7.11: Case study #5 - Narrow illuminance goal: Performance for the micro-GA and expert system

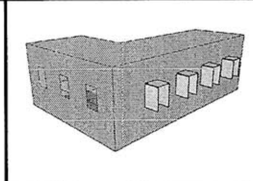
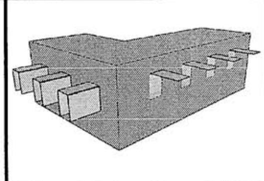
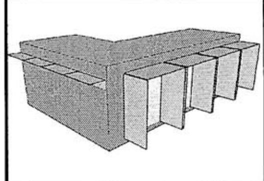
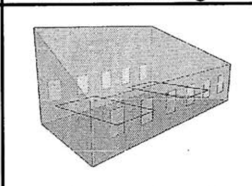
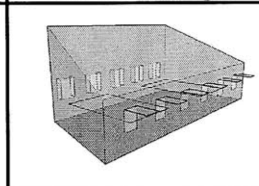
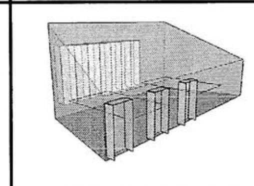
Performance Goal	Expert System Starter Design	Expert System Best Design	Micro-GA Best Design
Case Study #6 Two Illuminance Goals: Sensors Parallel to Facades	 64.5% In Range	 96.1% In Range	 95.3% In Range
Case Study #7 Two Illuminance Goals: Sensors Perpendicular to Facades	 61.3% In Range	 82.6% In Range	 87.0% In Range

Figure 7.12: Comparison of best performing final designs from the expert system and micro-GA for case studies #6 and #7

7.2.5 Complex Case Studies: Multiple Performance Goals

This section describes four case studies which each have two performance goals. The first two case studies consider sets of goals which are considered non-conflicting because there exist design solutions which come reasonably close to meeting both goals at once. The final two case studies are both examples of conflicting goal scenarios, as there are no true solutions which meet both performance goals at once. All case studies were considered for Boston, MA.

For the first two case studies, the performances of both sensor planes are averaged together to simplify the scenarios into single-objective problems. In these cases, it is still possible to directly compare the performance of the final "best case" designs produced by the micro-GA and by the expert system. Three designs for these two cases are shown in Figure 7.12: the starting expert system design, the best design found by the expert system, and the best design generated by the micro-GA.

For the final two case studies, it is not possible to isolate a "best case" design due to the conflicting nature of the performance goals. For these studies, a multi-objective micro-GA was run for 50 generations to approximate the Pareto front. The expert system was run three different times: first, with equal priority for all goals, then two additional times where each of the two goals was given priority over the other. Designs generated during these three expert system runs were compared against the designs located along the approximated Pareto front to determine how well the expert system could deal with a conflicting goal situation.

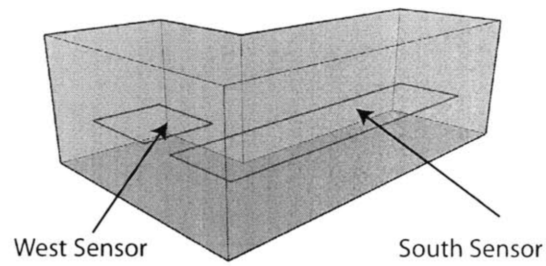


Figure 7.13: Case study #6 - L-shaped massing model with two illuminance sensors indicated

7.2.5.1 Case Study #6: Two Illuminance Goals – Sensors Parallel to Facades

The first non-conflicting goals case study features an L-shaped space with two illuminance goals. The two facades of interest are oriented towards the south and west, and the two illuminance sensors are located parallel to these facades (Figure 7.13). The illuminance goals for each sensor are:

- South zone: 400 lux minimum preferred (200 lux accepted); No maximum.
- West zone: No minimum; 500 lux maximum preferred (800 lux accepted).

Based on these goals, the known design solutions to this problem featured small, shaded windows on the west facade and larger windows on the south facade.

The performances of the micro-GA and the expert system system are shown in Figure 7.14. In these charts, the performance shown represents the average performance of the two illuminance goals. It is clear from the chart that the expert system proposes a design change which results in a significant decrease in performance at design iteration #4; however, the expert system is able to correct itself and find a design solution that is an average of 96.1% in range by the 8th design iteration. The micro-GA finds a similarly good solution with an average performance of 95.3% after ten generations. As expected, both "best" designs have either very small or highly shaded windows on the west facade with larger or less shaded windows on the south facade (Figure 7.12).

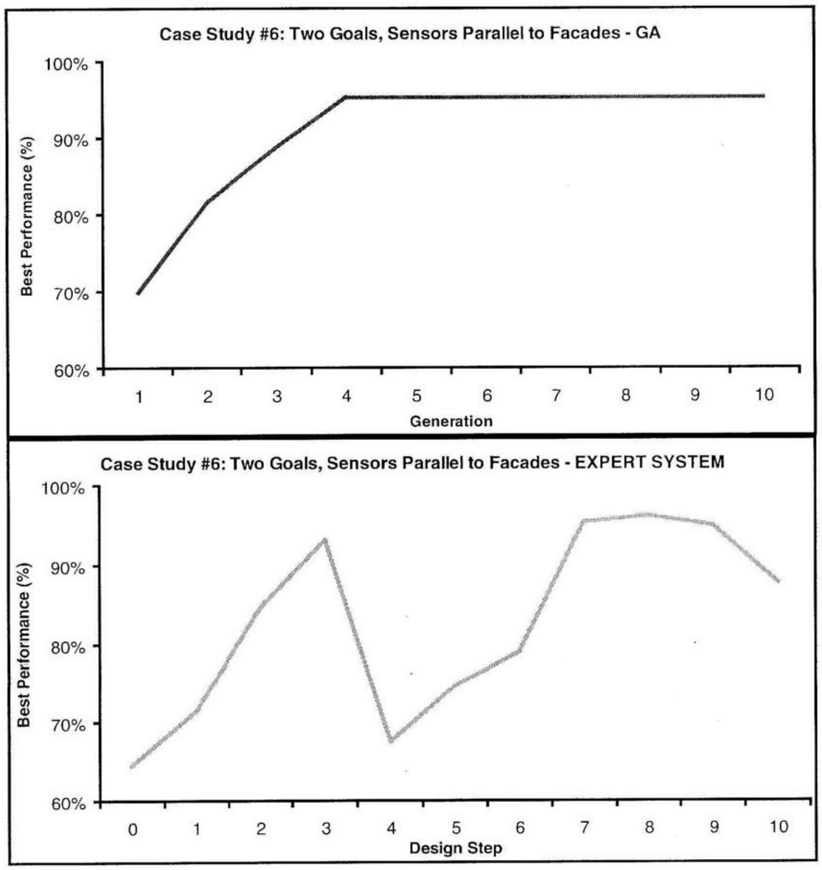


Figure 7.14: Case study #6 - Two illuminance goals with sensors located parallel to facades: Performance for the micro-GA and expert system

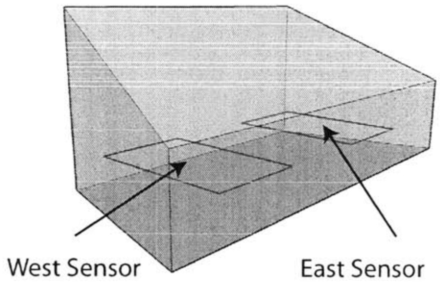


Figure 7.15: Case study #7 - Trapezoidal massing model with two illuminance sensors indicated

7.2.5.2 Case Study #7: Two Illuminance Goals – Sensors Perpendicular to Facades

The second non-conflicting goals case study features a trapezoidal space with a sloped roof. The two facades considered are oriented towards the south and north, and the two illuminance sensors are located perpendicular to these facades in the east and west ends of the space (Figure 7.15). In this case study, the height of the north facade is twice the height of the south facade. The illuminance goals for each sensor are:

- East zone: 200 lux minimum preferred (100 lux accepted); 800 lux maximum preferred (1200 lux accepted)
- West zone: 400 lux minimum preferred (200 lux accepted); No maximum.

The performances of the micro-GA and the expert system system are shown in Figure 7.16. As with the previous case study, these charts consider the average performance of the two illuminance goals. The micro-GA was able to find a solution with an average performance of 87.0% while the expert system's best design had an average performance of 82.6% (Figure 7.12).

It is evident from Figure 7.16 that the expert system struggled with this case study, exhibiting a see-sawing effect similar to that seen in the narrow illuminance range case study. This behavior is due to the window uniformity scheme selected for the expert system (all windows on the facade must be uniform) and the univariate ("step-by-step") nature of the expert system algorithm. While the micro-GA finds a design solution that features windows clustered towards the west end of both facades, the expert system focuses on changing the properties of the windows without moving them. The option to translate the windows towards one end of the facade does appear on the list of suggested design changes, but this option was never the first one presented to the user during the ten design iterations considered, so it was never chosen.

This type of design scenario is one in which the ability to change the window uniformity scheme will greatly enhance the performance of the expert system, as it would allow the

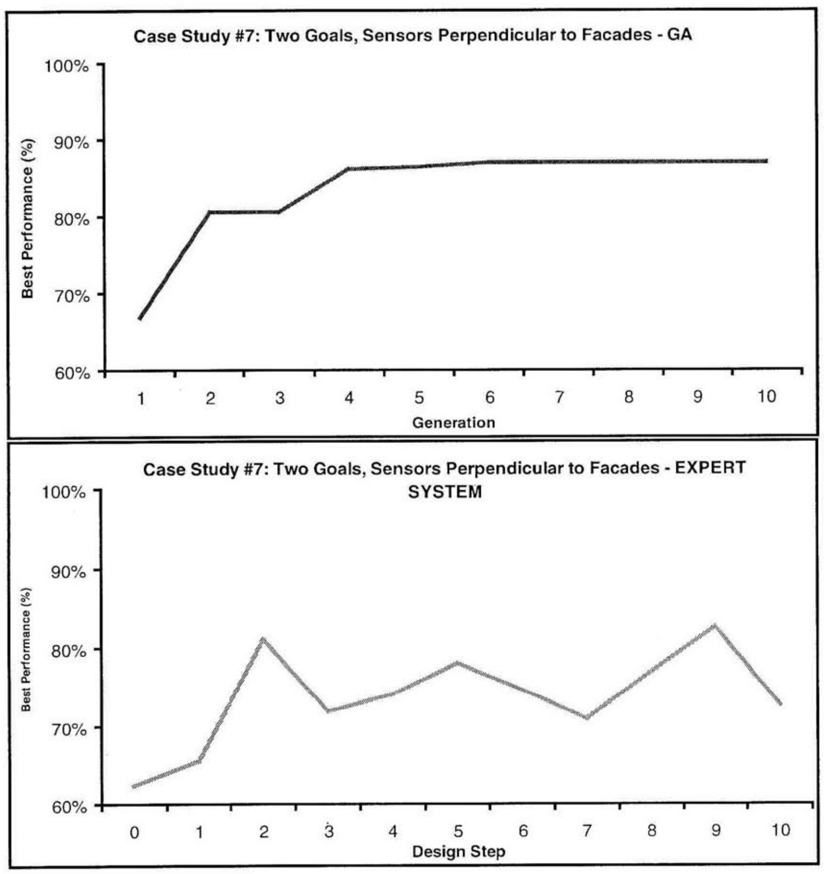
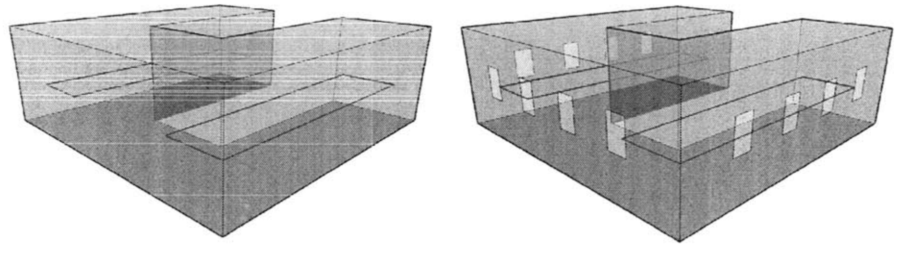


Figure 7.16: Case study #7 - Two illuminance goals with sensors located perpendicular to facades: Performance for the micro-GA and expert system



(a) Massing Model with Illuminance Sensors (b) Starting Model for Expert System

Figure 7.17: Case study #8 - U-shaped model for conflicting illuminance goals study: (a) Original massing model with sensors shown, (b) Generic starting model for the expert system

system to work only with the windows directly in front of a given sensor instead of working with all windows on each facade. The effect of changing the uniformity scheme for this case study will be discussed in section 7.3.2.

7.2.5.3 Case Study #8: Conflicting Illuminance Goals

The first conflicting goals case study has two illuminance goals, one with a goal range of low illuminance values and one with a goal range of high illuminance values. This case study considers a space with a U-shaped floorplan, and the three facades of interest face south, west, and north (Figure 7.17a). The two illuminance sensors planes are located in the north and south zones, and the illuminance goals for each sensor are:

- South zone: 1000 lux minimum preferred (800 lux accepted); No maximum.
- North zone: 200 lux minimum preferred (0 lux accepted); 500 lux maximum preferred (700 lux accepted).

This case study had an additional constraint that all facades must be uniform and that all three facades must be identical. This constraint ensured that the performance goals would be conflicting.

Because there cannot be a single "best" solution to a problem with conflicting goals for this case study, an approximated Pareto front was generated using the multi-objective micro-GA described in section 7.2.1. The approximated Pareto front demonstrates the range of possible solutions which are considered non-dominated. By examining the set of all non-dominated solutions, one can begin to understand the relationship between the two conflicting performance goals.

To compare the results of the expert system to those generated by the multi-objective micro-GA, the expert system was run three different times, each for five design iterations,

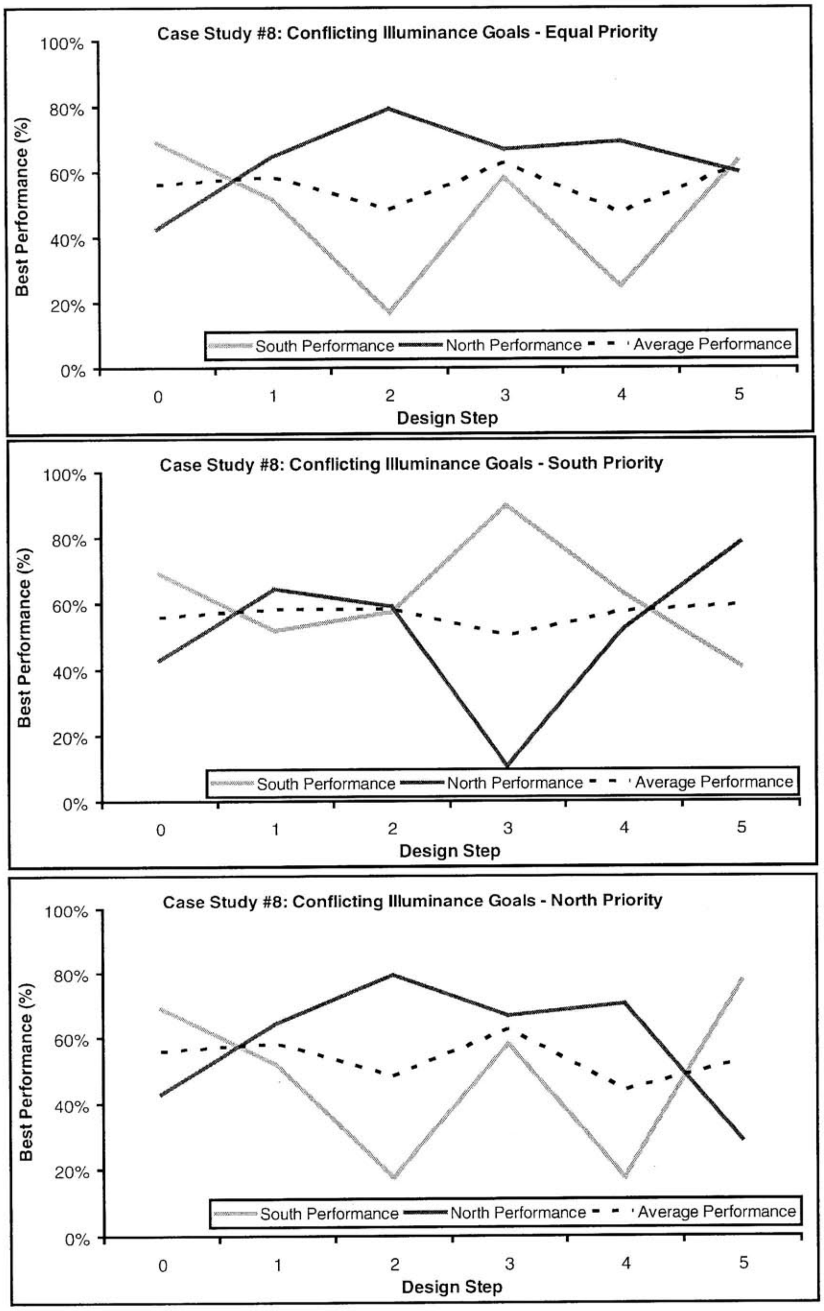


Figure 7.18: Case study #8 - Conflicting illuminance goals: Performance for the expert system for three different goal priority scenarios

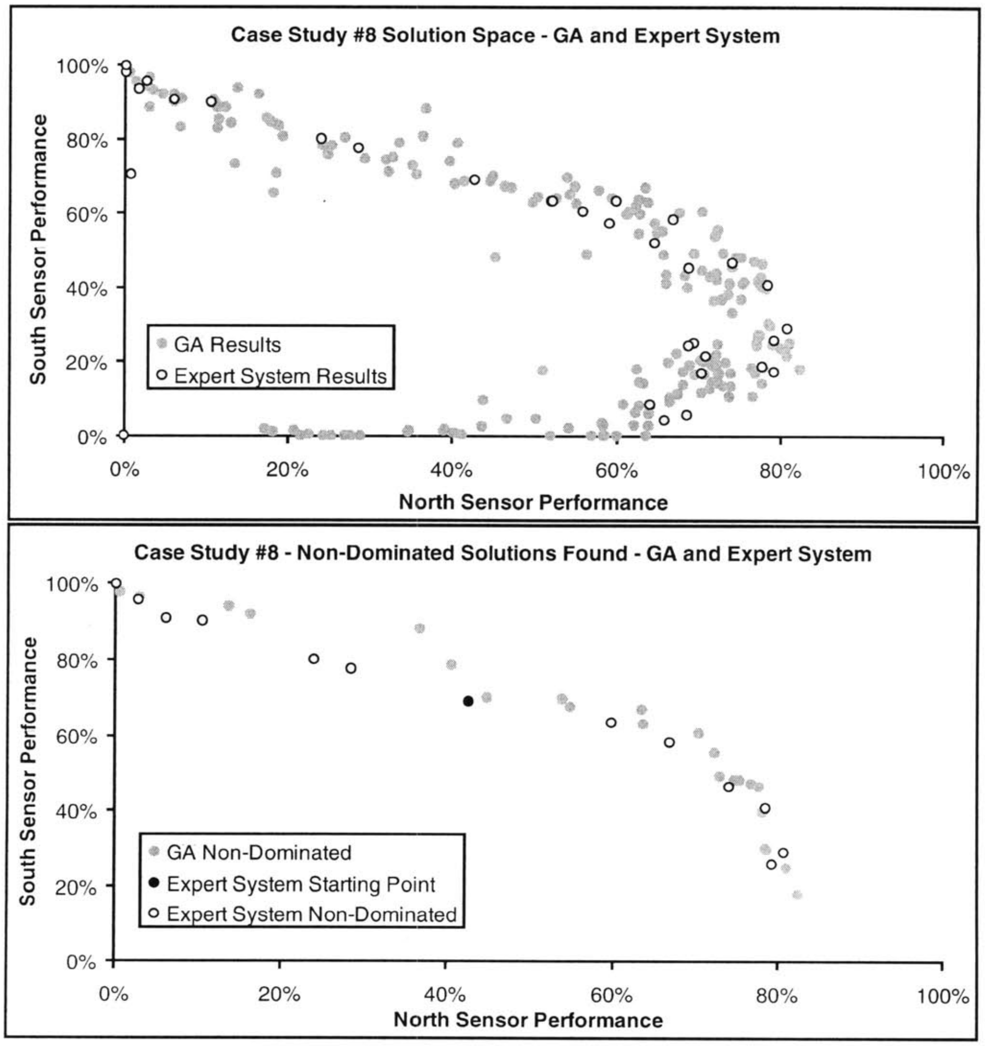


Figure 7.19: Case study #8 - Conflicting illuminance goals: Performance for the expert system for three different goal priority scenarios over the entire solution space (upper) and over the approximated Pareto front (lower).

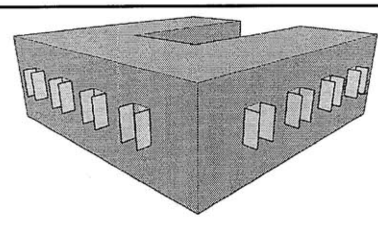
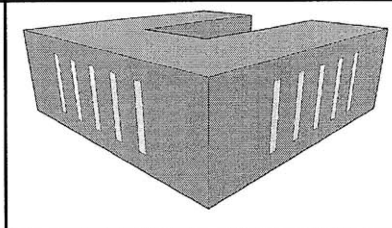
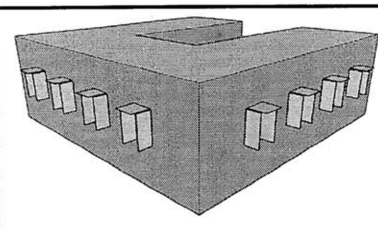
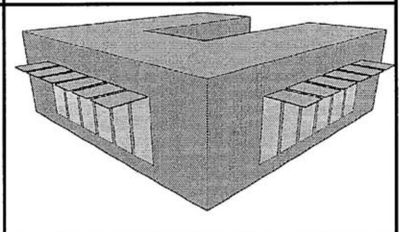
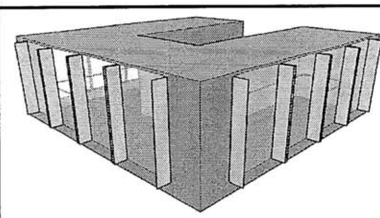
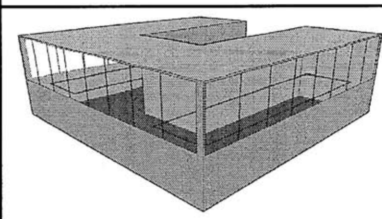
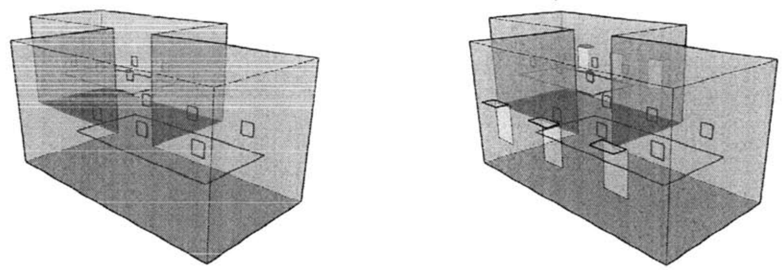
Case	Created by Expert System	Generated by Micro-GA
Best Average Performance	 **Average: 65.4%** North: 70.5% South: 60.5%	 **Average: 62.6%** North: 66.9% South: 58.2%
Best North Sensor Performance	 Average: 54.9% **North: 80.8%** South: 29.0%	 Average: 50.3% **North: 82.5%** South: 18.1%
Best South Sensor Performance	 Average: 49.9% North: 0.0% **South: 99.5%**	 Average: 49.2% North: 0.0% **South: 97.7%**

Figure 7.20: Case study #8 - Conflicting illuminance goals: Comparison of designs for best average, north, and south sensor performance for the expert system and the micro-GA



(a) Massing Model with Illuminance and Glare Sensors

(b) Starting Model for Expert System

Figure 7.21: Case study #9 - Z-shaped model for conflicting illuminance and glare goals study: (a) Original massing model with sensors shown, (b) Generic starting model for the expert system

with a different sensor priority given for each of the three runs. The performance of the expert system over each of these three runs is shown in Figure 7.18. The generic initial starting point model for the expert system is shown in Figure 7.17b.

As is clear from the charts in Figure 7.18, the conflicting goals presented a challenge for the expert system, which tended to see-saw from working towards one goal to working towards the other. This behavior resulted in the average performance staying relatively constant throughout the process. However, when all the designs generated by the expert system over the three runs are compared to those generated by the micro-GA over 50 generations (Figure 7.19), it is clear that the set of expert system designs do cover a wide area within the solution space and offer a way of approximating the Pareto front using fewer total simulations than those required by the micro-GA.

Although the expert system does not generate a Pareto front itself, it is interesting to compare three designs generated by each algorithm: the design with the best average performance, the design with the best north sensor performance, and the design with the best south sensor performance (Figure 7.20). The expert system was able to find designs with similar performance and somewhat similar design characteristics to those generated by the micro-GA in each of these cases. Such designs may provide the user with an understanding of the trade-offs between one performance goal and the other in a conflicting goal scenario.

7.2.5.4 Case Study #9: Conflicting Illuminance and Glare Goals

The second conflicting goals case study has one illuminance goal with a desired range of high illuminance values and one glare goal. The goals are conflicting because in this particular case, achieving the illuminance goal is likely to cause glare to increase for the views considered. This case study considers a Z-shaped floorplan, and the two facades of interest face east and west (Figure 7.21a). Two illuminance sensors planes are located

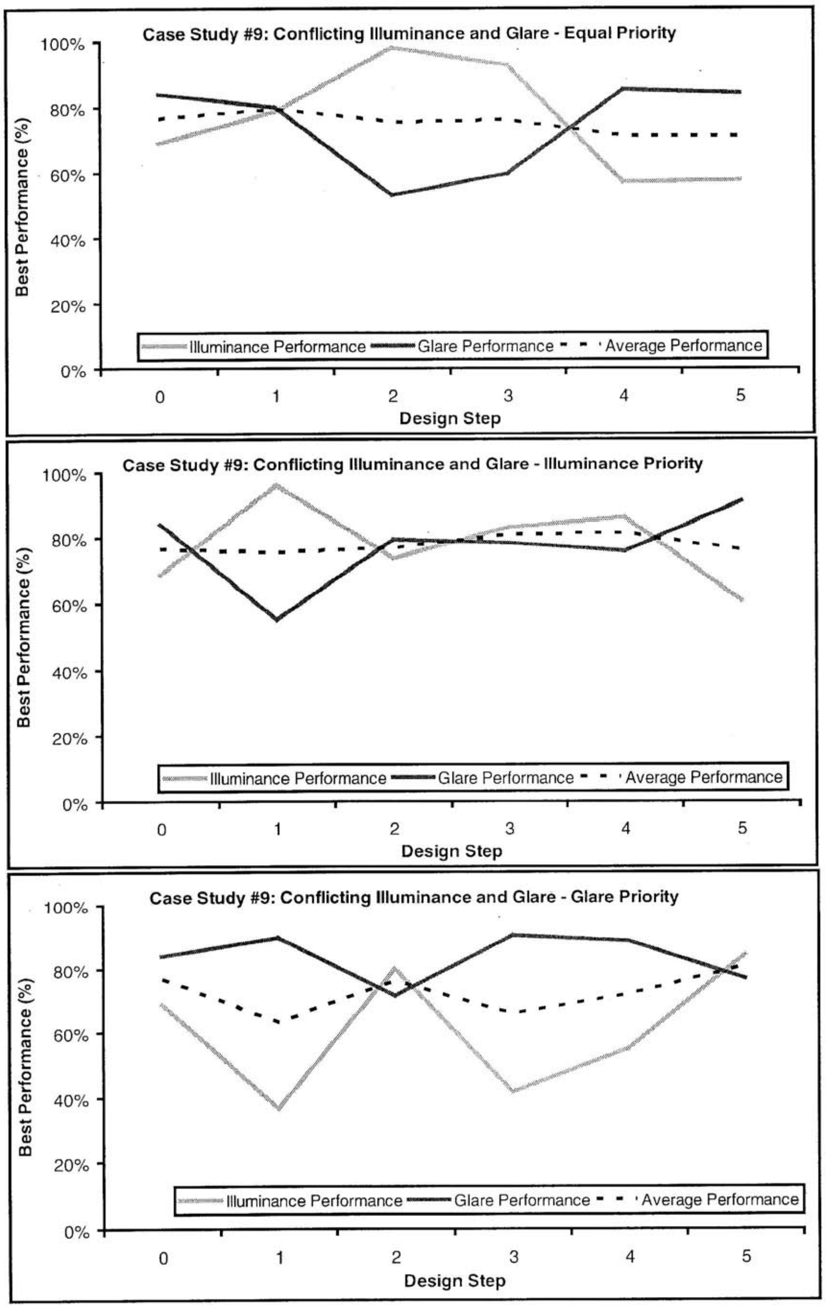


Figure 7.22: Case study #9 - Conflicting illuminance and glare goals: Performance for the expert system for three different goal priority scenarios

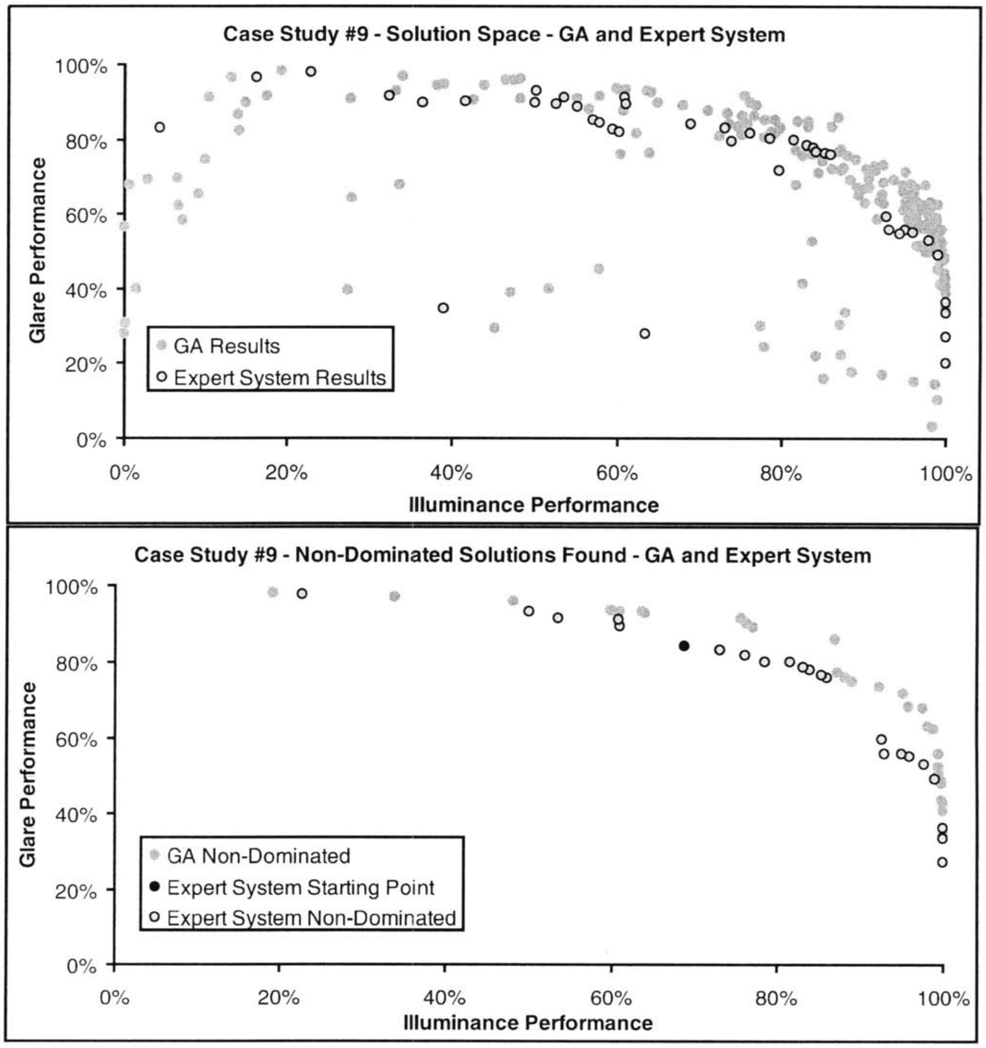


Figure 7.23: Case study #9 - Conflicting illuminance and glare goals: Performance for the expert system for three different goal priority scenarios over the entire solution space (upper) and over the approximated Pareto front (lower).

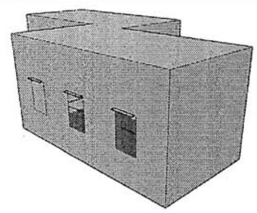
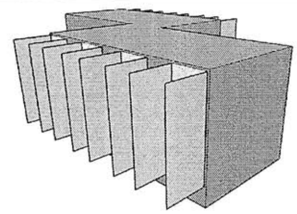
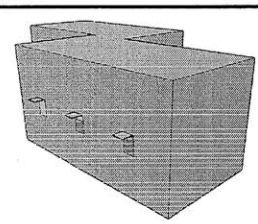
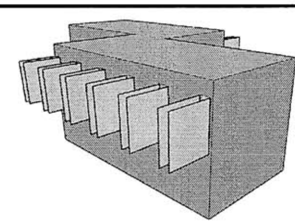
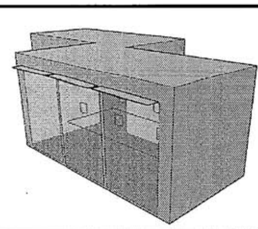
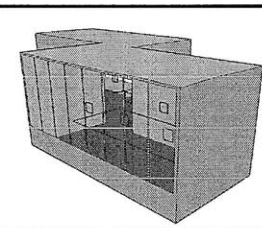
Case	Created by Expert System	Generated by Micro-GA
Best Average Performance	 **Average: 80.9%** Glare: 75.8% Illuminance: 86.0%	 **Average: 86.4%** Glare: 85.5% Illuminance: 87.0%
Best Glare Sensor Performance	 Average: 60.3% **Glare: 97.7%** Illuminance: 22.8%	 Average: 58.7% **Glare: 98.1%** Illuminance: 19.3%
Best Illuminance Sensor Performance	 Average: 63.6% Glare: 27.2% **Illuminance: 100.0%**	 Average: 67.2% Glare: 34.5% **Illuminance: 100.0%**

Figure 7.24: Case study #9 - Conflicting illuminance and glare goals: Comparison of designs for best average, glare, and illuminance sensor performance for the expert system and the micro-GA

in the east and west zones and glare arrays are located within the same zones with views facing outwards. The performance goals for the two sensor groups are:

- Illuminance: 200 lux minimum preferred (0 lux accepted); No maximum.
- Glare: Zero glare tolerance (only imperceptible glare allowed).

As with the previous case study, this case study had an additional constraint that all facades must be uniform and that all facades must be identical to ensured that the performance goals would be conflicting.

The behavior of the expert system during this case study displays a similar see-saw effect as the previous case study (Figure 7.22). Again, the three expert system runs produce a set of designs that provide a good representation of the solution space and of the Pareto front for this problem (Figure 7.23).

A comparison of three designs (the design with the best average performance, the design with the best glare sensor performance, and the design with the best illuminance sensor performance) is shown in Figure 7.24. In this case study, the micro-GA was able to find a design with an average performance that is over 5% better than the design found by the expert system, and in general, it is clear from Figure 7.23 that the expert system designs tended to have slightly lower glare performance than the micro-GA designs in the middle area of the Pareto front. However, the expert system is still able to effectively provide the user with a rough approximation of the Pareto front and a set of designs that explores the trade-offs between illuminance and glare performance.

7.2.6 Comments on the Stopping Criteria

For the case studies described in sections 7.2.4 and 7.2.5, both the expert system and the micro-GA were run until a "perfect" solution was found or for ten generations or design iterations. The maximum number of design iterations was chosen to be ten based on initial runs which indicated that the best performing designs for the expert system case studies were found relatively quickly and within ten design iterations. In the case studies presented in this section, the best designs were found after the following number of design iterations:

1. Maximum only illuminance goal: 4 Iterations (100%)
2. Minimum only illuminance goal: 3 Iterations (100%)
3. Glare only goal: 7 Iterations (100%)
4. Wide illuminance range goal: 9 Iterations (99.8%)
5. Narrow illuminance range goal: 8 Iterations (94.4%)

6. Two illuminance goals - Sensors parallel to facades: 8 Iterations (96.1%)
7. Two illuminance goals - Sensors perpendicular to facades: 9 Iterations (82.6%)

The nature of the expert system is such that the fastest improvement is generally seen in the first few design iterations. After a certain number of design iterations have been completed, the design changes suggested by the expert system become less likely to improve performance. In each of the case studies considered, the performance of designs had already begun to worsen by the time the tenth design iteration was completed. If additional design iterations had been completed, it is likely that performance would have continued to decline.

This behavior is different from that of the micro-GA. Due to the elitist strategy used in the micro-GA, design performance will always improve or stay the same as more generations are completed. Although the micro-GA process was stopped after ten generations for the case studies considered, it should be noted that the micro-GA may have found slightly better performing designs if it had been allowed to continue for more generations.

As an example, the micro-GA was allowed to run for an additional 15 generations for case study #7 (trapezoidal design), which was the most difficult case study for both the micro-GA and the expert system. The original ten generation run produced a design that was an average of 87.0% within the goal range. After 25 generations, the micro-GA was able to generate a design with an average performance of 90.2%, an increase of 3.2% (Figure 7.25). However, this case study was the only one in which either the micro-GA or the expert failed to produce a design which performed higher than 94% after ten iterations or generations. Therefore, although the micro-GA may have been able to find slightly better designs if it had been allowed to run for more generations for the other case studies, the difference in performance could not have been higher than a few percentage points.

7.2.7 Discussion

The nine case studies discussed in sections 7.2.4 and 7.2.5 have demonstrated that the expert system can produce designs which perform similarly to those generated by a micro-GA for a variety of different design scenarios and problem types when a "perfect user" is assumed. The behavior of the expert system was found to differ based on the complexity of the problem, and the following trends were observed:

- For problems with a single goal threshold (for example, only a minimum desired illuminance value) such as case studies #1 through #3, the expert system was able to propose design changes which resulted in improved performance at almost every design iteration. The expert system was able to find "perfect" solutions to these problems.
- For problems with an illuminance goal range, the expert system displayed trends of see-sawing. For example, the expert system might note that the illuminance was

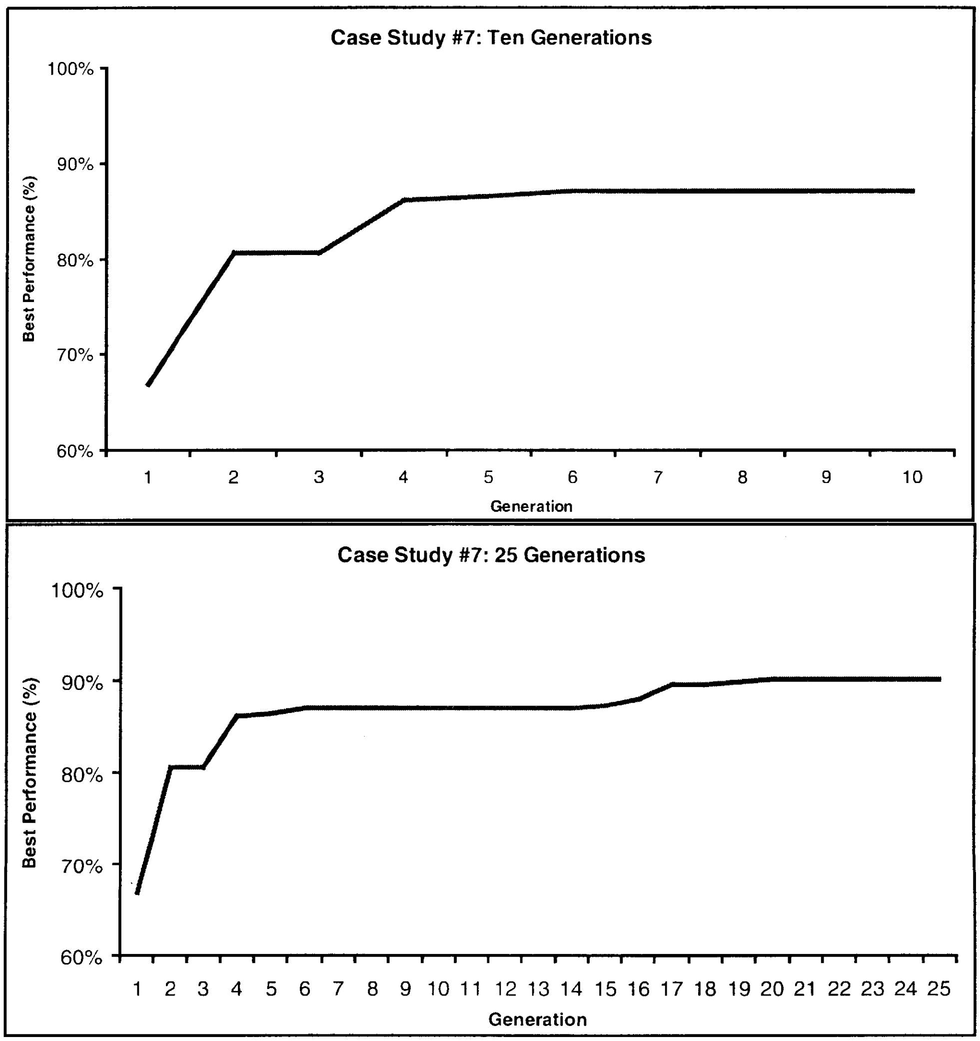


Figure 7.25: Micro-GA runs for 10 and 25 generations for case study #7

too low and propose a change to increase illuminance, and the subsequent changes might decrease performance by pushing the illuminance values past the maximum allowable illuminance threshold. The see-sawing trend was more evident for a narrow goal range (case study #5) than for a wide goal range (case study #4). However, the see-sawing behavior indicates that the expert system always corrects itself when it discovers that a previous suggested design change has decreased performance, so good performing solutions were still found for goals of this type.

- For problems with two non-conflicting goals such as case studies #6 and #7, the expert system may exhibit a similar see-sawing behavior as it tries to work with two different goals at once. Again, it was found that the system can effectively correct itself and that good solutions may be found using the system for these types of problems. The worse comparison case study was for case study #7, where the micro-GA was able to more effectively propose designs within the "uniform facade" constraint than the expert system. However, even in this case, the expert system was able to find a solution which performed within 4% of the performance of the best design generated by micro-GA (and within 7.5% if the 25 generation micro-GA run is taken into consideration).

- For problems with conflicting goals such as case studies #8 and #9, the expert system struggled to find solutions which met both goals at once. The behavior of the expert system is similar to that with non-conflicting goals but more extreme, as the process generally alternates from working towards one goal and working towards the other, which results in the average performance remaining relatively constant through all design iterations. Such behavior is not unexpected, given the nature of the problem and the fact that there is generally not a single perfect solution to a multi-objective problem. While a single "best" solution was not found, it was discovered that the expert system could produce a set of designs which provide a reasonable approximation of the Pareto front generated by the micro-GA. Such a set of designs may provide a designer with an understanding of the trade-offs between the two performance goals for a conflicting goal design scenario. In these types of problems, it is clear that the expert system has an educational potential, which will be discussed further in Chapter 8.

It is important to note that the purpose of the comparison study was not to directly compare the expert system algorithm to the micro-GA, as the two algorithms are inherently different. In these case studies, the micro-GA was used to verify that a reasonable solution existed for each case study by generating a very good (although not necessarily "perfect") benchmark solution for each problem. The purpose of the case studies was to evaluate the performance and behavior of the expert system and to confirm that the proposed design changes made by the expert system can improve performance for a variety of design scenarios. As it was shown that the expert system could find designs that were within an average of about 1% (and a maximum of 7.5%) of the performance of the best designs generated by the micro-GA in each case study, a potential user can have confidence in the advice given by the expert system and can assume that by making the suggested design changes, the performance of his or her design will improve over a series of design iterations.

Although the performance of the expert system was slightly lower than that of the micro-GA in some cases, this small degree of performance is sacrificed for the purposes of allowing user-interactivity into the expert system process. Although the studies presented in this chapter do not consider any user-interactivity, Chapter 8 will describe a user study which demonstrates that the expert system can improve the performance of designs even when human behavior is included in the process. In the user study, it was found that participants never acted as a "perfect user," and that the degree of improvement found by the expert system was influenced by the choices made by participants during the process. The value of the interactivity allowed during the expert system process will be discussed further in Chapter 8.

7.3 Effect of Initial Facade Constraints

While the previous set of case studies was able to demonstrate that the expert system can indeed produce designs that are comparable to those produced by a micro-GA, the behavior and performance of the expert system is also dependent on several variables that do not affect the micro-GA. One important difference between the micro-GA and the expert system is that the micro-GA generates its own designs while the expert system begins with an initial design and suggests changes to be made to that specific design based on its current performance. This feature was designed purposefully based on the expert system design process described previously in section 3.5. The best design produced by the expert system is thus highly dependent on the initial design given to the system.

The expert system also allows the user to select a uniformity scheme for the windows in his or her design. In the micro-GA comparison studies, both the micro-GA and the expert system designs were constrained to have uniform facades, which meant that all windows on a single facade had the same characteristics. However, the expert system also includes the option of having non-uniform facades. Selecting this option would allow the system to make changes to individual windows on a facade instead of all of them at once. For certain types of design scenarios, selection of the non-uniform window option will result in greater performance improvement than the uniform window option.

This section presents two brief case studies which examine the effects of the initial facade design and the window uniformity scheme on the overall improvement found by the expert system.

7.3.1 Specificity of Initial Facade Design

This study examines the relationship between the performance of the expert system and the initial facade design. For this study, the problems in the micro-GA case studies #4 and #5 are considered: a simple box model with an illuminance sensor in the core zone, facades on the east and south, and an illuminance range as a performance goal (Figure 7.3a). The two case studies differ as the performance goal for one is a wider illuminance goal range than the other, which is therefore an easier problem to solve. These case studies were chosen for comparison because they enable a comparison of the expert system behavior for different facade types on two different levels of goal difficulty. The two illuminance goal ranges are:

- Wide range: 300 lux minimum preferred (100 lux accepted); 1500 lux maximum preferred (2500 lux accepted).
- Narrow range: 300 lux minimum preferred (100 lux accepted); 800 lux maximum preferred (1200 lux accepted).

In the original case studies presented, the expert system process began with a relatively generic facade design with mediocre performance (around 50%). In this study, four different facade designs are considered. The four different starting facades are shown in Figure 7.27 and have varying levels of specificity: the first is the generic facade used for the micro-GA studies with square windows, the second has slightly more elongated windows, the third has extremely elongated windows, and the fourth has elongated windows clustered towards one end of each facade.

In this study, the expert system process was run for each of the four initial facade types and for each of the two illuminance goal ranges. The expert system was run for ten design iterations in all cases, and the "uniform facade" scheme was selected. The performance of the expert system for all cases is shown in Figure 7.26.

For the wide illuminance range scenario, it is clear that during the first four design iterations, the starting facade does affect performance: the most generic facade improves the most quickly while the more stylized designs see a lower amount of improvement. After all ten generations, however, it is seen that very good designs are found for all four starting designs, with performances ranging from 94.5% to 99.8%. There is one significant drop in performance for one of the four cases, but in general, the expert system displays similar behavior for all four starting models. The final models are shown in Figure 7.27 and it is clear that in each case, the final design is similar in aesthetics to the initial design.

In the narrow illuminance range case, an interesting phenomenon is seen as the see-sawing effect that was noted during the micro-GA case studies becomes smoothed out as the starting design becomes more stylized (Figure 7.26). This behavior is likely due to the fact that the expert system can make larger design changes to the generic facades than to the more stylized facades, based on the constraints of the original designs, and therefore it can explore a range of facade designs with a potentially wider range of performances. The expert

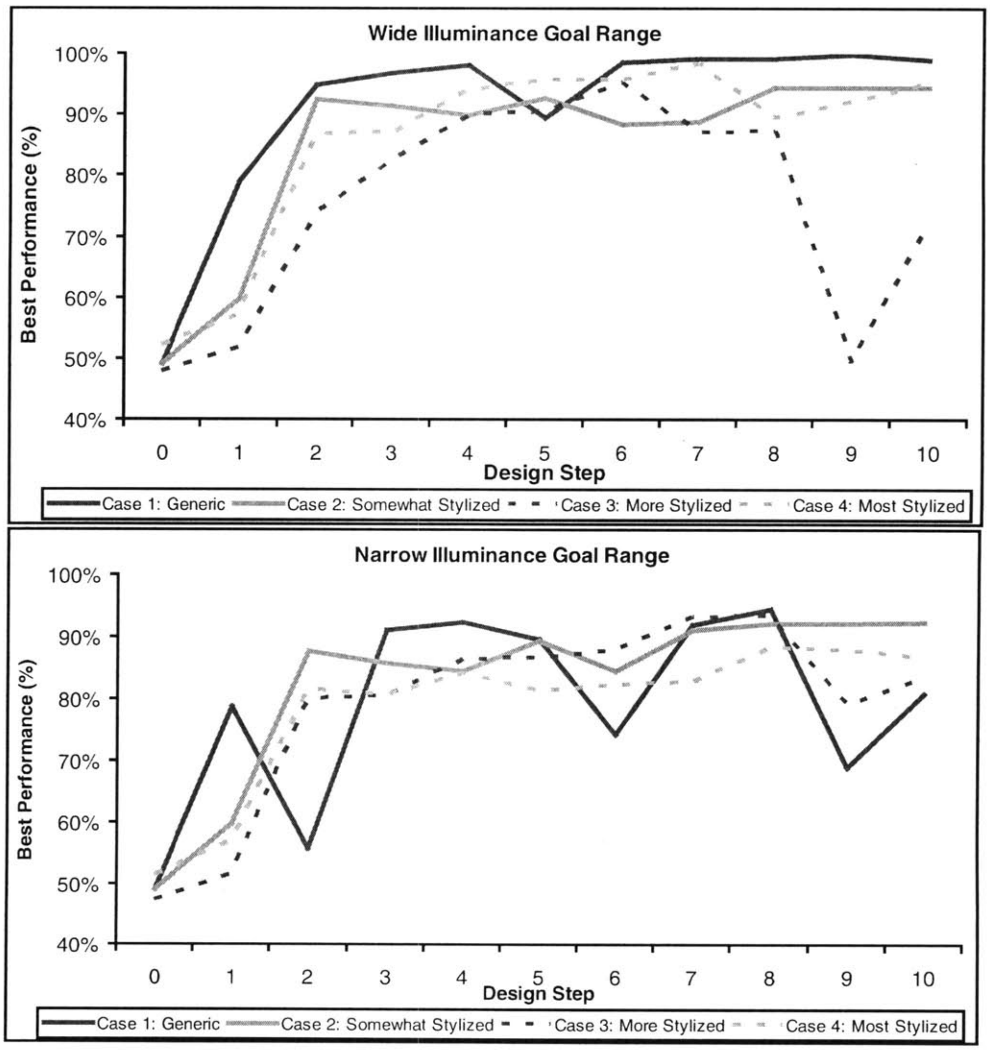


Figure 7.26: Performance over ten design steps for facades of varying specificity and for two levels of goal ranges

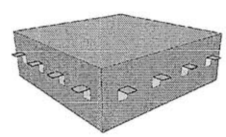
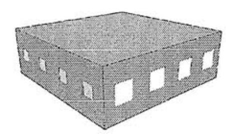
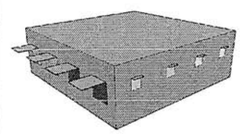
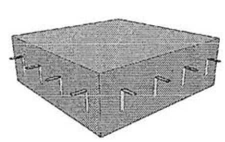
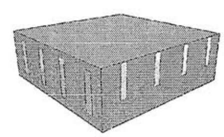
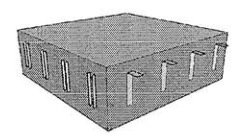
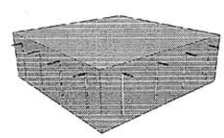
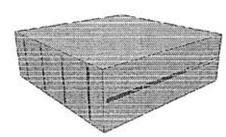
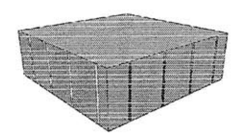
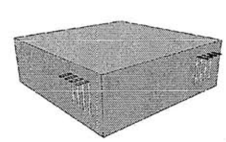
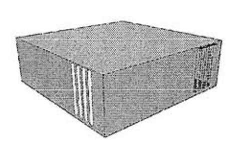
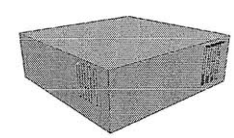
Facade Type	Starter Design	Best Design: Wide Range	Best Design: Narrow Range
Generic			
	Wide: 49.0% Narrow: 49.2%	99.8% In Range	94.4% In Range
Somewhat Stylized			
	Wide: 49.1% Narrow: 49.2%	94.5% In Range	92.3% In Range
More Stylized			
	Wide: 47.8% Narrow: 47.3%	95.1% In Range	93.4% In Range
Most Stylized			
	Wide: 52.0% Narrow: 51.6%	98.3% In Range	88.4% In Range

Figure 7.27: Starter and best designs for four levels of facade specificity and two levels of goal ranges

system is again able to find good solutions from all four starting designs, with performance ranging from 88.4% to 94.4%. Similarly to the wide illuminance range case, the best designs found during each of these runs greatly resembles the starting design (Figure 7.27).

From this brief study, it is clear that the starting facade design given to the expert system does have an effect on the behavior of the system and on the total improvement that can be found over a course of ten design iterations. In both studies, the final model that was generated starting with the most generic model had a higher performance than the final models generated by any of the more stylized facades; however, the difference between the best and worst final design in each case was less than 6%, and the most stylized initial models did not always result in the worse performance. The initial design was found to greatly influence the designs that the expert system considered. This study has shown that the expert system may work towards improved performance while maintaining an initial design aesthetic provided by the user, which was the intended behavior of the system based on the expert system as discussed in section 3.5.

7.3.2 Window Uniformity Scheme

In addition to being heavily influenced by the initial facade design, the expert system's search process is also dependent on the window uniformity scheme selected by the user at the beginning of the process. In this section, the models from case studies #6 and #7 from the micro-GA comparison studies were considered, once with the uniformity of the facade maintained and once with non-uniform facades allowed. These models were chosen for this study because it was hypothesized that the uniformity of the facade would be more influential on designs with more than one goal sensor than for single-goal scenarios. These two design problems each have two illuminance sensor planes with different performance goals, but they are considered non-conflicting goals because reasonable solutions exist which meet both goals at once. For the L-shaped room case study, the known good solutions featured small windows with shading devices on the west facade and larger windows on the south facade. For the trapezoidal case study, the known good solutions features windows clustered towards the west end of both facades. The performance of the expert system for both uniform and non-uniform facades for both case studies is shown in Figure 7.28.

It is clear from these charts that the non-uniform window scheme produces significantly better results for the trapezoidal case study, where the sensor planes are located perpendicular to the facades of interest. This improved performance is due to the fact that the non-uniform window scheme allows the expert system to target the two illuminance sensors individually by making different changes to the windows that are located closest to each one instead of making the same design change to all windows. This difference in the handling of windows based on their location relative to the different sensors is apparent in the final designs produced in each case (Figure 7.29). The final design for the non-uniform scheme still resembles the initial facade design, but the facades have each been divided into two, based on the locations of the two sensors. This final design has an

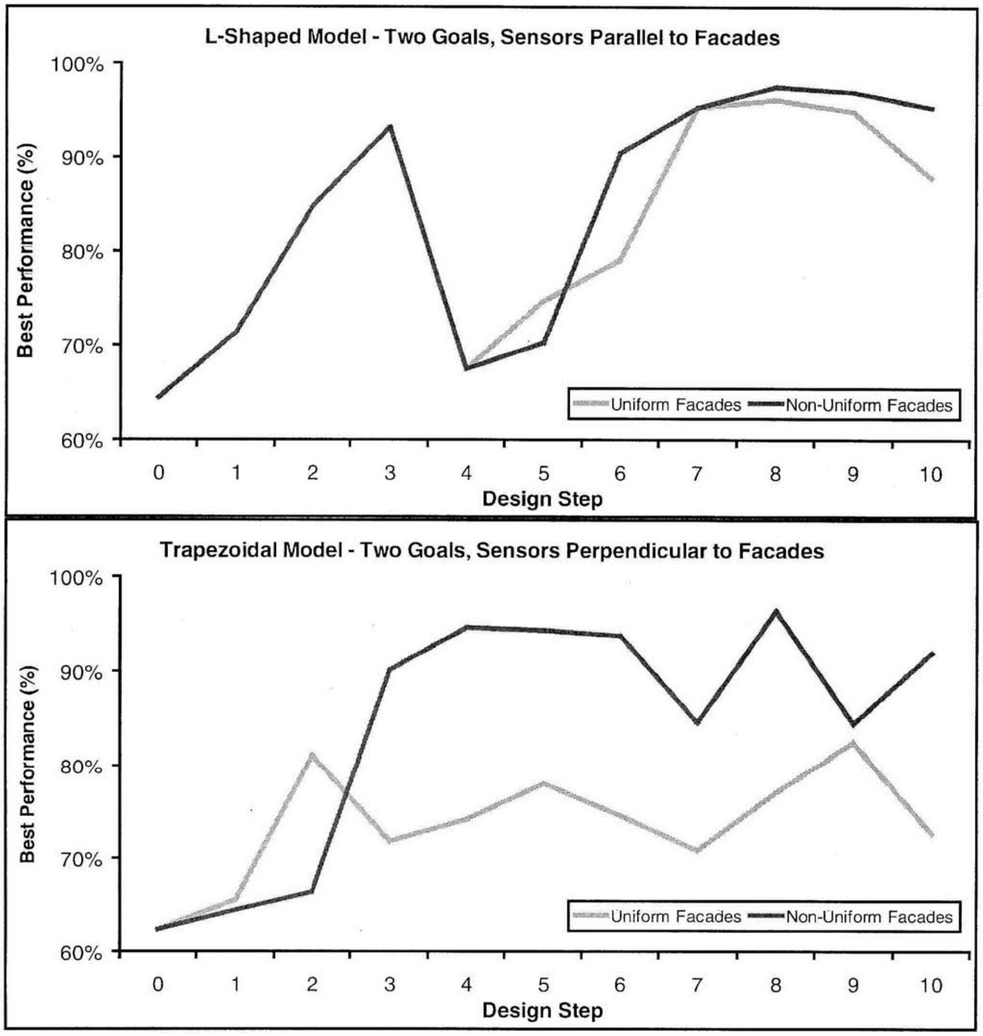


Figure 7.28: Expert system performance for L-shaped and trapezoidal models (from case studies #6 and #7) with two window uniformity schemes

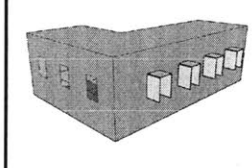
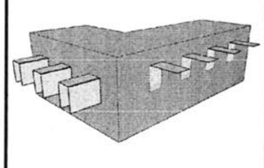
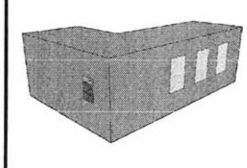
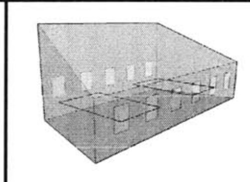
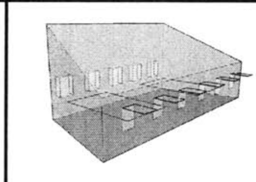
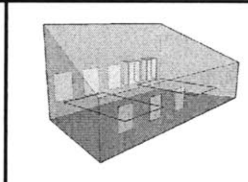
Performance Goal	Starter Design	Uniform Facade Best Design	Non-Uniform Facade Best Design
Two Illuminance Goals: Sensors Parallel to Facades			
	64.5% In Range	96.1% In Range	97.5% In Range
Two Illuminance Goals: Sensors Perpendicular to Facades			
	61.3% In Range	82.6% In Range	96.4% In Range

Figure 7.29: Comparison of best performing final designs from the expert system for L-shaped and trapezoidal models (from case studies #6 and #7) with two different uniformity schemes

average performance of 96.4%, which is 13.8% better than the performance of the uniform facade design.

In the L-shaped case study, where the sensor planes are located parallel to the facades of interest, the non-uniform facade scheme produced slightly better performance, but the final best designs for both schemes are within 1.1% of each other. In this case, the expert system process was exactly the same for the first four design iterations and followed a similar trend for the final six iterations (Figure 7.28).

It is clear from this set of case studies that the use of the non-uniform window scheme can produce dramatically better results for certain types of designs, particularly those where a facade could be effectively divided so that different windows would target different sensor planes. However, for designs such as the L-shaped room, where the facades are already located parallel to the sensor planes, the uniform and non-uniform window schemes produce similar results.

7.4 Expert System Strengths and Limitations

Based on the case studies presented in this chapter, several general conclusions may be made regarding the strengths and limitations of the expert system as an algorithm which works towards improving the daylighting performance of an initial design.

The strengths of the expert system are the following:

- The expert system can find designs which perform similarly to designs generated by a micro-GA for a variety of design scenarios.
  - It can work towards improved designs for single and multiple performance goals and for illuminance and/or glare metrics.
  - It is effective for a variety of initial models, including different types of building footprints and situations where some facades are of a different height than others.
  - It can solve for multiple goals at the same time and produce designs which have a high average performance.
  - For conflicting goal scenarios, it can produce a set of designs which provide the user with an understanding of the trade-offs between the conflicting goals.
- The expert system tends to improve designs very quickly (within fewer than ten design iterations).
- The expert system can find similarly good solutions for starting models with a range of different facade designs, including highly stylized facades. The system also maintains the initial design aesthetic during the design iteration process, so that final designs still resemble the initial design.
- The expert system can improve an initial design regardless of the window uniformity scheme selected, and it may find significantly better designs if the "non-uniform window" scheme is allowed.

The limitations of the expert system are:

- The expert system may not perform as well as a micro-GA in all situations, particularly those with narrow illuminance goal ranges and/or multiple goals, especially conflicting goals. However, this sacrifice in performance is necessary in order to include user-interactivity into the process.
- Unlike the micro-GA, which will always improve performance if allowed to run for more generations, the expert system may worsen the performance of a design if too many design iterations are completed.
- The expert system has a tendency to display a see-sawing behavior when faced with narrow illuminance ranges and/or multiple goals. This behavior is most extreme for conflicting goals, as the system may maintain a constant average performance as it alternates between working on one goal and the other. In such cases, the expert system may not find design scenarios with improved average performance.

- The expert system is greatly influenced by the design of the initial facade, which means that it does not explore as many design options as a GA. This is a limitation in cases where the initial facade design severely restricts the actions that the expert system can perform and the subsequent performance improvement that can be achieved.

- The expert system may perform very differently based on the window uniformity scheme selected by the user. The best performing design found by the system may be much worse for constrained designs than for unconstrained designs, which means that a user may not find the best performing designs if he or she constrains the problem.

7.5 Chapter Summary

This chapter presented an evaluation of the expert system performance based on a series of case studies. The first set of case studies compared the performance of designs created by the expert system to those generated by a micro-GA. These case studies demonstrated that the expert system was successful at improving the performance of designs for a variety of initial conditions and performance goal scenarios. The purpose of the comparison studies was to evaluate the performance of the expert system relative to a known optimization algorithm which could be relied upon to consistently generate designs with very good, if not globally optimal, performance. One early hypothesis of this thesis was that the expert system would likely not perform as well as a GA, particularly for problems with complex geometries or difficult performance goals. In order to accommodate the user-interactivity that would allow the expert system to be more naturally integrated into the design process than an algorithm such as a GA, it seemed likely that some degree of performance was to be sacrificed. The results of the case studies indicate that in some situations, the GA does indeed find designs which perform better than the expert system; however, the difference in performance between the designs found by the two algorithms in the case studies was small (a maximum of 4%, or 7.5% if extra micro-GA generations were included).

This chapter also presented a brief set of studies which demonstrated the effect of initial facade conditions and constraints on the performance of the expert system. It was shown that while the specificity of the initial facade design does affect the outcome of the expert system process, the expert system was able to find improved designs in all cases. The difference in performance between the best and worst final designs in these case studies was 6%, which was small relative to the total performance improvement. A second brief study investigated the effect of the window uniformity scheme on the resultant expert system behavior, and in this study, a more significant potential difference was found. For designs where the sensor planes are located parallel to the facades of interest, the selected window uniformity scheme had little effect on the performance of the expert system. However, for designs where the sensor planes were located perpendicularly from the facades of interest, the non-uniform window scheme selection was found to significantly improve expert system performance.

Based on the case studies discussed in this chapter, the expert system has been found to perform successfully. A potential user can have confidence that the design changes suggested by the system will improve the performance of his or her initial design if a number of design iterations are completed. However, the case studies described in this chapter all assumed that a "perfect user" was involved with the process. The "perfect user" scenario was one in which the "user" would always select the first design change suggested by the expert system and the magnitude of design change with the highest performance. Chapter 8 will evaluate the performance and success of the expert system when user-interactivity is included in the process and will describe a study in which human designers were asked to complete a design problem using the expert system.

Chapter 8

An Expert System Design Process

8.1 Introduction

A major goal of this thesis was to develop an intuitive tool and a performance-driven design process which could be used by designers to improve daylighting performance in the early design stages. In order to engage designers in the process, a high level of user-interactivity was integrated into the expert system. While optimization schemes often act as "black box" algorithms, producing designs with little or no input from a user, the expert system process *requires* user input in order to produce new designs. This level of interactivity was included to allow the user to be in control of the changes made to his or her design at all times and to enable the expert system to be more compatible with the traditional architectural design process. Because the user is fully involved in the decision-making process, it is also possible that the expert system will provide an educational value to the designer by informing him or her about how different design elements affect performance.

While the previous chapter investigated the expert system behavior and performance in the absence of human inputs, this chapter describes the results of a study which evaluated the performance of the expert system as a user-interactive process and as a design tool. During the study, designers were asked to interact with the expert system and to solve a brief design problem while focusing on daylighting performance. Several important results were obtained from this study. The first is an assessment of the ability of the expert system to find designs with improved daylighting performance when a human user is allowed to interact with it in an independent way. The second is an evaluation of the expert system process as a method for educating designers about daylighting and influencing their designs to include design elements which result in good daylighting performance. A final evaluation focuses on user satisfaction and the acceptance of the expert system by designers.

This chapter presents the details of the user study, including: the profile of participants, the study procedure, a description of the design problem that each participant was asked to solve, and the quantitative and qualitative results of the study. The chapter also describes the guided design process that a human user experiences when he or she works with the expert system. The chapter concludes with a discussion of the expert system as a user-driven and user-interactive design tool.

8.2 Guided Design Process

The expert system described in this thesis is intended to supplement the traditional design process by engaging designers in a guided, performance-driven search process. The expert system interface, described in section 6.3, provides an intuitive way for designers to view performance data and to interact with the expert system. The main steps involved with the expert system process are the following:

1. The user creates a 3d SketchUp model of his or her initial design which conforms to the guidelines specified in Appendix A.
2. The user initializes the expert system from a pull-down menu in Google SketchUp. The user is prompted to enter the following inputs (described in further detail in section 5.5.1):
    (a) Performance goals for all illuminance and/or glare sensors in the 3d model.
    (b) A priority level for each goal.
    (c) A window uniformity scheme.
    (d) Times of day and seasons of interest.
    (e) Location data and corresponding weather file.
3. The LSV engine simulates the initial model and calculates the goal-based performance metrics (described in section 5.5.2). Once the simulations are complete, the Lightsolve program opens automatically, and the expert system interface can be selected from within the Lightsolve program. Upon opening, the expert system interface is automatically populated with data from the user's initial model.
4. From the expert system interface, the user can view a list of suggested design changes that can be made to his or her initial model. The user may skip forward or go backwards between the various options on the list before choosing one.
5. After the user selects one design change to try, the expert system automatically makes the selected change to the 3d model in SketchUp. The expert system makes three different magnitudes of the selected change, using the logic described in section 6.4.2. For each change, the expert system creates and saves a new 3d model, runs the LSV engine, and calculates the goal-based performance.

6. After the three different magnitudes of change have been simulated, the expert system displays all three results in the interactive graph within the interface (Figure 6.3). The user may browse the views of the current design and the temporal maps to see how the performance and design have changed in each of the three options. The user must choose one of the three possibilities before continuing to the next design iteration.

7. After one or more design iterations have been made, the user may then choose either to select a new design change to try from the list presented by the expert system, or the user may return to a previous iteration of the design (including the initial model). Returning to a previous iteration will allow the user to investigate a new path of design iterations. Once the user elects to make another design change, steps 5 and 6 repeat.

8. After several iterations, the user should be able to view the progressive performance of the design, as in Figure 3 (in Chapter 6). The user may stop the process at any point.

This guided process, along with the corresponding tool and interface, was developed to support the design process by enabling the features described in section 3.5. The user study presented in this chapter was conducted to test how well real designers responded to this guided design process.

8.3 User Study: An Expert System Design Tool

To evaluate the performance of the expert system when incorporated into the design process, a user study was conducted during which participants were asked to interact with the system and solve a daylighting design problem. The study was designed to help answer the following questions:

- Can the expert system still improve the performance of a design when independent human interaction is included into the process?
- Can the process of working with the expert system positively influence a designer's final design, if the final design is one designed strictly by the designer and not generated by the expert system?
- Can the process of working with the expert system educate a designer about daylighting for a specific design problem? Can it educate a designer about general daylighting concepts?
- If there is a benefit to using the expert system design process, is this benefit greater to participants who have had little to no experience working with daylighting or to participants who have had more substantial experience?

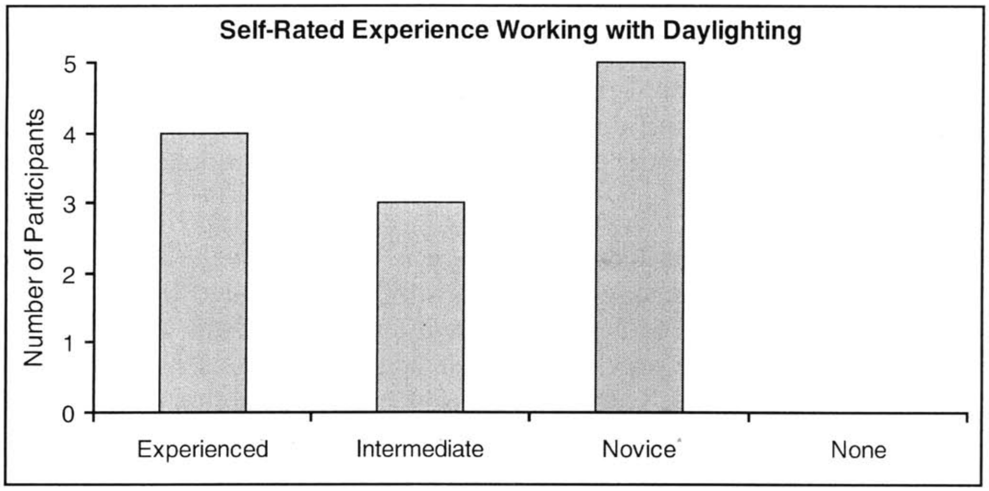


Figure 8.1: Breakdown of participants' previous experience working with daylighting

- Do the participants accept the expert system as a design tool? Would they use it for an actual design project?

8.3.1 Profile of Participants

Twelve participants were included in the user study. Because the goal of the study was to evaluate the expert system for use by designers, only those with at least one prior degree in architecture and with at least one year of experience working for an architectural design or architectural consulting firm were allowed to participate. The median work experience in a design firm was 3.8 years (minimum 1 year, maximum 8 years). The breakdown of the highest previously completed degrees was:

- Bachelor's level degree in architecture: 4 participants
- Master's level degree in architecture: 8 participants

Additionally, most participants were in the process of completing a second or third degree in architecture or a related field (building technology or computation) at the time of the study. The breakdown of the current status of participants was:

- Master's level student in architecture or related field: 5 participants
- Doctoral level student in architecture or related field: 6 participants
- Architectural design faculty: 1 participant

The participants were also selected so as to represent a variety of backgrounds in daylighting. They were asked to rate their experience level in working with daylighting using one of four categories: experienced, intermediate, novice, or none. The distribution of responses are shown in Figure 8.1. The group was fairly evenly split, with a total of four (self-ranked) experienced daylighters, three intermediates, and five novices. No participant chose the final option of "no daylighting experience."

Finally, participants were asked to describe their previous experiences working with daylighting simulation tools for different stages of the design process. In general, most participants had relatively little experience working with daylighting simulation programs. Additionally, although participants were asked to differentiate between programs used during the early, later, and final design stages, most participants did not respond differently based on the different stages of design.

The breakdown of participant responses were the following (please note that some participants indicated that they used multiple simulation programs):

- No experience using daylighting simulation at any time during the design process: 6 participants
- Daylighting simulation during early and later design phases:
  - Radiance and Daysim: 3 participants
  - Ecotect: 3 participants
  - Lightsolve: 1 participant
  - 3d Studio Max: 1 participant
- Daylighting simulation during final phase (for analysis only):
  - Radiance and Daysim: 2 participants
  - Ecotect: 1 participant

8.3.2 Study Procedure

The user study was conducted as a series of brief segments, which included three design sessions and two questionnaires. The total amount of time used for each participant was approximately two hours. The design problem will be described in detail in section 8.3.3, and the questionnaires, design briefs, and template sheets are included in Appendix B. The same design problem was solved by each participant three times: first by hand, then using the expert system, and finally by hand again. During all sessions, participants were allowed access to pencils, blank paper, a calculator, and a sun course diagram for Boston, MA. Participants were also allowed to re-visit the two tutorial PowerPoint slides at any point during the study.

The sessions of the study were the following:

1. Introductory Questionnaire: The participant filled out a basic information questionnaire about his or her previous education and experience in design, daylighting, and simulation-based tools.

2. General Tutorial: The participant viewed a PowerPoint slideshow tutorial. The tutorial introduced the participant to:

   (a) A description of the tasks he or she was expected to perform during the study.

   (b) A description of the design rules that the participant should follow during the design sessions.

   (c) A brief explanation of basic daylighting information and metrics.

3. Design Session #1: The participant was asked to complete a design problem. For this problem, the participant was given an initial massing model and a set of daylighting performance goals. The participant was asked to design two facades on the massing model and attempt to meet the daylighting performance goals as well as satisfy him- or herself as a designer. An example of a previous facade design was also provided, along with the performance of that example design (Figure 8.2a). In this design session, the participant was asked to sketch his or her design by hand and to draw the final design on a template sheet (Figures B.7 and B.8, Appendix B,). The participant was given as much time as necessary to read through the design brief and 20 minutes to sketch his or her design.

4. Expert System Tutorial: The participant viewed a PowerPoint slideshow tutorial that introduced him or her to the expert system and its interface.

5. Design Session #2: The participant was asked to complete the same design problem as the previous design session using only the expert system. For this session, the participant began the process with the same example model shown to them in the previous session (Figure 8.2a). During this session, the participant was allowed to choose to accept or decline design changes suggested by the expert system, to choose the magnitude of the design change, and to return to previous design iterations. The participant was also allowed to explore designs which resulted in decreased performance if desired. During this session, the participant was not allowed to change the design by hand or in SketchUp. This session lasted for 40 minutes, after which the participant was asked to choose a "favorite" design.

6. Design Session #3: The participant was asked to complete the design problem for the third time and was told that this design would be considered the "final" version of the design. The participant was told that he or she could re-visit either or both of the designs produced during the first two sessions or completely start over. The participant was given 15 minutes to sketch by hand and draw the final design on a provided template sheet.

7. Final Questionnaire: The participant filled out a final questionnaire about his or her satisfaction with the final design, the experience of using the expert system, and whether he or she would use the tool in a real design context.

The purpose of the three design session format was to determine if the process of using the expert system was able to positively influence each participant's final design. During the first design session, participants relied primarily on their own intuition and understanding of daylighting in order to create an initial design. Participants were not told how well these initial designs performed based on the daylighting goals. During the second session, participants all worked with the same starting design, which in many cases was quite different from their own initial design. During this session, however, participants were able to view the performance of the model after they applied various changes to it.

One hypothesis of this study was that if participants choose design changes that resulted in improved performance during the expert system session, they might choose to apply some of those design changes to their own designs during the third session. A corollary of this hypothesis was that if participants made these design changes to their own initial designs, the performance of those designs should improve. This improved performance would indicate that participants were able to learn something about working with daylighting by using the expert system, and that the process of using the expert system, even for a seemingly unrelated model, could influence participants to incorporate certain design elements into their own designs.

8.3.3 Design Problem

This section describes the design problem that participants were asked to work through during the user study. This design problem was developed to be of medium level difficulty so that designers who were experienced with daylighting concepts could create a very good solution using only their intuition. The problem was meant to be challenging for those designers who were not experienced with daylighting, but not so difficult as to discourage them.

The participants were asked to work through a conceptual design for the facade of a school library wing in Boston, MA which should use natural light instead of artificial light as much as possible. Participants were informed that they were taking over the project from a colleague who had already started working on the design. They were required to keep the original massing model that their colleague had designed (footprint, wall heights, and interior walls). However, they were allowed to change the facade elements as necessary to meet the daylighting goals. They were allowed to choose the size and placement of windows, the types of glass used, and the types, size, and placement of shading devices. The two facades that were considered were those oriented towards South and East.

The library space has three main areas: 1. A double-height main study area, which should receive lots of light; 2. A smaller study area that overlooks the main study area, which

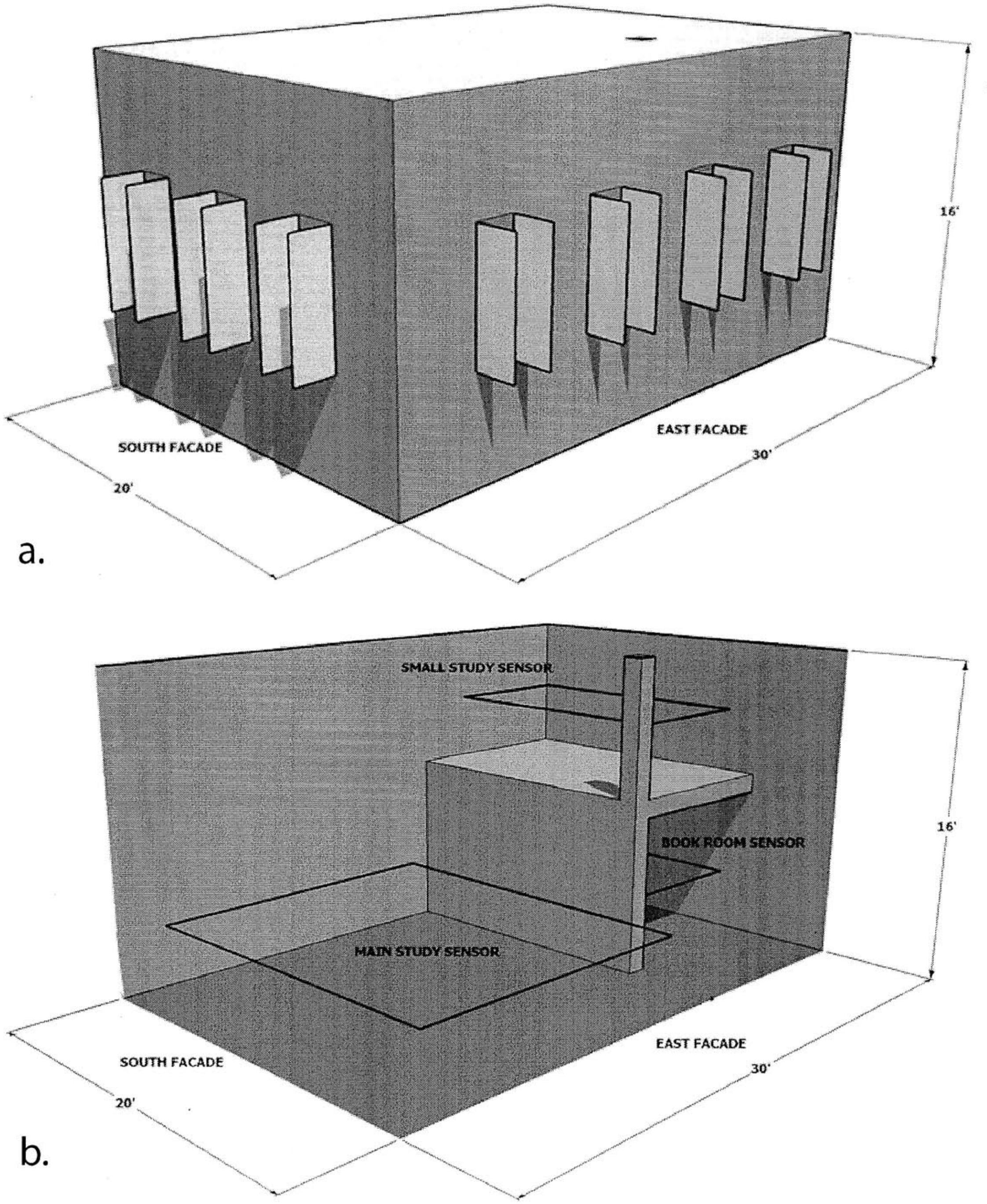


Figure 8.2: Library massing model with (a) example facades and (b) three sensors shown.

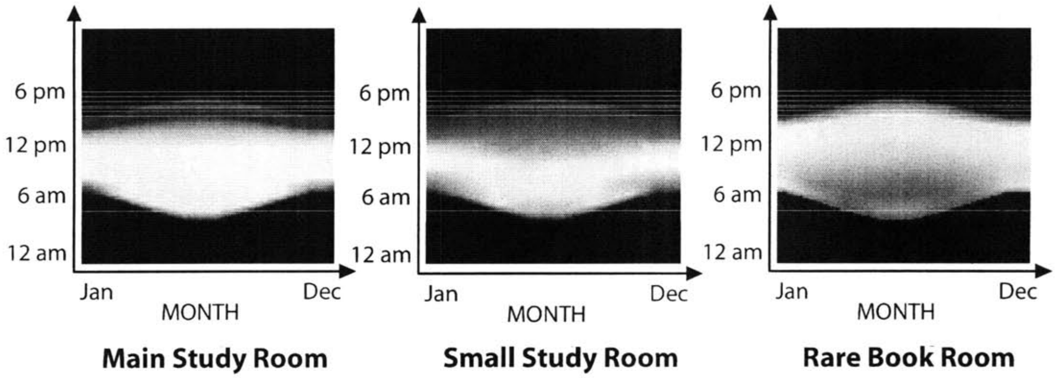


Figure 8.3: Temporal maps for the three sensor planes in the example library model.

should receive an adequate amount of light; 3. A rare book room, in which light must be carefully controlled. The original design and the location of sensors within the space are shown in Figure 8.2. Participants were told that, based on the client's description of the space, their firm had decided that the specific daylighting goals they should work towards were:

- Main Study Space:
  - Minimum illuminance levels: 500 lx (desired) down to 400 lx (acceptable).
  - No maximum illuminance levels.
- Small Study Space:
  - Minimum illuminance levels: 200 lx (desired) down to 0 lx (acceptable).
  - Maximum illuminance levels: 800 lx (desired) up to 1000 lx (acceptable).
- Rare Book Room:
  - No minimum illuminance levels.
  - Maximum illuminance levels: 200 lx (desired) up to 400 lx (acceptable).

Participants were told that if the illuminance on the entire area of a sensor plane falls within the desired range during all daylit times of the year, the performance of that sensor would be 100%. As an example, participants were shown temporal maps of the performance of the three sensor planes for their colleague's design (Figure 8.3). From these maps, it is clear that the main study space has the best performance, although the illuminance is too low during parts of the year. The rare book room performs fairly well but the illuminance is too high during the morning hours throughout the year, except for December and January.

The small study space performs the least well out of the three, and it does not receive enough illuminance during the afternoon over most of the year.

The average performance of each sensor in the colleague's design were also provided to the participants:

- Main Study Space: 80%
- Small Study Space: 65%
- Rare Book Room: 70%
- Average of All Spaces: 72%

Finally, the participants were informed that the client had requested a certain aesthetic which must be maintained. The following design rules were given:

- Windows must be rectangular or square.
- Glass may be normal or translucent, but not tinted with color.
- Shading devices must be opaque, and must be vertical or horizontal.
- Both vertical and horizontal shading devices may be used on the same window.
- No advanced systems may be used.
- It is up to the designer to determine if a uniform facade aesthetic should be maintained.
- The design should achieve the daylighting goals *and* also satisfy the designer.

8.3.4 Procedure for Modeling Designs

For this study, it was necessary to determine the performance of models from all three design sessions, including those which the participants completed by hand. Following each participant's study session, the author constructed SketchUp models of the participant's initial and final designs and calculated the performance on all sensor using the LSV simulation engine. The dimensions and locations of windows and shading devices on each facade were modeled based on the template sheet drawings. An example template sheet and the corresponding SketchUp model are shown in Figures 8.4 and 8.5, respectively.

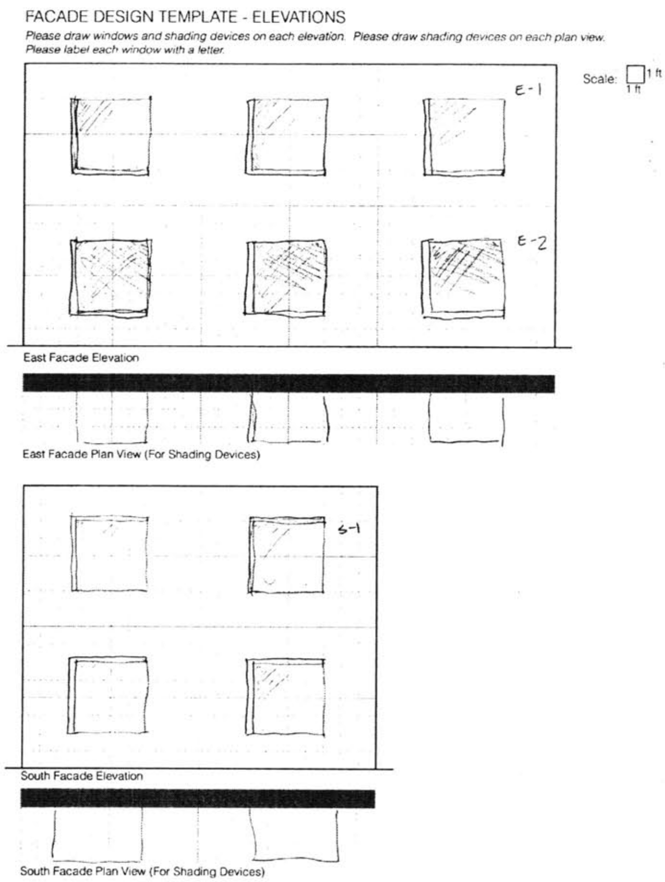


Figure 8.4: Example template sheet with facade design drawn.

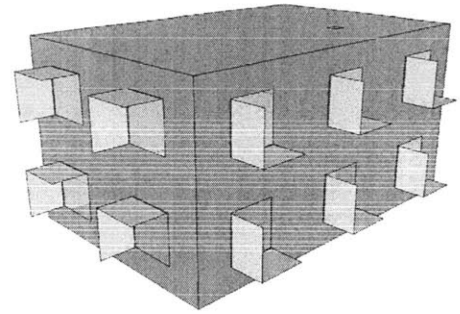


Figure 8.5: SketchUp model based on the filled-in template sheet in Figure 8.4.

For glazing types, participants were allowed to check one option in each of two categories, "View" and "Amount of Light Let In," as described below. The corresponding material values for each choice are:

- View:
  - Transparent (Clear View): All specular transmittance.
  - Translucent/Frosty (Blurred View): Equal contributions of specular and diffuse transmittance.
  - Opalescent/Milky (No View): All diffuse transmittance.
- Amount of Light Let In:
  - Most (For ex., Single-Glazed Clear): Total transmissivity = 80%
  - Intermediate (For ex., Double-Glazed Low-E): Total transmissivity = 60%
  - Least (For ex., Grey Tinted): Total transmissivity = 40%

For example, if a participant selected "Translucent/Frosty" and "Intermediate," the glazing would be modeled as 30% specular transmissivity and 30% diffuse transmissivity.

8.4 Quantitative Results

To determine performance of each model produced during the three design sessions, the LSV simulation engine was used. For each design, the performance was considered to be the percentage of the total area of each sensor plane that the illuminance was calculated to fall within the desired goal range, averaged over the whole year. The performance of each sensor plane was averaged into a single value which represented the total performance of each design over the whole year. A design which met all goals would be one for which this average value would equal 100%.

8.4.1 Performance of First Session Designs

In the first design session, participants were asked to complete the design problem by hand, using only their intuition. The results of this session are shown in Figure 8.6, where the designs have been ordered from least successful to most successful in terms of average whole-year performance across all three illuminance goals. Despite staying within fairly rigid aesthetic constraints, the designers produced a wide variety of designs.

The mean performance of these twelve designs was 73.9%, which is similar to the performance of the example design shown to each participant (71.9%). As indicated in the chart, six participants produced designs that performed at least 10% above the example, three participants produced designs that performed similarly to the example, and three participants produced designs that performed well below the example. It is interesting to note that while the best performing designs were created by participants who rated themselves as either "experienced" or "intermediate" in working with daylighting, there was no clear correlation between the self-rated daylighting experience ranking and performance of the initial design.

8.4.2 Performance of Designs Generated by the Expert System

In the second design session, participants were asked to use the expert system for a fixed amount of time, starting with the example model. In general, most participants were able to make four design iterations during the session (one participant was able to make five iterations and one was only able to make three iterations). The designs produced and their performances during the first two sessions are shown in Figure 8.7.

The mean performance of these twelve designs was 87.6%, and the performances of all twelve designs were higher than that of the example model, which indicates that every participant was able to improve the performance of the starting model by using the expert system. Additionally, eleven out of twelve designs were improved by 10% or more; the design which saw the smallest improvement was created by the participant who was only able to complete three design iterations during the session. This participant also stated during the session that he or she consciously choose to explore worse performing options due to aesthetic preferences.

The result that every participant was able to find a better performing design than the starting model is important because it demonstrates that the expert system can improve the performance of designs even when the participants' unique sets of design choices were introduced into the process. Although participants only had a short amount of time to interact with the system, and although participants were not specifically told to choose design changes which improved performance, the expert system was able to find good solutions nevertheless.

The range of designs found during this session also hints at how one cannot predict how designers will interact with the tool: although all twelve participants began with the same initial model and design problem, no two resultant designs were the same after only a small number of design iterations. This result demonstrates how unique a designer's individual set of design decisions may be, even for a simple problem such as that presented in the user study, and it underlines the value of the user interactivity of the expert system. It is also apparent from the results of this session that the designs that the expert system can produce are limited by the starting geometry: all twelve designs shown in Figure 8.7 are clearly derivative of the example model. Nevertheless, it can be seen as a positive result of the expert system that the initial geometry is respected in such a way.

Another interesting result of this session is that the performances of the designs found by the expert system were more uniform than those designed strictly by the participants in

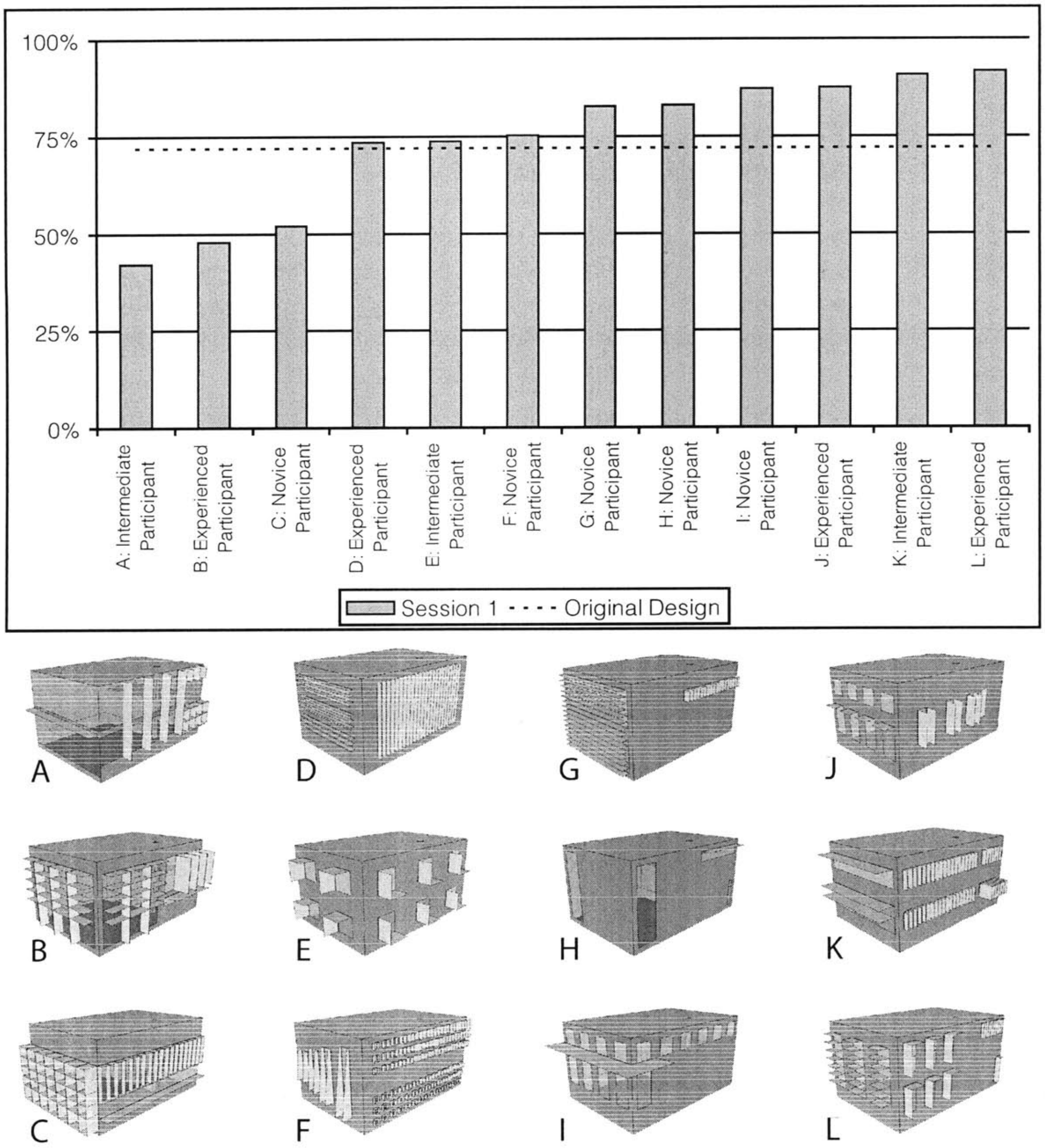


Figure 8.6: Performances for session 1 designs.

the first session. Using the expert system, those participants who struggled during the first session were able to find designs which performed similarly to the designs generated by those who were successful in the first session. Additionally, nine out of twelve of the second session designs outperformed the designs created by the same participant during the first session. In this session, it was found that the expert system was able to effectively guide both novices and more experienced daylighting designers towards improved performance.

8.4.3 Performance of Final Designs

In the final design session, participants were asked to re-visit the same problem for a third time and to draw their final design by hand. Participants were not restricted and were allowed to draw inspiration from either or both of the first two design sessions. They could also completely start over if desired. It was the hope of the author that participants would combine their initial design with elements from the expert system design during this session to create a better performing final design.

This positive scenario occurred in many of the twelve studies. The results of all three sessions are shown in Figure 8.8, and it is clear from this graph that in many cases, particularly for those participants who struggled with the first design session, the final design had a higher performance than the initial design. The mean performance of these twelve designs was 82.9%, which was 9.0% higher than the mean performance of the initial set of designs.

These results are interesting because participants completed both the first and third sessions by hand, with no performance feedback. During the second session, participants did not work with their own initial design, but instead with an example design that may have had little in common aesthetically with their own initial design. Nevertheless, nine out of twelve designers were able to produce final designs which performed better than their initial designs. In many cases, a participant's final design demonstrated aesthetic qualities that reflected both the first and second designs (Figure 8.8, to compare with Figures 8.6 and 8.7). This result indicates that the process of using the expert system, even with a generic example instead of their own design, was able to educate some of the designers about ways in which performance could be improved.

8.4.4 Comparison of Results Between Participants With and Without Previous Daylighting Experience

One hypothesis of this study was that participants with little to no previous daylighting experience would benefit more from the process of using the expert system than participants who had more substantial previous experience working with daylighting. The results of this study instead indicate that those who benefited the most from the process of using the expert system were those who produced the least successful initial designs, and that

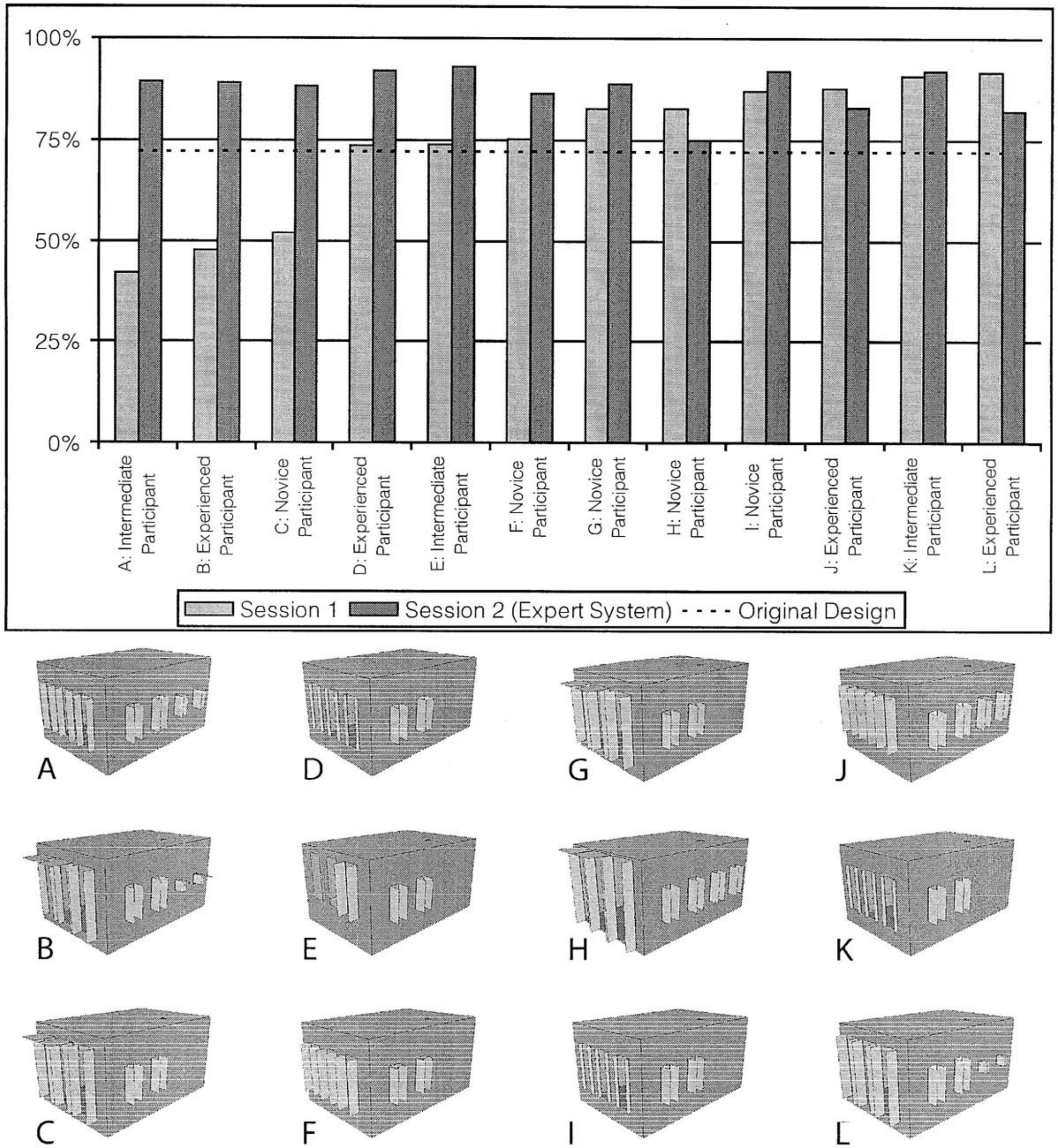


Figure 8.7: Performances for designs from sessions 1 and 2.

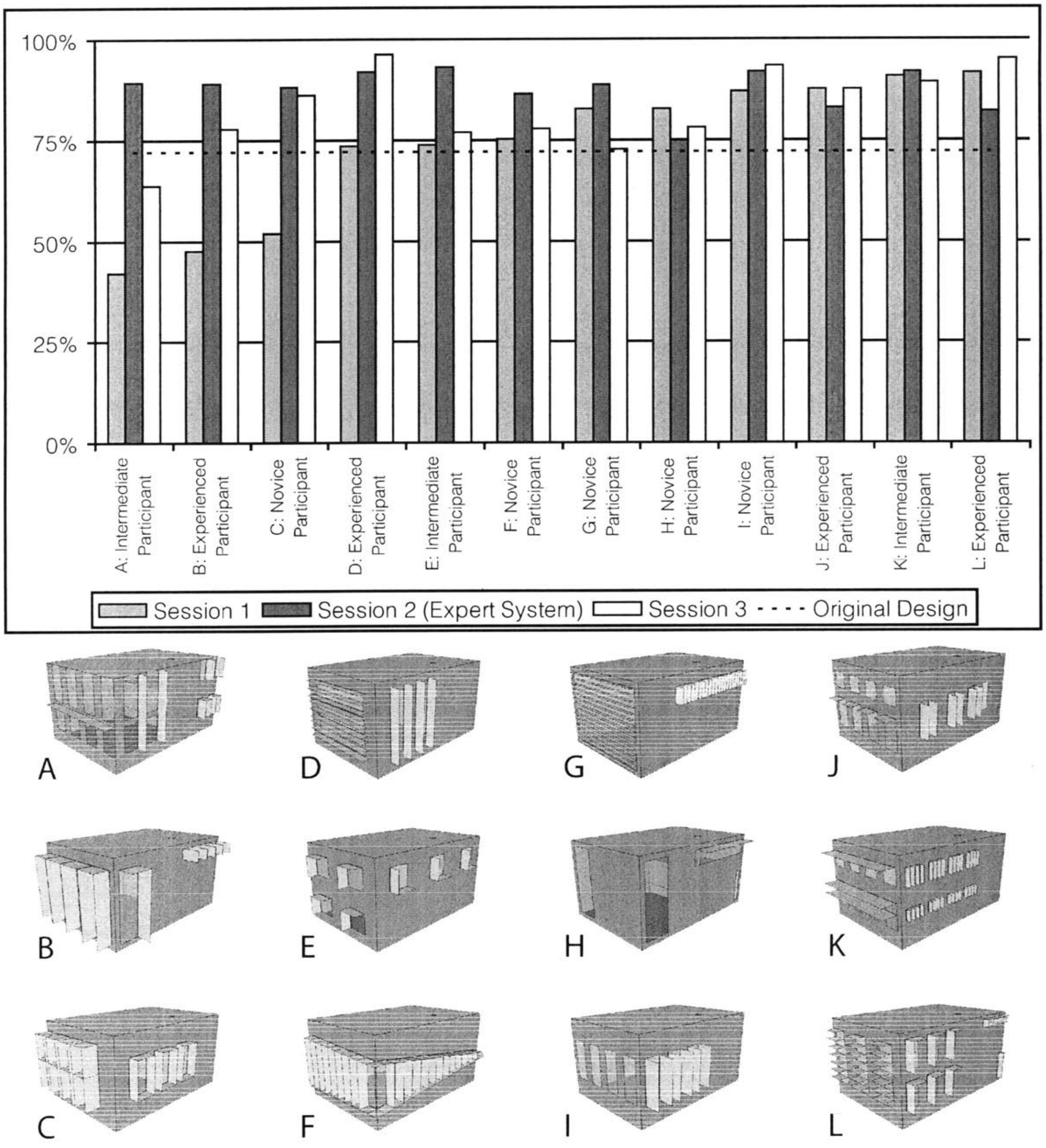


Figure 8.8: Performances for designs from all three sessions.

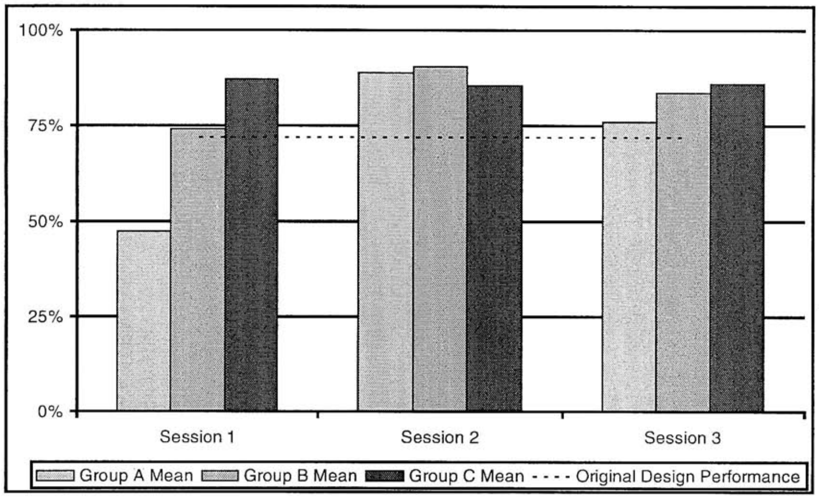


Figure 8.9: Mean performance for Groups A, B, and C for each design session.

these participants did not necessarily consider themselves inexperienced at working with daylighting.

Based on the performance of their initial designs, the participants can be divided into three general groups: Group A consists of three participants whose initial designs performed worse (more than 20% lower) than the example design, Group B consists of three participants whose initial designs performed about the same as the example (within 3.5%), and Group C consists of six participants whose initial designs performed better than the example by 10% or higher. The performance of each of the three groups were the following (also shown in Figure 8.9):

- Group A:
  - Mean initial performance: 47.2%
  - Mean improvement between sessions 1 & 3: 28.7% (min 21.5%, max 34.4%)
- Group B:
  - Mean initial performance: 74.1%
  - Mean improvement between sessions 1 & 3: 9.5% (min 2.6%, max 22.7%)
- Group C:
  - Mean initial performance: 87.1%
  - Mean improvement between sessions 1 & 3: -1.1% (min -10.0%, max 6.4%)

These results indicate that the greatest benefits from using the expert system occurred for those participants whose initial designs performed the least successfully. These results are intuitive as there was more room for improvement if participants began with a lower performing design than if they began with a high performing design. However, these results also demonstrate that the expert system was able to allow those participants who produced the weakest initial designs to ultimately produce designs whose performance approached those developed by the more successful designers. While the difference in mean performance between Groups A and C was close to 40% for the initial designs, the process of using the expert system reduced this difference to only 10% for the final designs.

Additionally, the results show that the mean performance of the Group A final designs was about the same as the mean performance of the Group B initial designs, and likewise, the mean performance of the Group B final designs was about the same as the mean performance of the Group C initial designs. Such results indicate that the process of working with the expert system between the first and third design sessions effectively allowed participants in the lower two groups to "move up" one group.

It is interesting to note that Groups A and B each consisted of one self-rated novice, one intermediate and one experienced daylighter. Group C consisted of three novices, one intermediate, and two experienced daylighters. It is clear that for this particular study, the self-rated experience level of each participant had little to no correlation with the performance of his or her initial design.

8.5 Qualitative Results of Using the Expert System

In addition to quantitative results based on design performance, the user study produced qualitative results, based primarily on the participants' responses on the final questionnaire and observed behavior of participants during the study. This section includes these qualitative results as well as a brief analysis of the influence of the expert system on the aesthetics of each participant's final design.

8.5.1 Influence of the Expert System on Design

The first two questions on the final questionnaire asked the participants how they felt about their final design, as compared to their first design, when they considered performance and aesthetics. These two questions were designed to help determine how the process of using the expert system during the second design session influenced each participant's final design. The participants responses to these questions are shown in Figure 8.10. Additionally, each participant was asked directly if the process of using the expert system influenced their final design (Figure 8.11).

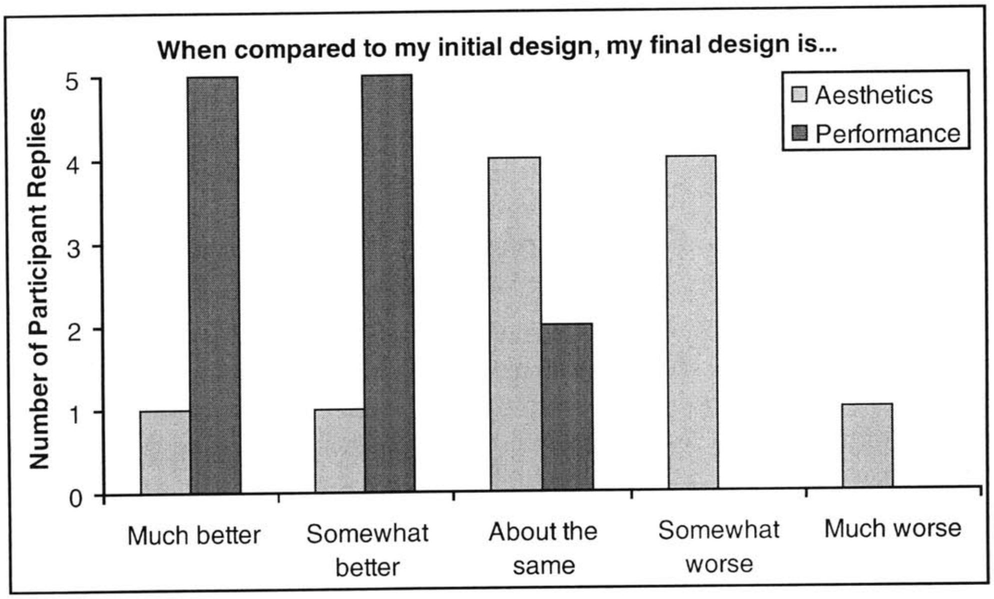


Figure 8.10: Participant responses to questions asking them to compare their final design to their initial design.

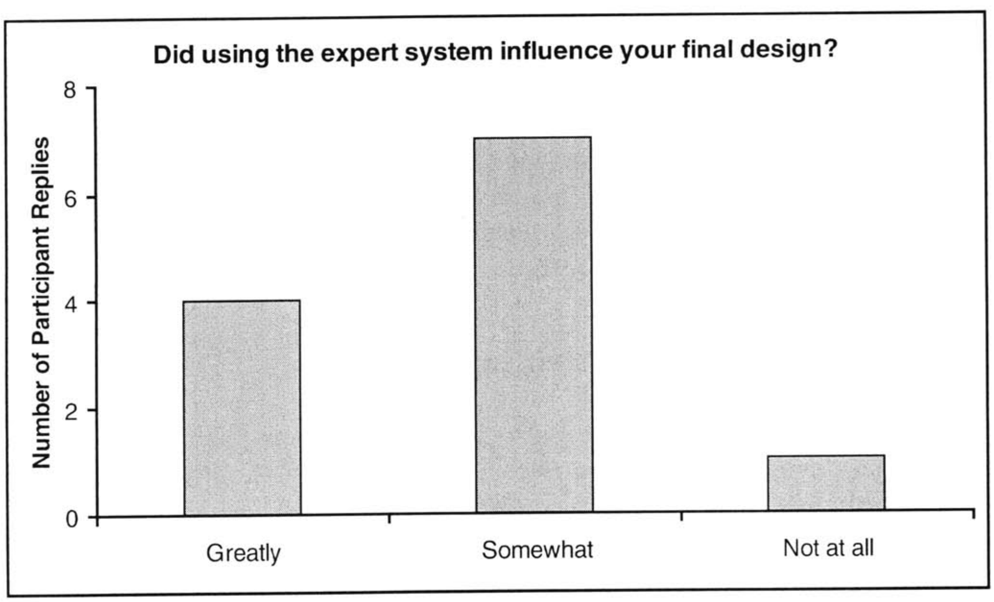


Figure 8.11: Participant responses to question asking them if their final design was influenced by the process of using the expert system.

From Figure 8.11, one can note that eleven out of the twelve participants felt that they were at least somewhat influenced by the process of using the expert system, and that four participants were greatly influenced. Figure 8.10 indicates that, in general, the participants felt that their final designs performed better or the same as their initial designs, but that the aesthetics of their final designs were the same or worse than their initial designs. These responses indicate that in many cases, the expert system may have influenced the designers to sacrifice aesthetics in some way for performance, despite the fact that they were told during each design session that they were supposed to try to meet the daylighting goals *and* to satisfy themselves as a designer.

Additionally, it is worth noting that the designers did not actually know if their final design outperformed their initial design; instead, the final questionnaire asked them only if they "felt" that their final design performed better. In nine out of twelve studies, the participant's response about whether their final design performed better was correct. This result is interesting because it indicates that the process of using the expert system may have helped the designers develop a small amount of intuition about the design problem.

An analysis of the aesthetics of the designs themselves confirms that in many cases, the participants seem to have been influenced by the design changes proposed to them by the expert system during the second session. It is striking that in most cases, the initial and final designs do resemble each other more than they resemble the second design. Through the three sessions, the initial design aesthetic of almost all participants was maintained. In many cases, certain elements of the second design appear to have also been incorporated into the final design. One clear example is removal or minimizing of windows on the northern end of the east facade, which is a common characteristic of many of the designs generated in the second session by the expert system. In designs A, B, C, D, F, I, K and L, such a design change is visible between the first and final models (Figures 8.6 and 8.8). In seven out of those eight cases, the performance of the final design was also found to be better than that of the initial design (and in the one case where performance was decreased, it was only lowered by 1%).

8.5.2 Educational Value

It was shown using quantitative data that the process of using the expert system helped many of the participants improve their designs, particularly those whose initial design did not perform successfully. Participants were also asked two questions on the final questionnaire about the educational value of using the expert system: first, whether they thought that they learned something new which helped them approach the specific design problem, and second, whether they thought that they learned something new about daylighting in general. The responses to these questions are shown in Figure 8.12.

It is clear from Figure 8.12 that the majority of participants responded that they learned a "small amount" about both the specific design problem and about daylighting in general. That they learned a "small amount" rather than a "large amount" may be partially the

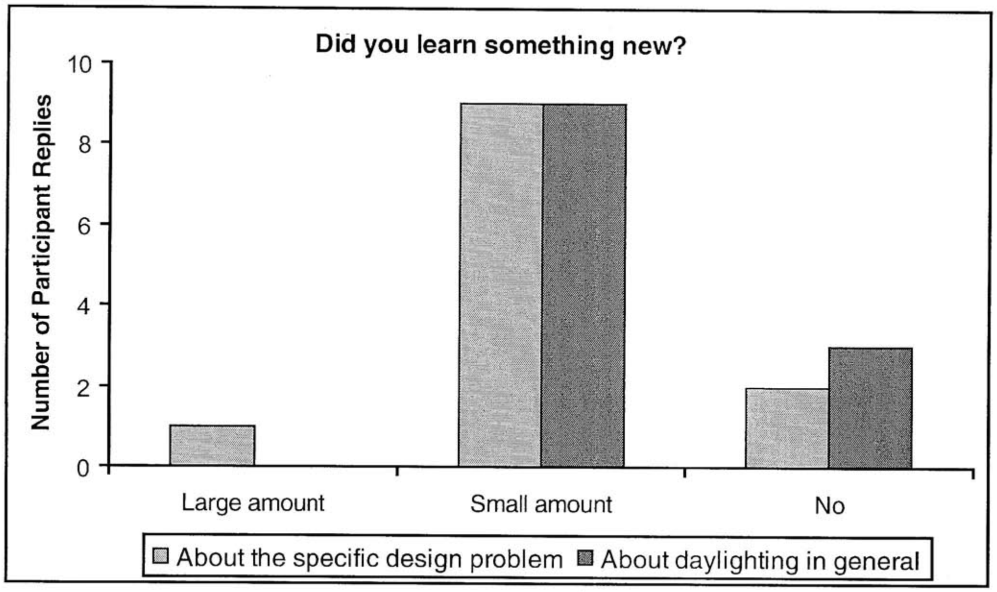


Figure 8.12: Participant responses to questions about the educational value of the expert system.

result of having only a limited amount of time (40 minutes) to work with the tool. Additionally, it is likely that if the question had been re-phrased to include one or two other possible responses (for example, "moderate amount"), the diversity of responses would have increased.

Nevertheless, the fact that the majority of participants claimed to learn even a small amount during their limited time with the expert system is a positive result. It is also interesting to note that although two participants claimed that they did not learn anything new about the design problem by using the expert system, these two participants were those who saw the highest amount of improvement between their initial and final designs (30.2% and 34.4%). These two participants also both responded that their final design performed "much better" than this initial design, so the fact that they claimed to have learned nothing from the expert system is puzzling.

8.5.3 The Expert System as a Design Tool

Participants were asked two questions on the final questionnaire which helped determine whether they were satisfied with the expert system as a design tool: first, whether they would consider using the expert system again for a studio project, and second, whether they would consider using the expert system again for a professional design project. The participants' responses to these questions are shown in Figure 8.13.

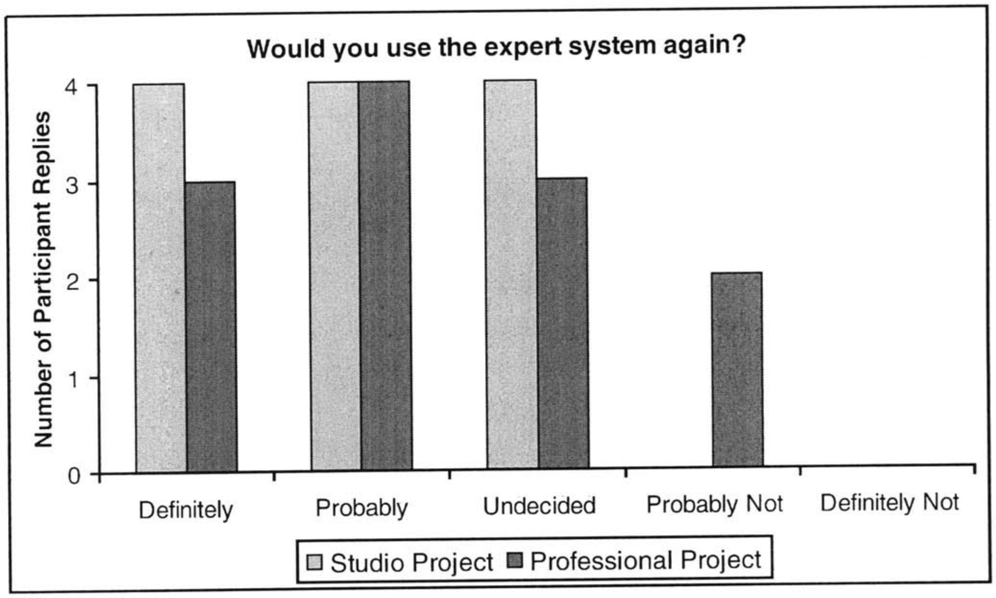


Figure 8.13: Participant responses to questions asking them if they would consider using the expert system again.

For both studio and professional projects, about two-thirds of the participants responded that they would consider using the expert system. Given the limited amount of time and control that participants had in using the expert system during the study, this is a positive result. Additionally, although some participants were "undecided", there were no participants who responded that they would "definitely not" use the tool for a future project, and only two participants replied that they would "probably not" use the tool for a professional project. The reasons given for these negative responses were the lack of complex geometries that the expert system was able to consider, and in one case, a lack of trust in the expert system advice. It is worth noting that the participant who lacked trust in the tool was able to improve the performance of his or her design by more than any other participant, so this lack of trust was without cause. In general, the responses to these questions indicated mostly positive reception of the expert system.

8.5.4 Additional Participant Feedback

The final questionnaire allowed participants to comment on their responses to the multiple choice questions and to write in their general likes and dislikes about the expert system tool.

In general, the most popular responses were:

- Likes:
  - The interactive graph with bar representation of performance, which provided a way to track performance over multiple iterations.
  - That the expert system provided a way to test the relationship between a certain design change and performance.
  - The interface itself and the ease of its use.
- Dislikes:
  - The limited number of design changes that the expert system could suggest.
  - The limited geometry that the expert system could handle.
- Some participants liked, some disliked:
  - The speed of simulations (in general, those who were familiar with more advanced simulation programs were impressed, while novices thought the system was slow).
  - Temporal maps representation of performance (some preferred a spatial map).
  - The automated design changes (some would have preferred to make their own design changes by hand).

Overall, most of the responses were positive, and the participants seemed to respond well to the expert system design process and interface.

8.5.5 Observed Behavior

During the study, the participants were not asked to explain their design decisions or the choices they made while interacting with the expert system, but many participants chose to talk about their thought process aloud nonetheless. Additionally, the author was able to observe the behavior of the participants when they used the expert system.

One result that became apparent after several user studies had been completed was that each participant behaved differently and made different choices. For example, one participant mentioned explicitly that he or she did not plan to use the expert system to improve performance, but rather to simply explore aesthetic changes that might actually decrease performance. Another participant chose not to change anything about his or her initial design when asked to create a final design, stating that the process of using the expert system had verified his or her original design ideas. Several participants had specific design changes in mind to try out with the expert system and used the tool to test out those options.

In general, the following behavior was observed of nearly all participants:

- They selected a design change that was the first presented by the expert system for the first iteration.
- They selected a design change that *wasn't* the first presented by the expert system for subsequent iterations.
- They selected a design change with lower performance than another option due to aesthetic preference.
- They did not return to a previous iteration of the design (likely due to limited time allotment).

As mentioned previously in section 8.4.2, one unexpected outcome of the study was that no two participants used the expert system in exactly the same way, i.e. all participants made different sets of decisions. One consequence of this behavior was that the final design found by each participant was unique. This result was somewhat surprising because the author assumed that some of the novice users might use the expert system as an optimization method rather than as a design tool by choosing only the first design suggestion given at each step and by always accepting the best performing option. Instead, however, it was found that all participants had strong opinions of their own about which design changes to try and about how their final design should look. This type of behavior helps to demonstrate why many designers would not readily accept a design solution generated by a "black box" algorithm. Instead, the highly interactive nature of the expert system allowed each participant to remain actively involved in the expert system design process by retaining control over design decisions. It is the belief of the author that the mostly positive responses given during the final questionnaire were due to the interactive nature of the expert system tool.

One possible limitation of the user study presented in this chapter is the short amount of time that each participant was able to spend designing and interacting with the expert system. Because the sessions were restricted to a maximum of two hours, the designers may not have been able to respond as creatively as they may have been with more time. In many design situations, the designer is allowed to assess and re-design many times before selecting a final design (see Chapter 3). Therefore, some of the observed behavior may not have been indicative of how participants would have reacted in a less formal and less time-constrained environment. Nevertheless, the study was able to provide a small glimpse at the nature of the human design process and how human designers might respond to a tool such as the expert system.

8.6 Strengths and Limitations of the Expert System for Design

The user study provided valuable feedback on the use of the expert system as a method that can improve performance and as a design tool. Overall, the results of the study were mostly positive. Based on these results, conclusions can be made about the expert system as a user-interactive method and as a potential design tool.

The strengths of the expert system for use as a design tool are the following:

- The expert system was able to find better performing solutions than the initial design even when human interactions and unique sets of design decisions were added into the process.
  - In the second design session, all twelve participants made unique sets of design decisions, producing twelve different designs which all performed better than the initial design.
  - Improved performance occurred even when participants explicitly stated that they were working towards aesthetics that they preferred instead of performance.
- The expert system provided some educational value to most participants.
  - The majority of participants reported learning at least a "small amount" about both the specific design problem at hand and about daylighting in general.
  - Those participants that didn't report learning anything about the specific design problem were able to improve the performances of their designs by an average of over 30%, indicating that they may indeed have learned something but not realized it.
- The expert system positively influenced the final designs of most participants.
  - Most participants reported that their final design was influenced by the process of using the expert system.
  - The aesthetics of many of the final designs are reminiscent of the aesthetics of the initial designs with certain specific elements influenced by the designs produced by the expert system.
  - The mean performance of the set of final designs was closer to the mean performance of the designs produced by the expert system than of the performance of the initial designs.
- By using the expert system, designers with poor initial daylighting intuition may be able to produce designs which perform similarly to those produced by designers with good daylighting intuition.
  - The designers with the lowest initial performances achieved the highest amount of improvement with their final designs.

- Many participants would use the expert system again in a studio or professional design context.

The limitations of the expert system for use as a design tool are the following:

- The expert system only considers performance, not aesthetics.
  - Close to half of the participants responded that the aesthetics of their final design was worse than that of their initial design. The expert system process seems to have influenced some of the participants to sacrifice aesthetics for performance.
- The expert system has a limited number of design changes in its knowledge base that it can offer as suggestions.
  - This limitation was cited by nearly all participants who responded that they probably would not use the expert system for an actual design project.
  - Some participants had design changes in mind that were not offered in the list (for example, adding blinds instead of overhangs).
- The expert system only works on a restricted set of initial geometries.
  - Some participants liked the expert system but did not believe that they could use it for the more complex types of projects that they typically would design.
- Due to the user-interactivity allowed, the expert system will not always improve performance.
  - Users may elect the explore only those design changes which decrease performance, and the expert system currently has no checks in place to discourage users from doing so. This is actually a a positive feature because it allows designers freedom to explore and may provide educational value even if performance is decreased; however, it could also be seen as a limitation in scenarios where designers have a limited amount to time to work with the system and cannot find good solutions efficiently.
- The expert system may not be helpful to those designers who already have very good daylighting intuition. However, the extent of this limitation is strongly dependent on both the daylighting intuition of the designer and the complexity of the design problem, which was about average for the user study.

8.7 Chapter Summary

This chapter presented an evaluation of the expert system as a user-interactive method for performance-driven exploration and as a design tool, based on the results of a user study. During the user study, twelve designers were asked to solve a design problem with multiple daylighting goals, first using their own intuition, and second, using the expert system. The designers were then asked to solve the design problem a third time, again using their own intuition. This study procedure was developed to discover if the expert system had positively influenced the performance and aesthetics of the final designs, as compared to the initial designs. The study participants were also asked to fill out a questionnaire which allowed them to assess their own designs and their experiences using the expert system.

The results of the user study were generally positive and indicated that many of the major goals of the expert system as a user-interactive tool were met. One important result was that every participant was able to find a design with improved performance during their session with the expert system. While Chapter 7 verified that the expert system could successfully work towards improved designs in the absence of a human user, this study demonstrated that the expert system algorithm is also successful when human input is included in the process.

Another important result of the user study was that many of the participants were positively influenced by the process of using the expert system. Most participants also seem to have learned something about the specific design problem which allowed them to intuitively develop better performing designs after they had interacted with the expert system. These results were supported by both the data and by participant response. A final important result is that the majority of the participants responded that they would use the expert system again for a studio or professional design project. These responses demonstrate that the participants may have accepted the expert system as a design tool.

In this chapter, the results of a limited evaluation demonstrated that the expert system was generally successful as a performance-driven design tool and as a method for influencing and educating designers in ways that they can improve the daylighting performance of their designs.

Part IV

Conclusions

Chapter 9

Conclusions and Future Work

9.1 Summary of Results

This dissertation presented a new user-interactive expert system approach which enables architects to consider daylighting goals in the early design stages by engaging them in a performance-driven design exploration process. The main results of this project can be discussed in three parts: 1.) the development of the expert system methodology, including the daylighting knowledge base and fuzzy logic rules; 2.) an assessment of the expert system as a decision-making algorithm, as compared to a true optimization algorithm; and 3.) the application of the expert system as a design tool and as a method which can be integrated into the design process.

9.1.1 Daylighting Expert System Methodology

The expert system developed for this dissertation was a fuzzy rule-based system combined with an external database of previously computed daylighting simulation data, also called the daylighting knowledge base. Below is a summary of each of the major components of the system:

- The daylighting knowledge base, described in detail in Chapter 4, was populated using simulations from a set of 512 models with differing facade characteristics based on the Design of Experiments method. For each model, illuminance and a model-based approximation of the daylight glare probability (DGPm) were calculated in five different zones within the space (and four different views from within each zone for the glare metric), over the whole year. These climate-based metrics were calculated using LSV, the simulation engine native to the Lightsolve program. The knowledge base contains information about the relative effects of ten different facade parameters on each of the two daylighting metrics from the various zones and views within the space.

- The expert system rule base, described in detail in Chapter 5, is a decision-making algorithm that assesses a specific design situation and generates a list of suggested design changes which should improve the current performance. This list is based on a large number of initial inputs, including a 3d model of an original design, user preferences for performance goals and goal priorities, the simulated goal-based performance metrics, and a customized daylighting database, populated using only the most relevant data found in the daylighting knowledge base. The rule base uses fuzzy logic, which allows it to better emulate the human thought process than classical logic, and it has been developed to be a flexible algorithm which can accommodate a wide variety of initial design scenarios. The system was also created in such a way that it requires user interaction and user inputs in order to function.

- The expert system has been implemented within the framework of the Lightsolve project and has a functional, stand-alone interface which allows designers to interact with the system. The interface has been designed to provide a clear and intuitive way of communicating the current performance of a model and the list of design changes suggested by the expert system. The interface also allows designers to view the performance of their design over multiple iterations of the exploration process. Within the system, a simple building data model has been developed to enable automated design changes and to provide additional information about the geometry and material properties of the current design to the decision-making algorithm. The implementation of the system is described in detail in Chapter 6.

9.1.2 Assessment of the Expert System as a Decision-Making Algorithm

The expert system was shown to be successful at making design decisions which improved the daylighting performance of various case study designs, described in detail in Chapter 7. In these case studies, the performances of designs found using the expert system were compared to baseline examples generated by a micro-genetic algorithm (micro-GA). The purpose of the comparison studies was to evaluate the performance of the expert system relative to a known optimization algorithm which could be relied upon to consistently generate designs with very good, if not globally optimal, performance. Nine scenarios were considered, ranging from simple designs with a single performance goal to more complex designs with multiple conflicting goals. The results of these case studies indicated that the expert system was successful at improving the performance of designs for a variety of initial conditions and performance goal scenarios. In some situations, the micro-GA was able to find designs which performed slightly better than those found using the expert system, but this difference in performance was small (7.5% at most for all case studies considered) and acceptable given the fact that the expert system was designed with user-interactivity in mind, which the micro-GA was not.

Two additional short studies were completed which investigated the effect of initial facade conditions and the effect of user-selected window uniformity constraints on the performance of the expert system. In the first study, it was found that although the initial facade design may affect the expert system performance, the system was still able to improve performance for even highly designed facades. Additionally, the difference between the best performing design found using on a generic facade and that which was generated based on a highly designed facade was small (6% for the case studies considered). In the second study, it was found that the window uniformity scheme selected by a user can have a significant effect for certain types of model geometries. For models in which the sensor planes are located parallel to the facades of interest, the selected window uniformity scheme had little effect on the performance of the expert system. For models in which the sensor planes are located perpendicular to the facades of interest, the non-uniform window scheme selection was found to significantly improve expert system performance (by over 10% in the case study considered).

9.1.3 The Expert System as a Design Tool

Because the expert system was developed to be a user-interactive tool, a critical part of this dissertation was to assess the system for use by human designers. Chapter 8 described a user study which was conducted to determine how well the expert system decision-making algorithm would work when independent human interactions and decisions were included into the process. The study also tested how well designers responded to the system and if they were able to benefit from it.

The results of this study were largely positive and indicated that the expert system was successful as a user-interactive design tool. One important result was that all participants were able to improve the performance of an initial design, despite each making a unique set of choices while using the tool. Many participants reported being influenced by the process of using the system, and these responses were verified by the aesthetic choices that the participants made. Many participants also responded that they felt they learned something about the specific design problem and about daylighting in general while using the expert system tool, and this result was confirmed by the performance improvements that were achieved by many participants. Finally, many of the participants claimed that they would use the expert system again for either a studio or professional project, which supports the idea that the expert system may have been accepted by many of the designers as a design tool.

9.2 Future Work

While the expert system was found to be successful at providing a guided design exploration process based on daylighting goals, it also provides a foundation for future work:

- *Expansion of the current system*: The expert system developed for this dissertation was a prototype tool which can consider two daylighting metrics, illuminance and glare, for designs specifically located in Boston, MA (USA). The current system could act as a framework for future expansion in areas such as:

- Addition of more locations, which can be added in one of two ways: a.) the creation of new climate-specific knowledge bases using the method described in Chapter 4, or b.) the creation of a more generic meta-database, which could potentially be designed to be applicable to multiple locations using weight factors based on climate and latitude.
- Addition of a larger set of possible design changes and geometries, which could be added by creating an expanded knowledge base. Such a database could be populated in a similar way to the current knowledge base, using the DoE method with a larger set of variables. Due to the structure of the expert system, which reads the knowledge base data from a text file kept separate from the coded logic, an expanded knowledge base could be added into the current system with only a small amount of editing to the code, mostly to accommodate the addition of new automated design changes. The addition of a significantly larger knowledge base could potentially increase calculation time; however, such an increase would likely be negligible compared to the simulation time.
- Introduction of two-way interaction effects data into the knowledge base. The raw data used to create the knowledge base can yield two-way interaction effects, as described in section 4.7. This data could be introduced into the main effects database directly and each pair could be considered as one "design action" (for example, remove overhangs and make windows larger). Alternatively, it could be used by the expert system decision-making algorithm to better determine which design changes to suggest based on a previously selected design change.

• *Performance metrics for perception and visual interest*: One missing piece of information in the current system is the perceived quality of light, i.e. the level of visual interest that an occupant might have in the space. Although the expert system does produce renderings of views within the space, it is the responsibility of the designer to determine whether the quality of light in the space is high. One possible area of future work is the design of a method which would guide designers towards improved visual interest in addition to daylighting performance. Due to the highly subjective nature of the idea of visual interest and the many different ways in which quality of light may be defined, such research would be difficult to develop. Quality of light is dependent on a variety of factors, including quantitative metrics such as those considered in this thesis as well as more general information such as the type of building, the programmatic use of a space, the cultural context (i.e. the visual preferences of the local population), and the preferences of individual occupants. Other factors which may affect light quality and visual interest include the color(s) of light, the patterns of light distribution across a space, the changing of light distributions over time, and the ways in which the light supports the architectural design and organization of the space.

- *Addition of thermal or energy metrics*: While the current system considers only daylighting performance, it could be possible to expand the system to consider performance in other domains by considering solar thermal gains or building energy use. The solar thermal gains metric can already be calculated using the LSV engine, while building energy use could be calculated using an existing Google SketchUp functionality such as the EnergyPlus plug-in (OpenStudio, 2010). The addition of new metrics would require a substantial expansion of the current system, including the development of new knowledge bases and new fuzzy logic rules to work with the additional information. More research would be required to assess the feasibility of such a scheme.

- *Introduction of the system into the design community:* The current system has been tested in a limited study involving a small group of participants who have design experience but who are all currently working in an academic environment. In the future, it would be beneficial to test the expert system with more designers in more varied settings. For example, the system could be tested by students in a course setting for educational value, or it could be tested by designers who are currently practicing. A goal of the author's is to ultimately introduce the expert system tool in the design community as an extension of the Lightsolve project.

9.3 Broader Outlooks

The earliest expert systems developed for architectural problems were introduced close to 30 years ago. Unfortunately, such systems have yet to gain the popularity of other digital design tools, such as 3d modeling or simulation programs. At this stage in time, there exists a very large number of digital building design and simulation tools, and they cover a tremendous range in terms of simulation domains, features and functionalities, ease of use, and target user type. At present, certain types of "intelligent" tools are gaining popularity, such as Building Information Models (BIMs) and computer-based design methods such as algorithmic, generative, and parametric design. Designers are increasingly looking towards computer programs to aid in their design process, and there is an opportunity for expert systems to become more prevalent in architectural practice.

There are many ways in which expert systems could be useful to designers today. However, any new systems must consider the digital design environment and the demands of the modern designer. One way in which this thesis attempted to create a more designer-friendly expert system tool was to combine it with a known 3d modeler, Google SketchUp, and to allow users to model their own design instead of relying on a set of generic choices. Because the method proposed in this thesis was created to be relevant to specific designs, it was necessary to rely on a simulation engine which many of the user study participants found too slow for use in a professional environment. A possible alternative to the proposed system could be a very fast expert system tool which uses generic designs and is able to provide feedback in many different domains at once (for example, daylighting, energy consumption, energy costs, and LEED compliance). Instead of calculating quantitative data, such a system could generate qualitative diagnostic information, similarly to the Leso-DIAL tool (Paule and Scartezzini, 1997). A quick diagnostic expert system could quickly gain popularity if it could be combined with a popular BIM, such as Revit (Autodesk, 2010).

Another possibility for the future of expert systems is to use such systems to help designers work more accurately with simulation programs. Because expert systems are traditionally rule-based, they are well-suited to diagnostic problems, such as determining when a user may have input incorrect values into a simulation program or when the simulation results are not of the correct order of magnitude based on a building's program, square footage, and climate. Simple diagnoses could also be included to guide designers towards better design decisions at a more general level than the method presented in this thesis.

Finally, as shown in this thesis, expert systems may be valuable as tools which can engage designers and interactively guide them towards improved performance. One possibility for including expert systems into the design process in a different way is to combine them with other types of guided digital design processes, such as parametric or generative design. While some parametric design methods, such as DIVA (Lagios et al., 2010), have already begun to include performance information, there are no current systems which feed performance information back into the algorithm itself, and there is great potential for expert systems to be used as a guide for parametric algorithms to help them explore better performing designs. As the expert system proposed in this study was only able to explore relatively traditional architectural forms (floorplans and window shapes with orthogonal angles only), a system which combines an expert system with a parametric tool would have the additional benefit of being able to explore more interesting forms.

As a final note, the questions of how designers interact with expert systems and how such tools can be evaluated should be considered more broadly. This dissertation included a short study involving designers to evaluate the proposed daylighting expert system tool, but more work in this area would be beneficial for all new systems. Such research could consist of surveys or interviews with designers, or studies or workshops in which designers are asked to interact with digital tools. The results could be used to determine how researchers can make future expert systems more designer-friendly and thus, more likely to be used.

9.4 Final Remarks

This dissertation has presented a user-interactive expert system for improving daylighting in architectural designs. Although much of this work has focused on the development of the method and accompanying tool, it is important to point out that the position of the architect as the final decision maker in any design process was fundamental to the development of this work. The expert system was designed to respect the designer and to support his or her process by providing performance information and intelligent design proposals. Ultimately, the goal of this work was to inspire the creative process in a novel way, not by developing a tool that would replace the role of the designer, but by establishing a new type of dialogue between the designer and the expert system tool.

Part V

Appendices

Appendix A

Expert System Modeling Guidelines

In order for the expert system to understand a 3d massing model correctly, the model must conform to the following guidelines.

- Modeling a Space:
  - All walls, floors, and ceilings must be opaque. Windows must be transparent or translucent.
  - The normal vectors of walls, windows, floors, and ceilings should point towards the interior of the space.
  - Facades must all face cardinal directions and must be orthogonal to each other.
  - All models must be watertight.
  - At present, interior walls can be modeled and simulated correctly but the expert system will not recognize or consider them.
- Special Materials:
  - Any plane that represents an external shading device must have the word "EXTERNAL" in its material name. External shading devices must be completely opaque.
  - Any plane that represents a window must have the word "GLASS" in its material name. Users may specify normal transmissivity using the opacity value of the material in SketchUp (for example, opacity of 20% will result in glass transmissivity of 80%). Alternatively, users may specify both specular and diffuse transmissivity using the following naming convention: "GLASS_Sx_Dy", where x is the percentage of regular transmissivity and y is the percentage of diffuse transmissivity. The sum of x and y cannot exceed 100%. For example, "GLASS_S60_D20" refers to a glass type which have 80% total transmissivity, which includes 60% specular transmission and 20% diffuse transmission.

- Modeling Sensors:
  - The normal vectors of sensors should point towards the direction in which illuminance or glare will be measured (for example, to measure illuminance on a horizontal surface from above, the sensor should point upwards).
  - Illuminance sensor planes must be vertical or horizontal 2D planes and must include the word "SENSOR" in their material names. A valid material name for an illuminance sensor is "SENSOR_Classroom."
  - Glare sensor planes must be vertical 2D planes and must include the term "GLARE_SENSOR" in their material names. A valid material name is "SENSOR_SouthView."
  - Sensors may be opaque or transparent. Opaque sensors will block light like normal opaque surfaces, while transparent sensors will measure illuminance or glare over a surface which does not block light. A transparent sensors must be modeled with its material opacity set to 0% and must include the word "INVISIBLE" in its material name. For example, valid transparent sensor names are "INVISIBLE_SENSOR_Classroom" and "INVISIBLE_SENSOR_SouthView."

Appendix B

User Study Materials

This appendix contains the materials used during the user studies, including the design briefs, template sheets, and questionnaires.

Design Session 1

You are an architect in a design firm and have been assigned to work through a conceptual design for the façade of a **school library wing** for a building in **Boston, MA**. The client wishes to use natural light instead of artificial light as much as possible.

You will be taking over the project from a colleague who has already started working on the design. You have been told that you must keep the original massing model (footprint, wall heights, and interior walls).

You may change the **façade elements** as much as you need to in order to create a design which meets the daylighting goals. This means that you can choose the size and placement of windows, the types of glass used, and the types, size, and placement of shading devices. The two façades you will be working with face **South** and **East**.

The client has indicated that a certain aesthetic must be maintained:
- Windows must be rectangular or square.
- Glass may be normal or translucent, but not tinted or colored.
- Shading devices must be opaque, and must be vertical or horizontal.
- Both vertical and horizontal shading devices may be used on the same window.

It is up to you to determine if the windows on each façade should all be the same or if some of them are different. No advanced systems may be used.

The library space has three main areas:

1. A double-height main study area, which should receive lots of light
2. A smaller study area which overlooks the main study area, which should receive an adequate amount of light
3. A rare book room, in which light must be carefully controlled

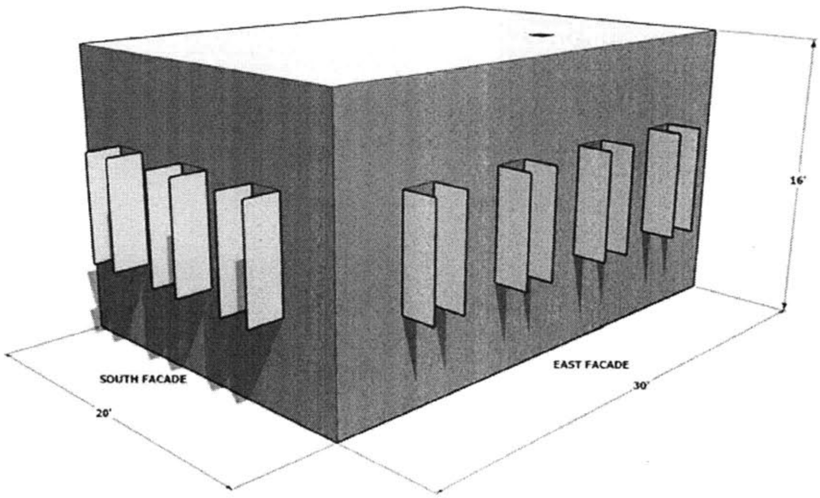
On the attached sheet, you will find an image of the current design, along with the specific illuminance goals you should try to meet and the location of the illuminance sensors within the space. Sensors are all at workplane height.

Please spend **20 minutes** working on a design for this space. At the end of this time, you will need to draw your final facade design on the provided **template sheet** and submit it to the investigator.

You may use any of the provided materials if desired; however, it is not required that you do so. You should mainly be designing using your own knowledge and intuition about daylighting. Please remember that this design is one intended to achieve the daylighting goals <u>and</u> also satisfy you as the designer.

Figure B.1: Design Session 1 brief, page 1.

Current Design



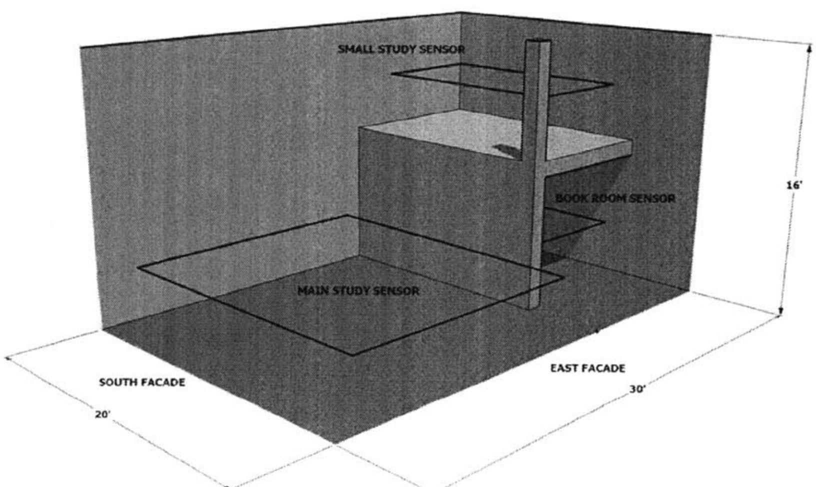
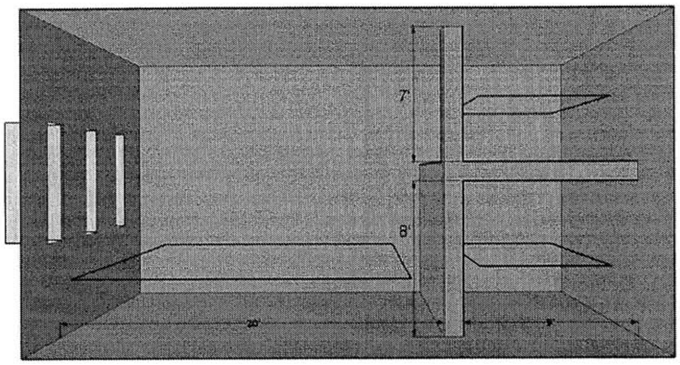
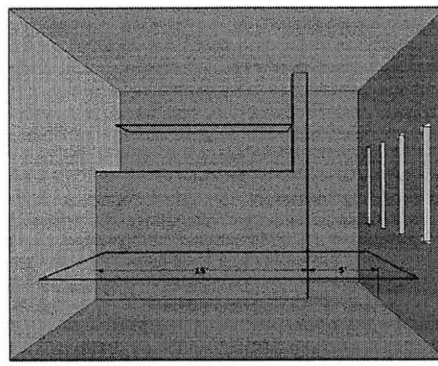


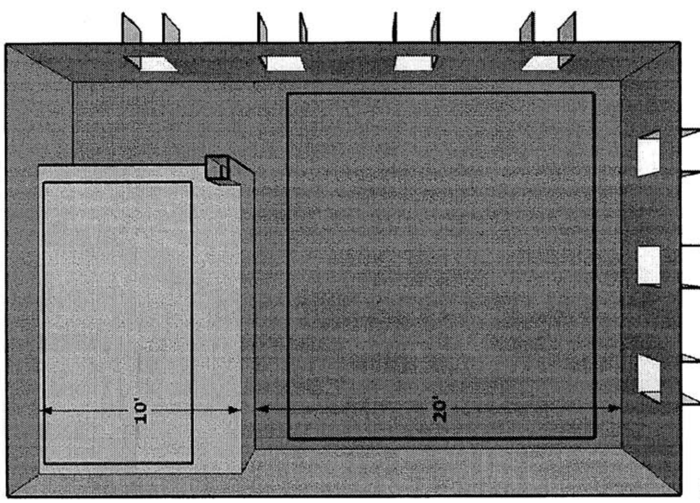
Figure B.2: Design Session 1 brief, page 2.



Section (looking towards West)



Section (looking towards North)



Section (looking from top)

Illuminance Goals for Library

Based on the client's description of the space, your firm has decided that the specific daylighting goals you should work towards are:

1. Main Study Space:
   a. Minimum illuminance levels: 500 lx (desired) down to 400 lx (acceptable)
   b. No maximum illuminance levels
2. Small Study Space
   a. Minimum illuminance levels: 200 lx (desired) down to 0 lx (acceptable)
   b. Maximum illuminance levels: 800 lx (desired) up to 1000 lx (acceptable)
3. Rare Book Room
   a. No minimum illuminance levels
   b. Maximum illuminance levels: 200 lx (desired) up to 400 lx (acceptable)

If the illuminance on the entire area of a sensor plane falls within the desired range during all daylit times of the year, the performance of that sensor will be 100%.

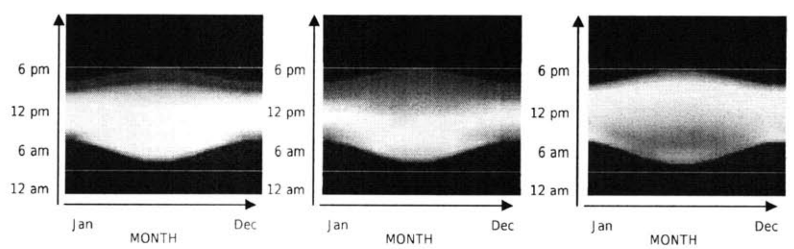
Performance of Current Design

The current design has been previously simulated to evaluate performance.

The average performance over the whole year is:

Main Study Space:	80%
Small Study Space	65%
Rare Book Room:	70%
Average of All Spaces:	**72%**

The temporal maps for each space look like this:



Main Study Room | Small Study Room | Rare Book Room

Red indicates illuminance that is too high on the sensor plane.
Yellow indicates illuminance in range on the sensor plane.
Blue indicates illuminance that is too low on the sensor plane.

Figure B.4: Design Session 1 brief, page 4.

Design Session 2

Your supervisor has just come by your desk to tell you that she would like you to try out a new **performance-based design system** that her friends at MIT have developed.

The system will start with an original design and suggest design changes that should improve the daylighting performance. If you select a design change, the system will automatically make that change for you in the model and simulate the performance. You will be able to see the performance of the current design at all times.

Your supervisor would like you to put your design aside for awhile and start over with your colleague's previous design. She says that you may accept or decline any design change, perform as many design changes as you'd like within the time limit, or go backwards to a previous design if you find that you do not like where the design process takes you.

Please interact only with the interface for this exercise, and do not attempt to change the model yourself in SketchUp.

You are allowed **40 minutes** to use the design system. At the end of that time, your supervisor would like you to choose your favorite design and report it to the investigator.

Please note:
It is more likely that a suggested design change will improve performance if you do not skip the first suggested change or if you only skip once.

Simulations may take <u>up to 10 minutes</u> until completion. While simulations are running, you are welcome to relax, sketch, or read any of the provided reading materials. Please refrain from using your cell phone or other devices during this time.

If you encounter any difficulties using the software, please do not hesitate to ask the investigator for assistance.

Figure B.5: Design Session 2 brief.

Design Session 3

Your supervisor would like you to submit a **final design** to move into the schematic design phase. You are allowed to re-visit your original design, to re-visit the design that was generated using the expert system, or to completely start over.

You are allowed **15 minutes** to create this final design. At the end of this time, you will need to draw your final design on the provided **template sheet** and submit it to the investigator.

As before, you may use any of the provided materials if desired; however, it is not required that you do so. You should mainly be designing using your own knowledge and intuition about daylighting. Please remember that this design is one intended to achieve the daylighting goals <u>and</u> also satisfy you as the designer.

Figure B.6: Design Session 3 brief.

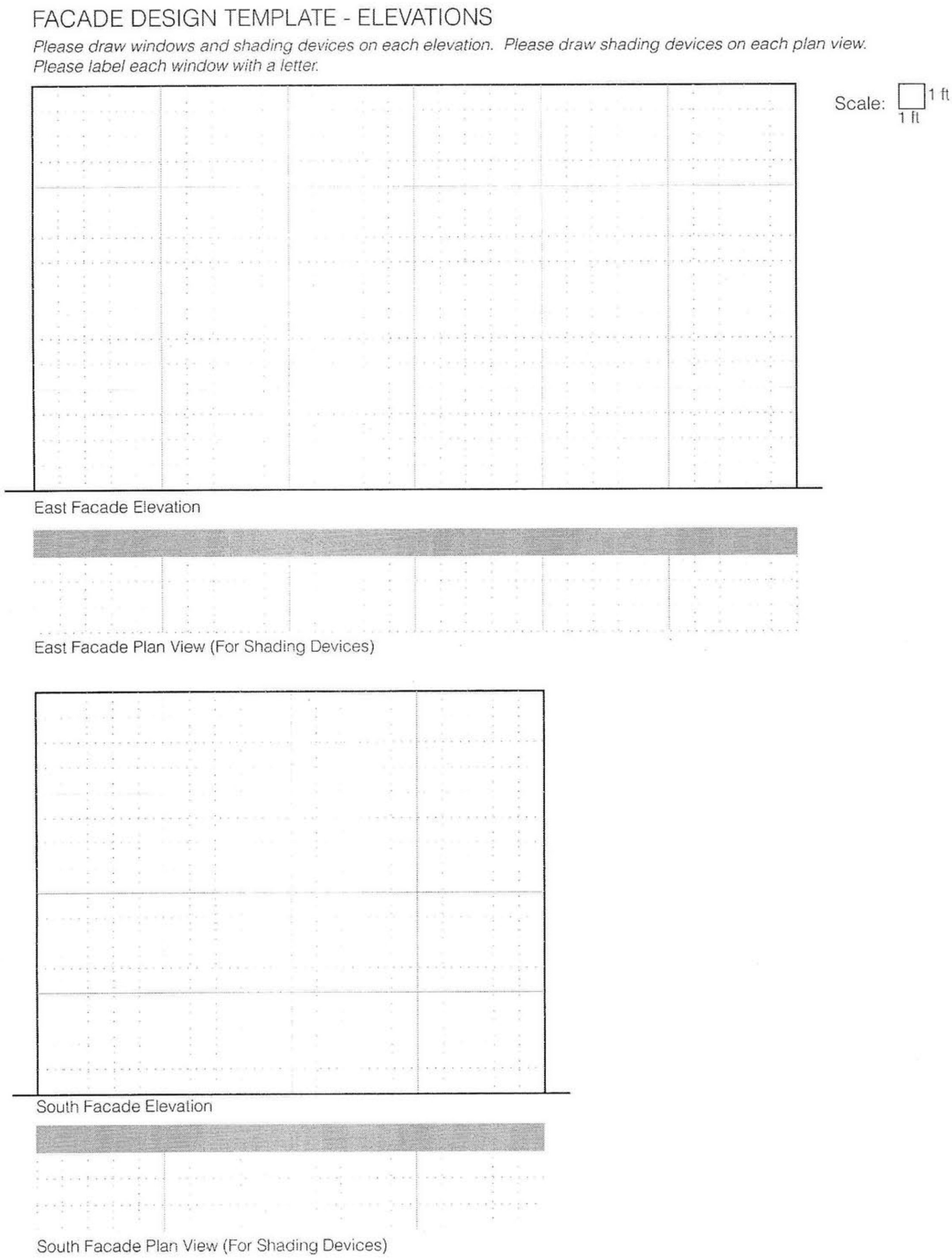


Figure B.7: Design template, sheet 1.

FACADE DESIGN TEMPLATE - SHADING DEVICES AND GLASS TYPES

Specify the appropriate windows using the letters from the elevation drawings.

Scale: ☐ 1 ft × 1 ft

Shading Devices - Draw one section per shading device type.

Window(s): ____________________

Window(s): ____________________

Window(s): ____________________

Glass Types - Check the appropriate box in each section. Use one column per glass type.

View (Choose One):

☐ Transparent (Clear View)

☐ Translucent/Frosted (Blurred View)

☐ Opalescent/Milky (No view)

Amount of Light Let In (Choose One):

☐ Most (For ex. Single-Glazed Clear)

☐ Intermediate (For ex. Double-Glazed Low-E)

☐ Least (For ex. Grey Tinted)

Window(s): ____________________

View (Choose One):

☐ Transparent (Clear View)

☐ Translucent/Frosted (Blurred View)

☐ Opalescent/Milky (No view)

Amount of Light Let In (Choose One):

☐ Most (For ex. Single-Glazed Clear)

☐ Intermediate (For ex. Double-Glazed Low-E)

☐ Least (For ex. Grey Tinted)

Window(s): ____________________

View (Choose One):

☐ Transparent (Clear View)

☐ Translucent/Frosted (Blurred View)

☐ Opalescent/Milky (No view)

Amount of Light Let In (Choose One):

☐ Most (For ex. Single-Glazed Clear)

☐ Intermediate (For ex. Double-Glazed Low-E)

☐ Least (For ex. Grey Tinted)

Window(s): ____________________

Figure B.8: Design template, sheet 2.

INTRODUCTORY SURVEY:

I. Please tell us about your experiences in design, daylighting, and simulation:

1. Please circle Yes or No:

a. I have completed one or more architectural studio courses. Yes / No
If Yes, how many?_________________

b. I have experience working in an architecture firm. Yes / No
If Yes, for how long? _______________

c. I have considered daylighting during one or more studio projects. Yes / No

d. I have considered daylighting during one or more professional projects. Yes / No

e. I have used daylighting simulation for one or more studio projects. Yes / No

f. I have used daylighting simulation for one or more professional projects. Yes / No

2. Please select the most appropriate statement:

a. ___ I consider myself to be an experienced designer.
___ I consider myself to be an intermediate designer.
___ I consider myself to be a novice designer.
___ I do not consider myself to be a designer.

b. ___ I consider myself to be experienced at working with daylighting.
___ I consider myself to be intermediate at working with daylighting.
___ I consider myself to be a novice at working with daylighting
___ I do not consider myself to be someone who works with daylighting.

Figure B.9: Introductory questionnaire, page 1.

3. Please rate your level of experience using the following simulation programs by checking the appropriate box (please check only one per row):

Software Title	I consider myself an expert at using this program.	I have used this program frequently and am comfortable using it.	I have used this program infrequently and am comfortable using it.	I have used this program infrequently and I am not comfortable using it.	I have heard of this program but never used it.	I have never heard of this program.
Radiance						
DaySim						
Ecotect						
AGI32						
3D Studio Max Mental Ray Daylight						
Lightscape						
LESODial						
Lightsolve						
Other (please specify: __________)						

4. Do you use any daylighting simulation programs during the early design stages (conceptual or schematic)? If so, which one(s)?

5. Do you use any daylighting simulation programs during the later design stages (design development)? If so, which one(s)?

6. Do you use any daylighting simulation programs for analysis only during the final stage of design? If so, which one(s)?

Figure B.10: Introductory questionnaire, page 2.

II. Please tell us about your education level:

1. I have previously completed (please check all that apply):
__ B.Arch.
__ B.A. or B.S. in Architecture
__ B.A. or B.S. in Architecture with Building Technology concentration
__ B.A. or B.S. in Other Major (please specify): _________________
__ M.Arch.
__ M.A. or M.S. in Architecture
__ MS in Building Technology, Building Science, or similar
__ SMarchS (please specify which area): ________________
__ Other Master's level degree (please specify): _______________
__ PhD in Architecture
__ PhD in Building Technology
__ Other degree not listed (please specify): _________________

2. I am currently working towards (please check all that apply):
__ B.Arch.
__ B.A. or B.S. in Architecture
__ B.A. or B.S. in Architecture with Building Technology concentration
__ B.A. or B.S. in Other Major (please specify): _________________
__ M.Arch.
__ M.A. or M.S. in Architecture
__ MS in Building Technology, Building Science, or similar
__ SMarchS (please specify which area): ________________
__ Other Master's level degree (please specify): _______________
__ PhD in Architecture
__ PhD in Building Technology
__ Other degree not listed (please specify): _________________

3. Please select the most appropriate statement (please check one):

__ I have never taken a course that involves daylighting in architecture
__ I have learned about daylighting in one or more studio or survey courses
__ I have taken one or more courses devoted exclusively to daylighting

Figure B.11: Introdcutory questionnaire, page 3.

FINAL QUESTIONNAIRE

1. Given the aesthetic constraints, when compared to my first design, I feel that my final design is...

__ Much more aesthetically pleasing
__ Somewhat more aesthetically pleasing
__ About the same
__ Somewhat less aesthetically pleasing
__ Much less aesthetically pleasing

Comments:

2. Given the daylighting goals, when compared to my first design, I feel that my final design performs...

__ Much better
__ Somewhat better
__ About the same
__ Somewhat worse
__ Much worse

Comments:

3. I feel that the process of using the expert system...

__ Greatly influenced my final design
__ Somewhat influenced my final design
__ Did not influence my final design

Comments:

Figure B.12: Final questionnaire, page 1.

4. Do you feel that by using the expert system, you learned something new which helped you approach the design problem?

__ Yes, I learned a large amount.
__ Yes, I learned a small amount.
__ No, I did not learn anything new.

Comments:

5. Do you feel that by using the expert system, you learned something new about daylighting in general?

__ Yes, I learned a large amount.
__ Yes, I learned a small amount.
__ No, I did not learn anything new.

Comments:

6. Would you consider using the expert system for a studio project? *Please explain your answer.*

__ Definitely
__ Probably
__ Undecided
__ Probably not
__ Definitely not

Please explain:

Figure B.13: Final questionnaire, page 2.

7. Would you consider using the expert system for a professional project? *Please explain your answer.*

__ Definitely
__ Probably
__ Undecided
__ Probably not
__ Definitely not

Please explain:

8. What did you like about the expert system?

9. What did you dislike about the expert system?

10. Any additional comments?

Figure B.14: Final questionnaire, page 3.

Bibliography

M. Adamski. Optimization of the form of a building on an oval base. *Building and Environment*, 42(4):1632–1643, Apr. 2007.

O. Akin. *Psychology of Architectural Design*. Pion Ltd, 1986.

L. B. Alberti. *On the Art of Building in Ten Books, translated by Joseph Rykwert*. MIT Press, Cambridge, MA, 1988.

R. J. Allwood, M. R. Shaw, J. L. Smith, D. J. Stewart, and E. G. Trimble. Building dampness diagnosing the causes. *Building Research & Information*, 16(1):37–42, 1988.

M. Andersen, S. Kleindienst, L. Yi, J. Lee, M. Bodart, and B. Cutler. An intuitive daylighting performance and optimization approach. *Building Research & Information*, 36(6):593–607, Nov. 2008.

G. Augenbroe. Integrated building performance evaluation in the early design stages. *Building and Environment*, 27(2):149–161, Apr. 1992.

Autodesk. Autodesk revit architecture. http://usa.autodesk.com/adsk/servlet/pc/index?id=3781831&siteID=123112, 2010. URL `http://usa.autodesk.com/adsk/servlet/pc/index?id=3781831&siteID=123112`.

K. M. Beauchemin and P. Hays. Sunny hospital rooms expedite recovery from severe and refractory depressions. *Journal of Affective Disorders*, 40(1-2):49–51, Sept. 1996.

M. Black. Vagueness. an exercise in logical analysis. *Philosophy of Science*, 4(4):427–455, Oct. 1937.

M. Bodart and A. D. Herde. Global energy savings in offices buildings by the use of daylighting. *Energy and Buildings*, 34(5):421–429, June 2002.

P. Boyce, C. Hunter, and O. Howlett. The benefits of daylight through windows. Technical report, U.S. Department of Energy, Sept. 2003a.

P. R. Boyce, J. H. Heerwagen, C. C. Jones, J. A. Veitch, and G. R. Newsham. Lighting quality and office work: A field simulation study. Technical report, Pacific Northwest National Laboratory, 2003b.

A. M. Brintrup, H. Takagi, and J. Ramsden. Evaluation of sequential, multi-objective, and parallel interactive genetic algorithms for multi-objective floor plan optimisation. In *Applications of Evolutionary Computing*, Lecture Notes in Computer Science, pages 586–598. Springer-Verlag, Berlin, 2006.

S. Bund and E. Y. L. Do. SPOT! fetch light: Interactive navigable 3D visualization of direct sunlight. *Automation in Construction*, 14(2):181–188, Mar. 2005.

L. Caldas. Generation of energy-efficient architecture solutions applying GENE_ARCH: an evolution-based generative design system. *Advanced Engineering Informatics*, 22(1): 59–70, Jan. 2008.

L. G. Caldas and L. K. Norford. A design optimization tool based on a genetic algorithm. *Automation in Construction*, 11(2):173–184, Feb. 2002.

P. W. Canada. Daylighting guide for canadian commercial buildings. Technical report, Public Works and Government Services Canda (PWGSC), Ottawa, Canada, 2002.

G. Carrara and G. Novembri. Knowledge assistants in the process of architectural design. *Building and Environment*, 25(3):199–207, 1990.

D. L. Carroll. Genetic algorithms and optimizing chemical oxygen-iodine lasers. *Developments in theoretical and applied mechanics*, 18(3):411–424, 1996.

R. Charron and A. K. Athienitis. Optimization of the performance of double-facades with integrated photovoltaic panels and motorized blinds. *Solar Energy*, 80(5):482–491, May 2006.

S. C. Chase. A model for user interaction in grammar-based design systems. *Automation in Construction*, 11(2):161–172, Feb. 2002.

S. C. Chase. Generative design tools for novice designers: Issues for selection. *Automation in Construction*, 14(6):689–698, Dec. 2005.

T. Chastain, Y. E. Kalay, and C. Peri. Square peg in a round hole or horseless carriage? reflections on the use of computing in architecture. *Automation in Construction*, 11(2): 237–248, Feb. 2002.

K. Chau and F. Albermani. A knowledge-based system for liquid retaining structure design with blackboard architecture. *Building and Environment*, 40(1):73–81, Jan. 2005.

H. Chen, R. Ooka, and S. Kato. Study on optimum design method for pleasant outdoor thermal environment using genetic algorithms (GA) and coupled simulation of convection, radiation and conduction. *Building and Environment*, 43(1):18–30, Jan. 2008.

F. Chlela, A. Husaunndee, C. Inard, and P. Riederer. A new methodology for the design of low energy buildings. *Energy and Buildings*, 41(9):982–990, Sept. 2009.

CHPS. High performance schools best practices manual volume II: design. Technical report, 2006.

H. W. Chun and E. M. K. Lai. Intelligent critic system for architectural design. *IEEE Transactions on Knowledge and Data Engineering*, 9(4):625–639, Aug. 1997.

A. Chutarat. *Experience of Light: The Use of an Inverse Method and a Genetic Algorithm in Daylighting Design.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2001.

CIBSE. *Code for Lighting.* Butterworth-Heinemann, Oxford, UK, 2002.

C. A. C. Coello and G. P. Pulido. A micro-genetic algorithm for multiobjective optimization. In *Evolutionary Multi-Criterion Optimization*, volume 1993 of *Lecture Notes in Computer Science*, pages 126–140. 2001.

D. A. Coley and S. Schukat. Low-energy design: combining computer-based optimisation and human judgement. *Building and Environment*, 37(12):1241–1247, Dec. 2002.

L. Corbusier. *Towards a New Architecture, translated by Frederick Etchells.* Dover, New York, 1986.

B. Cutler, Y. Sheng, S. Martin, D. Glaser, and M. Andersen. Interactive selection of optimal fenestration materials for schematic architectural daylighting design. *Automation in Construction*, 17(7):809–823, Oct. 2008.

E. de Groot, L. Zonneveldt, and B. Paule. Dial europe: A decision support tool for early lighting design. In *Proceedings of Building Simulation*, pages 421–426, Eindhoven, Netherlands, 2003.

E. Y. L. Do and M. D. Gross. Let there be light! Knowledge-Based 3-D sketching design tools. *International Journal of Architectural Computing*, 2(2):211–227, June 2004.

J. G. Doheny and P. F. Monaghan. IDABES: an expert system for the preliminary stages of conceptual design of building energy systems. *Artificial Intelligence in Engineering*, 2(2): 54–64, Apr. 1987.

H. Doukas, K. D. Patlitzianas, K. Iatropoulos, and J. Psarras. Intelligent building energy management system using rule sets. *Building and Environment*, 42(10):3562–3569, Oct. 2007.

Z. Du and X. Jin. Detection and diagnosis for multiple faults in VAV systems. *Energy and Buildings*, 39(8):923–934, Aug. 2007.

C. L. Dym, R. P. Henchey, E. A. Delis, and S. Gonick. A knowledge-based system for automated architectural code checking. *Computer-Aided Design*, 20(3):137–145, Apr. 1988.

L. Edwards and P. A. Torcellini. *A literature review of the effects of natural light on building occupants*. National Renewable Energy Laboratory, 2002.

P. Fazio, C. Bedard, and K. Gowri. Knowledge-based system approach to building envelope design. *Computer-Aided Design*, 21(8):519–527, Oct. 1989.

T. Fischer, M. Burry, and J. Frazer. Triangulation of generative form for parametric design and rapid prototyping. *Automation in Construction*, 14(2):233–240, Mar. 2005.

D. M. Freedman, M. Dosemeci, and K. McGlynn. Sunlight and mortality from breast, ovarian, colon, prostate, and non-melanoma skin cancer: a composite death certificate based case-control study. *Occupational & Environmental Medicine*, 59(4):257, 2002.

B. R. Gaines. Foundations of fuzzy reasoning. *International Journal of Man-Machine Studies*, 8(6):623–668, Nov. 1976.

A. D. Galasiu and C. Reinhart. Current daylighting design practice: a survey. *Building Research & Information*, 36(2):159–174, Mar. 2008.

F. J. Garijo and L. A. de Garrido. A knowledge based system for house design. In *Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics*, pages 806–809, Aug. 1988.

P. Geyer. Multidisciplinary grammars supporting design optimization of buildings. *Research in Engineering Design*, 18(4):197–216, Jan. 2008.

M. Golabchi. A knowledge-based expert system for selection of appropriate structural systems for large spans. *Asian Journal of Civil Engineering*, 9(2):179–191, 2008.

D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

Google. Google SketchUp. http://sketchup.google.com/, 2008. URL `http://sketchup.google.com/`.

E. Gullichsen and E. Chang. Generative design in architecture using an expert system. *The Visual Computer*, 1(3):161–168, Nov. 1985.

B. Guo, C. Belcher, and W. Roddis. RetroLite: an artificial intelligence tool for lighting energy-efficiency upgrade. *Energy and Buildings*, 20(2):115–120, 1993.

C. M. Herr and T. Kvan. Adapting cellular automata to support the architectural design process. *Automation in Construction*, 16(1):61–69, Jan. 2007.

L. Heschong, R. L. Wright, and S. Okura. Daylighting impacts on human performance in school. *Journal of the Illuminating Engineering Society*, 31(2):101–111, 2002.

R. Hitchcock and W. Carroll. DElight: a daylighting and electric lighting simulation engine. In *Proceedings of Building Simulation*, pages 483–488, Eindhoven, Netherlands, Aug. 2003.

C. A. Hviid, T. R. Nielsen, and S. Svendsen. Simple tool to evaluate the impact of daylight on building energy consumption. *Solar Energy*, 82(9):787–798, Sept. 2008.

R. L. Hwang, T. P. Lin, H. H. Liang, K. H. Yang, and T. C. Yeh. Additive model for thermal comfort generated by matrix experiment using orthogonal array. *Building and Environment*, 44(8):1730–1739, Aug. 2009.

P. Ihm, A. Nemri, and M. Krarti. Estimation of lighting energy savings from daylighting. *Building and Environment*, 44(3):509–514, Mar. 2009.

I. Jaffal, C. Inard, and C. Ghiaus. Fast method to predict building heating demand based on the design of experiments. *Energy and Buildings*, 41(6):669–677, June 2009.

T. Jung, M. Gross, and E. Do. Light pen, sketching light in 3D. In *Proceedings of the 10th International Conference on Computer Aided Architectural Design Futures*, Tainan, Taiwan, 2003.

Y. E. Kalay. Performance-based design. *Automation in Construction*, 8(4):395–409, Apr. 1999.

J. K. Kawai, J. S. Painter, and M. F. Cohen. Radioptimization: goal based rendering. In *SIGGRAPH 93 Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, pages 147–154, 1993.

S. Kleindienst and M. Andersen. The adaptation of daylight glare probability to dynamic metrics in a computational setting. In *Proceedings of LuxEuropa*, Istanbul, 2009.

S. Kleindienst, M. Bodart, and M. Andersen. Graphical representation of climate-based daylight performance to support architectural design. *LEUKOS - The Journal of the Illuminating Engineering Society of North America*, 5(1):39 – 61, July 2008.

M. Kolokotroni, S. Robinson-Gayle, S. Tanno, and A. Cripps. Environmental impact analysis for typical office facades. *Building Research & Information*, 32(1):2–16, Jan. 2004.

K. Krishnakumar. Micro-genetic algorithms for stationary and non-stationary function optimization. *SPIE Proceedings: Intelligent Control and Adaptive Systems*, 1989.

labs.plugins.ro. SketchUp bridge. http://plugins.ro/labs/bridge_tut.htm, 2006. URL `http://plugins.ro/labs/bridge_tut.htm`.

K. Lagios, J. Niemasz, and C. F. Reinhart. Animated building performance simulation (ABPS) - linking Rhinoceros/Grasshopper with Radiance/Daysim. In *Proceedings of Simbuild*, New York City, Aug. 2010.

W. Lam. *Sunlighting as Formgiver for Architecture*. Van Nostrand Reinhold, New York, 1986.

G. W. Lambert, C. Bedard, D. M. Kaye, G. L. Jennings, and M. D. Esler. Effect of sunlight and season on serotonin turnover in the brain. *The Lancet*, 360(9348):1840–1842, Dec. 2002.

B. Lawson. *How Designers Think: The Design Process Demystified*. Architectural Press, 2006.

J. H. Lee. Optimization of indoor climate conditioning with passive and active methods using GA and CFD. *Building and Environment*, 42(9):3333–3340, Sept. 2007.

J. M. Lee, M. Andersen, Y. Sheng, and B. Cutler. Goal-based daylighting design using an interactive simulation method. In *Proceedings of Building Simulation*, Glasgow, Scotland, 2009.

E. S. Lefkowitz and C. F. Garland. Sunlight, vitamin d, and ovarian cancer mortality rates in US women. *International Journal of Epidemiology*, 23(6):1133–1136, Dec. 1994.

M. Lehar and L. Glicksman. Rapid algorithm for modeling daylight distributions in office buildings. *Building and Environment*, 42(8):2908–2919, Aug. 2007.

D. H. W. Li and J. C. Lam. An investigation of daylighting performance and energy saving in a daylit corridor. *Energy and Buildings*, 35(4):365–373, May 2003.

S. P. Lin and F. Albermani. Lattice-dome design using a knowledge-based system approach. *Computer-Aided Civil and Infrastructure Engineering*, 16(4):268–286, July 2001.

H. Liu, T. Igusa, and B. W. Schafer. Knowledge-based global optimization of cold-formed steel columns. *Thin-Walled Structures*, 42(6):785–801, June 2004.

G. F. Luger. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Addison Wesley, Boston, 5th edition, 2004.

J. Lukasiewicz. Philosophical remarks on many-valued systems of propositional logic. *Polish Logic*, 1939:40–63, 1930.

L. Lutton. HYPEREX - a generic expert system to assist architects in the design of routine building types. *Building and Environment*, 30(2):165–180, Apr. 1995.

M. L. Maher. Expert systems for structural design. *Journal of Computing in Civil Engineering*, 1(4):270–283, 1987.

A. M. Malkawi, R. S. Srinivasan, Y. K. Yi, and R. Choudhary. Decision support and design evolution: integrating genetic algorithms, CFD and visualization. *Automation in Construction*, 14(1):33–44, Jan. 2005.

T. A. Markus. The significance of sunshine and view for office workers. In *Sunlight in Buildings*. Boewcentrum International, Rotterdam, The Netherlands, 1967.

A. Marsh. ECOTECT: shading design. http://www.squ1.com/products/ecotect/features/shadingdesign, 2008a. URL `http://www.squ1.com/products/ecotect/features/shadingdesign`.

A. Marsh. Ecotect. http://www.squ1.com/products/ecotect, 2008b. URL `http://www.squ1.com/products/ecotect`.

J. Marx. A proposal for alternative methods for teaching digital design. *Automation in Construction*, 9(1):19–35, Jan. 2000.

P. K. Mashood, C. S. Krishnamoorthy, and K. Ramamurthy. KB-GA-Based hybrid system for layout planning of multistory buildings. *Journal of Computing in Civil Engineering*, 21 (4):229–237, July 2007.

H. Matsumoto and S. Toyoda. A knowledge-based system for condensation diagnostics in houses. *Energy and Buildings*, 21(3):259–266, 1994.

J. Michalek, R. Choudary, and P. Papalambros. Architectural layout design optimization. *Engineering Optimization*, 34(5):461–484, 2002.

P. Mitusch. Expert system for the norwegian building regulations. *Building Research & Information*, 17(4):223–227, 1989.

M. Monks, B. M. Oh, and J. Dorsey. Audioptimization: Goal-Based acoustic design. *IEEE Computer Graphics and Applications*, 20(3):76–90, June 2000.

D. C. Montgomery. *Design and Analysis of Experiments*. John Wiley & Sons, 6 edition, 2004.

R. Mora, H. Rivard, S. Parent, and C. Bedard. Interactive Knowledge-Based assistance for conceptual design of building structures. *Advances in Engineering Structures, Mechanics & Construction*, 140(5):571–582, 2006.

M. Negnevitsky. Fuzzy expert systems. In *Artificial Intelligence: A Guide to Intelligent Systems*, pages 87–129. Addison-Wesley, Essex, UK, 2nd edition, 2005.

C. E. Ochoa and I. G. Capeluto. Advice tool for early design stages of intelligent facades based on energy and visual comfort approach. *Energy and Buildings*, 41(5):480–488, May 2009.

J. O'Connor, E. Lee, F. Rubinstein, and S. Selkowitz. Tips for daylighting with windows: The integrated approach. Technical Report LBNL-39945, Lawrence Berkeley National Laboratory, Berkeley, CA, 1997.

K. H. Oey and E. Passchier. Complying with practice codes. *Building Research & Information*, 16(1):30–36, 1988.

OpenStudio. OpenStudio plug-in for google SketchUp. http://apps1.eere.energy.gov/buildings/energyplus/openstudio.cfm, 2010. URL `http://apps1.eere.energy.gov/buildings/energyplus/openstudio.cfm`.

P. Y. Papalambros and D. J. Wilde. *Principles of Optimal Design: Modeling and Computation*. Cambridge University Press, 2nd edition, 2000.

C. S. Park, G. Augenbroe, and T. Messadi. Daylighting optimization in smart facade systems. In *Proceedings of Building Simulation*, pages 1001–1008, Eindhoven, Netherlands, 2003.

C. S. Park, G. Augenbroe, N. Sadegh, M. Thitisawat, and T. Messadi. Real-time optimization of a double-skin facade based on lumped modeling and occupant preference. *Building and Environment*, 39(8):939–948, Aug. 2004.

M. L. Parra-Saldivar and W. Batty. Thermal behaviour of adobe constructions. *Building and Environment*, 41(12):1892–1904, Dec. 2006.

G. Patow, X. Pueyo, and A. Vinacua. User-guided inverse reflector design. *Computers & Graphics*, 31(3):501–515, June 2007.

L. F. Pau and S. S. Nielsen. A knowledge-based system for computer-aided architectural design for energy savings and thermal comfort. *Artificial Intelligence for Engineering, Design, Analysis and Manufacturing*, 4(2):71–88, 1990.

B. Paule and J. Scartezzini. Leso-DIAL, a new computer-based daylighting design tool. In *Proceedings of Right Light 4*, pages 93–97, Copenhagen, Denmark, 1997.

A. J. Perelson. *iPlot: an intelligent lighting design assistant.* PhD thesis, Massachusetts Institute of Technology, 2005.

R. Perez, J. Michalsky, and R. Seals. Modeling sky luminance angular distribution for real sky conditions: experimental evaluation of existing algorithms. *Journal of the Illuminating Engineering Society*, 21(2):84–92, 1992.

S. Petersen and S. Svendsen. Method and simulation program informed decisions in the early stages of building design. *Energy and Buildings*, 42(7):1113–1119, July 2010.

P. Plympton, S. Conway, and K. Epstein. Daylighting in schools: improving student performance and health at a price schools can afford. In *Proceedings of the American Solar Energy Society Conference*, Madison, Wisconsin, June 2000.

J. Pohl and L. Myers. A distributed cooperative model for architectural design. *Automation in Construction*, 3(2-3):177–185, July 1994.

M. Rashid and C. Zimring. A review of the empirical literature on the relationships between indoor environment and stress in health care and office settings: Problems and prospects of sharing evidence. *Environment and Behavior*, 40(2):151–190, Mar. 2008.

M. S. Rea. *The IESNA Lighting Handbook: Reference & Application.* Illuminating Engineering Society of North America, 2000.

G. Reichard and K. Papamichael. Decision-making through performance simulation and code compliance from the early schematic phases of building design. *Automation in Construction*, 14(2):173–180, Mar. 2005.

C. F. Reinhart. A simulation-based review of the ubiquitous window-head-height to daylit zone depth rule-of-thumb. In *Proceedings of Building Simulation*, Montreal, Canada, 2005.

C. F. Reinhart and A. Fitz. Findings from a survey on the current use of daylight simulations in building design. *Energy & Buildings*, 38(7):824–835, July 2006.

C. F. Reinhart and V. M. LoVerso. A rules of thumb based design sequence for diffuse lighting. *Lighting Research & Technology*, 42(1):7–31, Mar. 2010.

C. F. Reinhart and J. Wienold. The daylighting dashboard - a simulation-based design analysis for daylit spaces. *Building and Environment*, 46(2):386–396, Feb. 2011.

C. Robin, J. Brau, and J. Roux. Integration of expert knowledge and simulation tools for the thermal design of buildings and energy systems. *Energy and Buildings*, 20(2):167–175, 1993.

M. A. Rosenman and J. S. Gero. Design codes as expert systems. *Computer-aided design*, 17 (9):399–409, 1985.

G. G. Roy. A comparative study of lighting simulation packages suitable for use in architectural design. Technical report, Murdoch University, Perth, Australia, 2000.

J. H. Rutherford and T. W. Maver. Knowledge-based design support. *Automation in Construction*, 3(2-3):187–202, July 1994.

A. R. Sabouni and O. M. Al-Mourad. Quantitative knowledge based approach for preliminary design of tall buildings. *Artificial Intelligence in Engineering*, 11(2):143–154, Apr. 1997.

P. Sachdeva. DAMP - a diagnostic system for architectural moisture damage problems. *Australian Computer Journal*, 17(1):27–32, 1985.

T. Sarawgi. Survey on the use of lighting design software in architecture and interior design undergraduate education. *International Journal of Architectural Computing*, 4(4): 91–108, Dec. 2006.

C. Schoeneman, J. Dorsey, B. Smits, J. Arvo, and D. Greenburg. Painting with light. In *SIGGRAPH 93 Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, pages 143–146, 1993.

D. A. Schon. *The Reflective Practitioner: How Professionals Think in Action*. Basic Books, 1983.

H. T. Seok, J. H. Jo, and K. W. Kim. Establishing the design process of double-skin facade elements through design parameter analysis. *Journal of Asian Architecture and Building Engineering*, 8(1):251–258, 2009.

R. Shacked and D. Lischinski. Automatic lighting design using a perceptual quality metric. *Computer Graphics Forum*, 20(3):215–227, Sept. 2001.

E. Shaviv, A. Yezioro, I. G. Capeluto, U. J. Peleg, and Y. E. Kalay. Simulations and knowledge-based computer-aided architectural design (CAAD) systems for passive and low energy architecture. *Energy and Buildings*, 23(3):257–269, Mar. 1996.

K. Shea, R. Aish, and M. Gourtovaia. Towards integrated performance-driven generative design tools. *Automation in Construction*, 14(2):253–264, Mar. 2005.

W. Siler and J. J. Buckley. *Fuzzy expert systems and fuzzy reasoning*. John Wiley and Sons, 2005. ISBN 9780471388593.

M. Sugeno. *Industrial Applications of Fuzzy Control*. North-Holland, 1985.

K. Talbott. Context-sensitivity and visual engagement in generative systems. *Automation in Construction*, 16(1):54–60, Jan. 2007.

J. E. Tena and I. R. Goldberg. An interactive system for solving inverse illumination problems using genetic algorithms. *Proceedings of Computacion Visual*, 1997.

S. L. Torres and Y. Sakamoto. Facade design optimization for daylight with a simple genetic algorithm. In *Proceedings of Building Simulation*, Beijing, China, 2007.

B. J. Urban and L. R. Glicksman. A rapid building energy model and interface for non-technical users. In *Proceedings of the 10th ORNL Thermal Performance of the Exterior Envelopes of Whole Buildings International Conference*, Clearwater, FL, Dec. 2007.

U.S.DOE. 2009 buildings energy data book. http://buildingsdatabook.eren.doe.gov/, 2009. URL `http://buildingsdatabook.eren.doe.gov/`.

E. Vartiainen, K. Peippo, and P. Lund. Daylight optimization of multifunctional solar facades. *Solar Energy*, 68(3):223–235, Mar. 2000.

Vitruvius. *The Ten Books on Architecture*. Dover, New York, 1960.

J. M. Walch, B. S. Rabin, R. Day, J. N. Williams, K. Choi, and J. D. Kang. The effect of sunlight on postoperative analgesic medication use: A prospective study of patients undergoing spinal surgery. *Psychosomatic Medicine*, 67(1):156–163, 2005.

W. Wang, R. Zmeureanu, and H. Rivard. Applying multi-objective genetic algorithms in green building design optimization. *Building and Environment*, 40(11):1512–1525, Nov. 2005.

A. R. Webb. Considerations for lighting in the built environment: Non-visual effects of light. *Energy and Buildings*, 38(7):721–727, July 2006.

M. Wetter. GenOpt - a generic optimization program. In *Proceedings of Building Simulation*, Rio de Janeiro, Aug. 2001.

M. Wetter and E. Polak. Building design optimization using a convergent pattern search algorithm with adaptive precision simulations. *Energy & Buildings*, 37(6):603–612, June 2005.

J. Wienold. Dynamic simulation of blind control strategies for visual comfort and energy balance analysis. In *Proceedings of Building Simulation*, Beijing, China, 2007.

J. Wienold. Dynamic daylight glare evaluation. In *Proceedings of Building Simulation*, Glasgow, Scotland, 2009.

J. Wienold and J. Christoffersen. Evaluation methods and development of a new glare prediction model for daylight environments with the use of CCD cameras. *Energy and Buildings*, 38(7):743–757, July 2006.

F. Wright. *An American Architecture*. Horizon Press, New York, 1955.

J. Wright and M. Mourshed. Geometric optimization of fenestration. In *Proceedings of Building Simulation*, Glasgow, Scotland, 2009.

L. Xiyu, T. Mingxi, and J. H. Frazer. An eco-conscious housing design model based on co-evolution. *Advances in Engineering Software*, 36(2):115–125, Feb. 2005.

I. C. Yeh. Architectural layout optimization using annealed neural network. *Automation in Construction*, 15(4):531–539, July 2006.

L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, June 1965.

E. Znouda, N. Ghrab-Morcos, and A. Hadj-Alouane. Optimization of mediterranean building design using genetic algorithms. *Energy & Buildings*, 39(2):148–153, Feb. 2007.