Integrated Control of Indoor Environmental Quality

R-296-000-102-112: Final Report
July 2010

Principal Investigator: Dr. Benny Raphael

Collaborators:
Associate Professor Tham Kwok Wai
Associate Professor Sekhar S. Chandra
Mr. Tan Cheow Beng
Professor I.F.C. Smith

Department of Building
National University of Singapore

Integrated Control of Indoor Environmental Quality

Final Report

This research project is funded by a research grant - R-296-000-102-112 from the National University of Singapore.

Members of the research Team

Principal Investigator:

Dr. Benny Raphael, Dept of Building, NUS

Collaborators:

Associate Professor Tham Kwok Wai, Dept of Building, NUS
Associate Professor Sekhar S. Chandra, Dept of Building, NUS
Mr. Tan Cheow Beng, Dept of Building, NUS
Professor I.F.C. Smith, Institute of Applied Mechanics and Computing, EPFL.

Graduate Student Researcher:

Mr. Chen Yixing, Dept of Building, NUS

Executive Summary

A framework and a methodology for the integrated control of IEQ parameters through multi-objective optimization were developed in this research. A new control strategy that maximizes building performance has been developed. The most important aspect of the strategy is the treatment of the task as a multi-criteria global optimization problem. The best control action is selected by the simultaneous evaluation of multiple objectives. This is a new development with respect to current industry practice. Control strategies that are commonly adopted in the industry apply local actions which affect a single parameter at a time. In this research, global search is employed for selecting solutions that achieve reasonable trade-offs among parameters when there are conflicting objectives. This approach makes it possible to include criteria such as energy consumption, user preferences and indoor environment quality in the control strategy.

Three new algorithms and their software implementations have been developed in this project. These are:

- MOPD – an algorithm for generating Pareto Front for multi-objective optimization
- RR-PARETO – an algorithm for selecting the best solution from a Pareto Front that achieves reasonable trade-offs between conflicting objectives
- NUS-CBRFIT – a machine learning algorithm using case based reasoning and statistical learning

In addition to the above modules, an interface to lighting and energy simulation software has been developed. This module called NUS-ILES permits both simulations to be done using a single building model expressed in the XML format.

The methodology as well as individual components have been tested and evaluated using several test cases involving both full scale working systems and hypothetical examples. Three control applications have been developed using this methodology. The first one is in the domain of control of window blinds. The second application deals with adaptable light shelves. The third application controls personalized ventilation.

Laboratory prototype of controllable window blind shows that the system performs well and exhibits patterns of behavior that are intuitive. The system is shown to take actions that conserve energy while respecting user preferences. Full-scale implementation of the personalized ventilation system has been rated highly in a series of subject studies that were conducted in this research.

Prototype implementations of window blind control and personalized ventilation control demonstrate that integrated control using multi-objective optimization is technically feasible. Tests conducted on the prototypes as well as hypothetical test cases show significant potential for energy savings.

Acknowledgements

This research is supported by the Singapore Ministry of Education's AcRF Tier 1 funding and the Office of Research, (ORE), NUS, through the grant R-296-000-102-112. This project started with and reached this stage through exchange of ideas with my collaborators Prof. Tham Kwok Wai, Prof. Chandra Sekhar, Mr. Tan Cheow Beng and Prof. Ian Smith. Their contributions to the ideas in this report are invaluable. The help and support of Mr. Tan Cheow Beng during the hardware installation and experimental setup is specially acknowledged. The work of Mr. Chen Yixing related to personalized ventilation subject studies is also gratefully acknowledged. Mr. Jovan Pantelic, Ms. Li Ruixin and Dr. Henry Wilhem Cahyady provided valuable advice during various stages of this project. The timely support of Mr. Sin Jia Hau and his team at CNA was crucial for the successful completion of the laboratory prototypes.

Table of contents

List of Figures

List of Tables

1 Introduction

Automatic controllers promise to improve energy efficiency and reduce maintenance costs. For example, a system called NEUROBAT (Morel et al., 2001) developed at the Swiss Federal Institute of Technology, Lausanne (EPFL) obtained energy savings of 13% (compared to conventional open loop controllers) in two occupied rooms during a complete heating season. Other examples of the use of automatic controllers for thermal comfort can be found in Chen (2001), Kummert (2000) and Chow et al. (2002). Guillemin and Morel (2002) developed a prototype that simultaneously controlled heating and lighting. Since such systems are still being tested in laboratories, they have not penetrated into industry. Commercial HVAC systems operate through taking actions sequentially in order to meet targets related to individual parameters. Integrated control of diverse indoor environmental quality (IEQ) parameters such as heating, lighting, humidity and CO2 do not exist as commercial products.

In most of the systems described above, control is treated as a single objective optimization problem. Multi-objective optimization approaches such as Pareto optimization have never been investigated for the control of IEQ. These techniques have been successfully applied to other domains such as structural control (Adam and Smith., 2005). Recent theoretical advances in the area of multi-objective control make it possible to identify solutions that achieve reasonable trade-offs amongst conflicting objectives such as minimizing energy and maximizing comfort.

Parameters that influence indoor environment quality such as temperature, humidity, level of CO2 and lighting can be adjusted through operating different types of devices. But it is not easy to obtain the right combinations of values of these parameters since operating a single device might have several effects. For example, suppose that the objective is to improve the level of lighting in a room. When other objectives are ignored, the most economic solution will be to open window blinds. However, it brings more radiant heat into the room and consequently, cooling load increases. When there are multiple objectives, we need to search for solutions that make compromises among conflicting objectives. Multi-objective optimization techniques make it possible to identify such solutions. These techniques have not yet been applied to the control of building systems.

The objective of this project is to develop a methodology for the integrated control of IEQ parameters through multi-objective optimization. Primary contributions of this research are in the area of total building performance and building automation; Innovative control strategies that aim to maximize building performance have been developed and tested in this project. Secondary contributions are in the area of applied computer science. This project illustrates the potential of advanced computing concepts such as machine learning and multi-objective search in the field of IEQ. These contributions have direct relevance to building automation and control which is assuming increasing importance in the industry which is moving towards intelligent facilities management.

1.1 Summary of research accomplishments

A methodology for the integrated control of IEQ parameters such as temperature, humidity, level of CO2 and lighting has been developed. Such an integrated approach is not currently being used in the industry. Control strategies that are commonly adopted in the industry apply local actions which affect individual parameters. In this research, global search is employed for selecting solutions that achieve reasonable trade-offs among parameters when there are conflicting objectives. This approach makes it possible to include criteria such as energy consumption, user preferences and indoor environment quality in the control strategy.

Advanced machine learning techniques are used to train models for predicting the effects of control actions. The model is calibrated through data from sensors during the training phase. This model is used to select the best control action in a given situation through global search.

The methodology has been demonstrated using the prototype implementation of a fully functional window-blind control system. The generality of the methodology is demonstrated by its application to another full-scale prototype implementation in the Field Environmental Chambers (FEC) of the department of building, NUS, for the control of personalized ventilation.

Results from this study will contribute towards the design and construction of “intelligent buildings”. Companies that specialize in facilities management and building systems have already expressed interest in these technologies and there is potential for commercial exploitation.

1.2 Structure of this report

Chapter 2 discusses the multi-objective optimization framework and the methodology that has been developed in this project. Hardware and software details related to the implementation are described in Chapter 3. Chapter 4 contains the description of a prototype implementation in the area of control of window blinds. Chapter 5 discusses the application of the methodology to the control of a new type of light shelves. The application of the methodology to the control of personalized ventilation is described in Chapter 6. Chapter 7 contains conclusions.

References

Adam, B. and Smith, I.F.C. (2006). Tensegrity Active Control: a Multi-Objective Approach, in press, J Computing in Civil Engineering.

Chen T., (2001). Real-time predictive supervisory operation of building thermal systems with thermal mass, Energy and Buildings, 33(2), pp. 141-150.

Chow T., Zhang G., Lin Z., and Song C., (2002,). Global optimization of absorption chiller system by genetic algorithm and neural network, Energy and Buildings, 34(1), pp. 103-109.

B. Domer, B. Raphael, K. Shea and I.F.C. Smith, (2003). A study of two stochastic search methods for structural control, Journal of Computing in Civil Engineering, (JCCE), Vol 17, No 3, pp 132-141.

Guillemin A., and Morel N., (2002). Experimental results of a self-adaptive integrated control system in buildings: a pilot study, Solar Energy, 72(5), pp. 397-403.,

Kummert M., Andre P., and Nicolas J., (2000). Optimal heating control in a passive solar commercial building, In proceedings of the eurosun 2000 Congress, Copenhagen, Denmark.

Morel N., Bauer M., El-Khoury M., and Krauss J., (2001). Neurobat, a predictive and adaptive heating control system using artificial neural networks, International journal of solar energy, 21, pp. 161-201.

Raphael B. and Smith I.F.C., (2003). Fundamentals of computer aided engineering, John Wiley and sons, Ltd, (306 pages).

2 Control and Optimization Framework

Summary

A framework and a methodology for the integrated control of IEQ parameters through multi-objective optimization were developed in this research. A new control strategy that maximizes building performance has been developed. The most important aspect of the strategy is the treatment of the task as a multi-criteria global optimization problem. The best control action is selected by the simultaneous evaluation of multiple objectives.

Three new algorithms and their software implementations have been developed in this project. These are:

- MOPD – an algorithm for generating Pareto Front for multi-objective optimization
- RR-PARETO – an algorithm for selecting the best solution from a Pareto Front that achieves reasonable trade-offs between conflicting objectives
- NUS-CBRFIT – a machine learning algorithm using case based reasoning and statistical learning

MOPD is a new algorithm that has been developed in this project for generating the Pareto front. This algorithm divides a multi-objective optimization problem into a series of single objective optimization problems that are easier to solve. This is done by selecting one objective at a time and treating remaining objectives as constraints. Tests have shown that this algorithm is able to generate good quality Pareto fronts.

RR-PARETO selects the best solution from the Pareto set by using information related to the relative importance of objectives and their sensitivity. The performance of this algorithm is empirically evaluated using a test case involving trade-offs between lighting and cooling energy.

NUS-CBRFIT is used to predict the responses (output variables) when the values of input variables are known. This algorithm can be used when predictive models are not available or to increase the speed of predictions when simulations are slow. The machine learning model is trained using data from previous simulations or using measurement data from sensors. Tests conducted indicate that the algorithm can significantly speed up calculations and the accuracy is high.

In addition to the above modules, an interface to lighting and energy simulation software has been developed. This module called NUS-ILES permits both simulations to be done using a single building model expressed in the XML format. The use of these modules is demonstrated through the prototype implementations described in later chapters.

2.1 Introduction

The objective of a control system is to achieve a set of objectives by taking actions based on measurements obtained through sensors. The measured parameters help to compute, directly or indirectly, the effect of actions and evaluate how far the current state is from the desired ideal situation in which all the objectives are satisfied perfectly.

In general, control is an abductive task (Raphael and Smith, 2003a) or in the terminology of mathematicians, an inverse problem. It involves working backwards against the normal direction of flow of information. The normal flow of information in physical systems is from cause to effect. That is, given the causes, the effects can be predicted from physical principles (using deduction). On the other hand, in control, the desired effects are known and the task is to compute the causes that might produce these effects. This is illustrated in Figure 2.1.

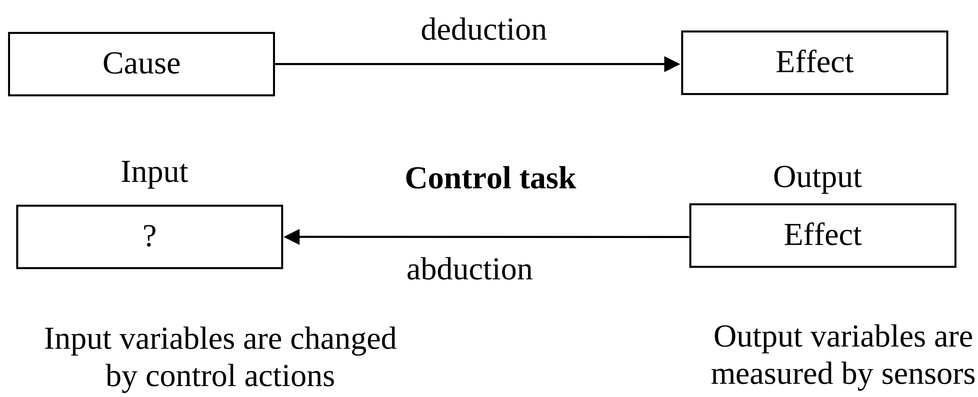


Figure 2.1 Illustration of control as an abductive task.

In general, there is no direct method to compute the causes from the effects. This makes control a difficult problem to solve. The control actions have to be determined through trial and error, or in other words, through procedures involving generation and testing of solutions. The task is further complicated by the non-uniqueness of the solution. There might be several causes that produce the same effects.

The following sections describe issues related to the development of a control system. This is followed by a description of the control system framework developed in this project (Section 2.2).

2.1.1 Traditional control strategies

Traditionally control systems in buildings are designed such that a single action directly affects only a single parameter. Several examples can be found in building systems. The room temperature is controlled by opening or closing dampers that regulate the air supply to the room. The CO2 level in a room is kept below recommended limits by operating the damper that limits the fresh air intake to the Air Handling Unit (AHU). The cooling load supply is adjusted to the demand by operating the valve that regulates the supply of chilled water to the AHU. In all these situations, a control action is taken based on the value of a single parameter. In this case, the control becomes a relatively simple task.

In practice, a complex control system is implemented by chaining together a series of controllers that operate locally using a single parameter at a time. The system works by the cascading of events which are initiated by environmental or human factors. For example, when the occupancy load increases, the room temperature increases. This causes dampers to open supplying more air to the room. This increases the cooling load in the AHU causing the chilled water supply valve to open more and so on.

When a single input parameter controls a single output parameter and when the relationship is monotonic, traditional algorithms such as PID (Proportional Integral Derivative) (Tan et al., 1999) are very effective. PID is extensively used in the control of HVAC appliances. This is mainly used to maintain set points of parameters such as temperature. Fuzzy versions of this algorithm has also been used recently. For example, see Alcala et al. (2005), Ziva et al. (2008), and Kolokosta (2006).

2.1.2 Search and optimization in control

The control task can be formulated as a global optimization problem. In a global optimization problem, an objective function is minimized (or maximized) subject to a set of constraints. The general global optimization problem is mathematically defined as following: Given a set **D** (feasible domain) and a function *f(x)* defined over **D,**

Minimize $f(x)$,
subject to the constraints $x \in \mathbf{D}$.

The function *f(x)* is called the objective function. The variable *x* represents an *n–dimensional* vector of unknown values.

A global optimization formulation makes it possible to achieve complex objectives such as minimization of energy or maximization of comfort. These objectives involve several parameters which cannot be handled using conventional simple algorithms such as PID. However, the drawback is that the control algorithm tends to be more complex and it may not be possible to implement them on simple hardware that are commonly used in traditional control systems involving Direct Digital Controllers (DDC) or Programmable Logic Controllers (PLC). Complex optimization routines require considerable amount of memory and processing power and may only be executed on modern digital computers. However, this is no more a drawback since computers have become relatively cheap and open standards and protocols have been developed. International standards such as BACNET have made the task of interfacing computers with controllers relatively easy. The prototype implementation of a control system that is described in Chapter 4 illustrates this point.

2.1.3 Multi-parameter search in the area of building performance

Building performance can be evaluated using a number of parameters such as energy, indoor environment quality (IEQ), perception of user comfort etc. Total Building Performance (TBP) is a holistic framework for evaluating the performance of a building from the most important considerations. Currently, it exists mainly as a set of guidelines for the design and evaluation of buildings. Putting this into operation in terms of a set of precise rules or an algorithm that can be applied automatically may not be feasible. However, the control of building systems requires automatic procedures for evaluating alternative options and selecting the best one. This is quite a difficult task when all the performance mandates are incorporated.

The perception of indoor environment quality is influenced by many aspects such as thermal comfort, air quality, and lighting. Air quality may be measured through a number of parameters such as relative humidity (RH), level of CO2, velocity of air movement, etc. Traditionally, HVAC systems control the air quality and thermal comfort using parameters set by users. In most cases, the only parameter that the end-user can set is the temperature. The air conditioning system provides adequate supply of cold air into the room such that the required temperature is attained. The amount of cold air supplied to a room is regulated by opening and closing of dampers to various levels that are determined by the control algorithm. The supply of cold air indirectly influences other parameters such as RH and CO2. Accurate simultaneous control of these parameters is not possible through this approach. In fact, these parameters are not even measured in the conventional systems.

It is quite well known that parameters that influence the Indoor Environment Quality (IEQ) strongly interact with each other. For example, increasing the natural daylight in a room might also increase the amount of heat transmitted. Increasing the supply of fresh air to improve the air quality will also increase the energy consumption. In practice, such conflicting requirements are managed by following limits sets by codes of practices and international standards. While, these standards provide general guidelines, taking appropriate decisions in specific situations is difficult.

There are many researchers who have studied the influence of interacting parameters on the indoor environment quality, for example, Diakaki et al. (2008), Suter et. al. (2007); and Mahdavi and Unzeitig (2005). In these works, multiple parameters are modelled for the purpose of simulation and performance evaluation, but not for adaptive control. Multi-objective optimization approaches such as Pareto optimization have never been investigated for the control of IEQ. These techniques have been successfully applied to other domains such as structural control (Adam and Smith., 2005). Recent theoretical advances in the area of multi-objective control make it possible to identify solutions that achieve reasonable trade-offs amongst conflicting objectives such as minimizing energy and maximizing comfort. Some approaches to multi-criteria search are described in the next Section.

2.1.4 Multi-criteria search

Multi-criteria search and optimization techniques are more widely adopted in design tasks. Conventionally, designs are optimized with respect to a single objective such as the total cost. In many engineering disciplines, this formulation is inflexible and impractical because designs have to satisfy conflicting requirements of different parties involved in the design process. All the requirements may not be combined into a single objective. In such situations, a multi-objective (or multi-criteria) formulation is attractive. In the multi-objective formulation, several objectives are simultaneously considered. The solution to a multi-objective optimization problem consists of a set of solutions which represent reasonable trade-offs among different objectives.

There are several approaches to multi-criteria search and optimization. One popular approach is Pareto optimization. In this approach, a population of solutions known as the Pareto set is generated (Raphael and Smith, 2003a). The Pareto set (front) consists of a set of solutions that satisfy what is known as the Pareto optimality criterion. According to this criterion, a solution point P (Figure 2.2) is accepted only if there are no solutions better than P with respect to all the objectives. For example, even if P is worse than another solution P1 with respect to one objective, P is accepted provided that it is better than P1 in at least one objective. Thus each Pareto optimal solution is good in some respect. The set of all Pareto optimal solutions form the Pareto set or Pareto front.

Many techniques for generating the Pareto front are found in the literature. These include multi-objective versions of genetic algorithms (Haralampidis et al., 2005, Deb and Tiwari, 2005, Grierson and Khajehpour, 2002), simulated annealing (Czyzak and Jaszkiewicz, 1998, Ulungu et al., 1999), weighting methods (Kim and Weck, 2006), and multi-start methods (Jaszkiewicz 2002). However, no mathematical proof exists that these methods do converge to the true Pareto optimal front.

Once the Pareto front is generated, a suitable solution point can be selected using a number of heuristics. A simple heuristic is to select the point on the normalised Pareto surface where the slope is close to 45 degrees. This is the point where the relative gain in one objective is compensated by an equivalent loss in the other objective. These simple techniques may not be applicable in all the situations.

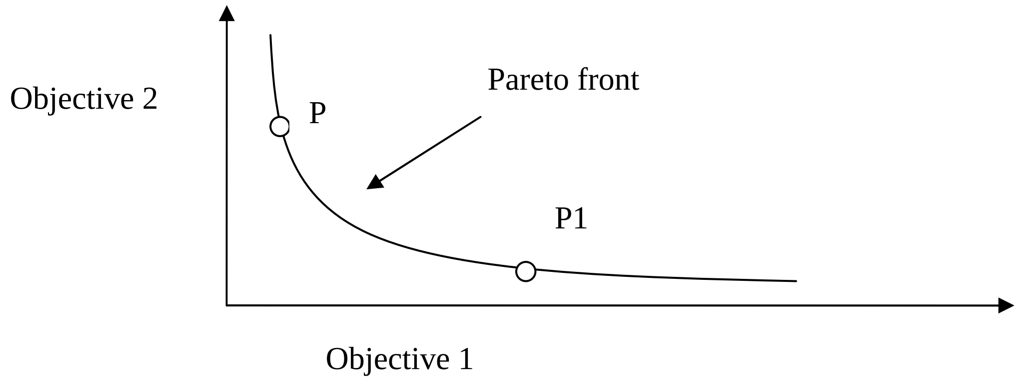


Figure 2.2. Pareto Front.

2.1.5 Machine learning in control

When global optimization is used to identify best control actions, we need to define an objective function. The objective function computes the difference between the desired state and the predicted state after the application of a control action. This requires an accurate prediction of the effects of control actions. In general, the effects of control actions cannot be predicted accurately since closed form equations are not available and theoretical models involve many parameters whose values are not known precisely. In these circumstances, machine learning techniques are used to train models to be used for prediction. Two types of models are used. The first type is based on physical principles, but contains parameters whose values are not known precisely. The values of parameters are determined through a process known as model updating in which the best combination of values of parameters is found through the minimization of prediction error. The second type of models is purely empirical and might be obtained through machine learning techniques such as artificial neural networks.

An example of the first type of model is the energy balance equation which predicts the change in the average temperature of the room due to the opening of the damper by a unit amount. Parameters in the model whose values are unknown include the heat influx into the room, internal heat gain, temperature and velocity of supply air, etc.

An example of an empirical model is the computation of air velocity for a specific opening of the damper installed in an air supply duct. Accurate determination of air velocity requires complex analysis using computational fluid dynamics and involves quite detailed modeling of damper characteristics which may not be available. However, a reasonable approximation might be obtained in a given setting through empirical observations, including previous observations where air velocities have been determined.

Model updating procedures have been proposed for improving the accuracy of predictions. In the model updating procedure proposed by Robert-Nicoud et. al. (2005), the error between model predictions and measurements is minimized using a global search algorithm called PGSL (Raphael and Smith, 2003b). The procedure results in the identification of a population of candidate models. The characteristics of this population are studied in order to evaluate the reliability of identification and to determine whether a unique model explains the data.

Conventionally, model identification is treated as an optimization problem. The model whose prediction has the minimum error with respect to measurements is chosen as the best model. It has been shown in Robert-Nicoud et. al. (2005) that this procedure results in the identification of wrong models in the presence of modelling and measurement errors. This is because different types of errors might compensate for each other such that the next error is very small. In order to avoid this problem, it is important to compute the error threshold through an estimate of modelling errors and sensor precision. All the models whose predictions lie below this threshold are chosen as candidate models. An examination of the variation in the values of model parameters will reveal whether a unique identification is possible. For example, if model parameters show wide variation, it means that many different models match observations within the error threshold. In such a situation, we need to install more sensors or increase the precision of sensors.

2.2 Integrated Control of IEQ

In this research, a new methodology for the integrated control of indoor environment quality has been developed. The highlight of this methodology is the treatment of the control task as a multi-objective optimization problem. Instead of optimizing a single objective, multiple objectives are considered simultaneously. There are objectives related to attaining desirable values of IEQ parameters as well as minimization of energy. The goal of active control is to achieve the objectives to the best possible extent. Since objectives frequently conflict with each other, multi-objective optimization should aim at achieving reasonable trade offs. For example, it may not be possible to attain the set values for both the temperature and RH, since excess humidity is usually removed by cooling the air which tends to lower the temperature below the set values. In such situations, the optimization algorithm identifies the best control action that will achieve reasonable compromises.

The general schematic of the methodology is shown in Figure 2.3. IEQ parameters are evaluated through data from sensors. This data is used by the multi-objective optimization module to select the best control actions that reasonably satisfy objectives and constraints. The selected actions are applied by sending commands to appropriate actuators. Responses to these actions are recorded by sensors and are input into the optimization module to complete the feedback loop. It should be noted that unlike conventional control systems, there is no direct relationship between one sensor and one actuator. All the subsystems are evaluated simultaneously in order to identify globally optimal solutions.

The sensors, objectives and constraints shown in Figure 2.3 are illustrative and representative. In specific installations, all the systems and sensors shown in the figure may not be available. The objectives and constraints might also be different. Nevertheless, conceptually all the available systems and sensors are considered simultaneously for the identification of the best control action at any point. The objectives and constraints are defined by the user for each application, and these involve all the available sensors and building systems in that installation. This idea is illustrated using prototype implementations described in subsequent chapters.

The most important component of the methodology is the multi-criteria optimization module. This is described in Section 2.3. Optimization involves the use of predictive models. Issues related to the development and training of models are described in Section 2.4.

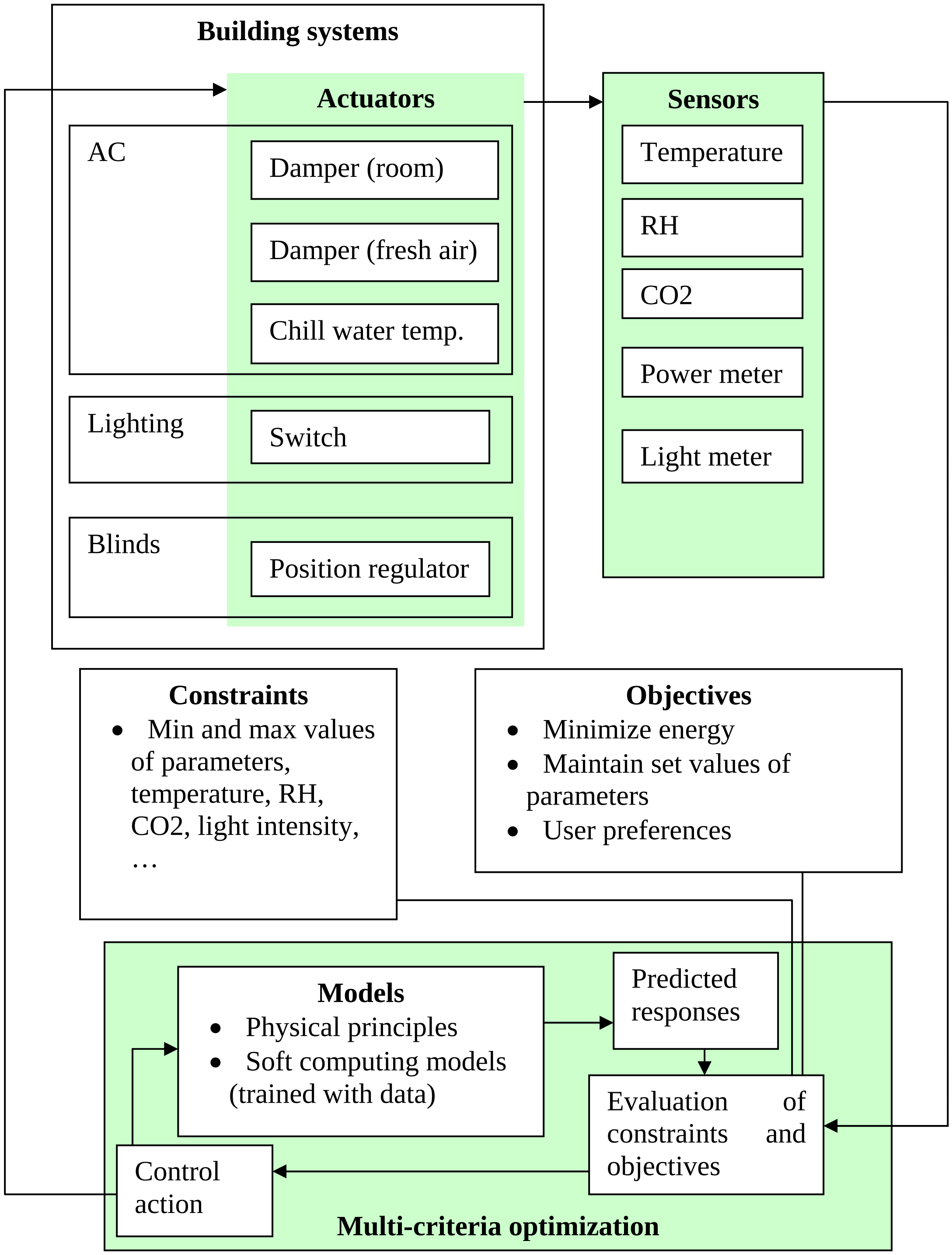


Figure 2.3 Schematic of integrated control

2.3 New algorithms for multi-criteria search

Two new algorithms for multi-criteria search have been developed in this research. The first one generates a Pareto front using a version of random search called MOPD (Raphael 2006). This is described in Section 2.3.1. The quality of the Pareto front generated by the new algorithm is evaluated using two examples from outside the domain of control. Its application to control is described in later chapters. The generated Pareto front can be used by engineers for visual inspection and evaluation of possible trade offs between conflicting objectives. This algorithm cannot be used in real time control because the control task requires the identification of a single solution completely autonomously, that is, without human intervention. Pareto front consists of a number of solution points which represent different trade-offs among objectives. It does not provide any direct information about which is the best solution to be chosen.

The second algorithm that has been developed in this research aims to select a single solution that can be used in real time control. This algorithm requires the ordering of objectives according to their importance and specifying the sensitivity of the values of the objective functions. This is described in Section 2.3.2.

2.3.1 A new algorithm for generating Pareto Front: MOPD

An algorithm for generating the Pareto front should aim to satisfy three conditions.
a. Many points should be generated on the Pareto front.
b. The points should be uniformly distributed.
c. The points should be close to the theoretical Pareto front.

If the Pareto front is a continuous surface, the number of points is determined by the available computational resources such as memory and time. In order to ensure that the points are uniformly distributed, it may be sufficient to remove points that lie too close to one another. The third condition is the most difficult one to achieve. In general, there is no way to ensure that the generated points lie close to the theoretical front.

The new algorithm aims to satisfy the above conditions by dividing a multi-objective optimization problem into a series of single objective optimization problems that are easier to solve. This is done by selecting one objective at a time and treating remaining objectives as constraints.

Let O1, O2, … , On be the objectives to be minimized. Let P1, P2, … , Pn be the global minima for each objective (Figure 2). Let mini and maxi be the minimum and maximum values of the i-th objective among the points P1, P2, etc. Suppose we want to generate p points on the Pareto front. The region between mini and maxi can be divided into p-1 intervals to form a grid. The grid points are obtained by solving the following constrained optimization problem:

Minimise Oj (Equation 2.1)

such that

Oi < mini + k*(maxi – mini)/(p-1), for each i not equal to j, and 0 < k < p

The above optimization problem should be solved for each j and the resulting set of points should be filtered according to the Pareto optimality criterion. If equation 1 is solved using global optimization, Pareto optimal points that are reasonably uniformly distributed along the grid lines are obtained. However, this is computationally inefficient when there are many objectives. Some heuristics are used to reduce the computational complexity. First of all, it is recognized that usually there is continuity along the Pareto front. This means that it might be possible to generate a neighboring point on the front by starting a local search from an already generated point. Secondly, there is no need to obtain precise values for maxi and mini. Estimates of the maximum and minimum values of the objective functions might be obtained by local search and these can be updated during the process of generating the front. Since the objective functions are minimized repeatedly, there is a good probability that the global minima are obtained in the process. Thirdly, it is not feasible to obtain points at every grid point when the number of objectives is more than 2. Therefore, we only aim to generate points which are distant from the current point by one grid step, without insisting on generating points in every direction. Thus the constraints in Equation 2.1 are modified as follows:

Minimise Oj (Equation 2.2)

such that

Oi < pci + dyi, for all i not equal to j, where dyi is the current grid width for the i-th objective and pci is the value of the i-th objective for the current point (starting point for local optimisation).

It can be seen from Equation 2.2 that we are aiming to reduce the value of j-th objective in the current point, allowing the values of all other objectives to increase by a maximum value (which is equal to the grid width). By the definition of Pareto optimality, this procedure results in the generation of a new Pareto optimal point. **Thus the procedure has a firm mathematical foundation.**

With the above simplifications, the following algorithm is used to generate the Pareto front.

Step 1: Create a list *plist* to store Pareto optimal solutions.

Step 2: Perform *n* pseudo-global optimizations (using a number of local search from random starting points), where *n* is the number of objectives. In each optimization only one objective is considered; remaining objectives are ignored. Find the minimum and maximum values of each objective from among all the optimization runs. Let *min[i]* and *max[i]* denote the minimum and maximum values of the *i-th* objective. Every point generated during the optimizations are filtered using the Pareto optimality criterion and added to the list of Pareto optimal solutions

Step 3: Loop 2.1: Repeat for each point pc in *plist.*

Loop 2.2: Repeat for each objective, *Oi:*

Calculate dy[i], the width of the grid for the i-th objective using the equation (max[i] – min[i])/p, where p is a predefined parameter that controls the spacing of points. Perform local search starting from the point pc to solve Equation 2.2. Pareto list, min[i] and max[i] are updated.

End of Loop 2.2

End of Loop 2.1

Step 4: If sufficient number of Pareto optimal points are obtained, terminate. Otherwise repeat from Step 2.

Loops 2.1 and 2.2 are terminated prematurely if the number of evaluations of the objective functions exceeds a user-specified value.

2.3.1.1 Evaluation of the algorithm

Two examples are used to demonstrate that the algorithm is capable of generating the correct Pareto front. The first one is a standard benchmark problem that involves the minimization of mathematical functions. The second involves the design of a shaft under combined bending and tension.

Bi-objective set covering problem

This problem involves the minimization of two objectives z1 and z2 subject to a set of linear constraints, when the variables are discrete and take only two values 0 or 1. The problem is defined as follows:

Minimize the Vector Z = {z1, z2}

Subject to

[A] {X} >= 1

where

Z = [C] {X}

C is a matrix of dimension 2 x n containing integer values

{X} is a column vector containing the optimization variables xi

[A] is a matrix of dimension m x n

m is the number of constraints, n is the number of variables

Several standard instances of the problem are available on the WWW1. The result of one instance, namely, 2scp41A is presented here. This instance contains 40 constraints and 200 variables. Comparison between the Pareto front obtained by MOPD and an algorithm called Pareto Memetic Algorithm (PMA) is shown in Figure 2.4. The points obtained by MOPD has lower values for both objectives. Therefore, the curve lies below that obtained by PMA. There is also a reasonable distribution of points in the front obtained by MOPD, even though, PMA scores better in this respect.

[1] http:// http://www.univ-valenciennes.fr/ROAD/MCDM/ListMOSCP.html

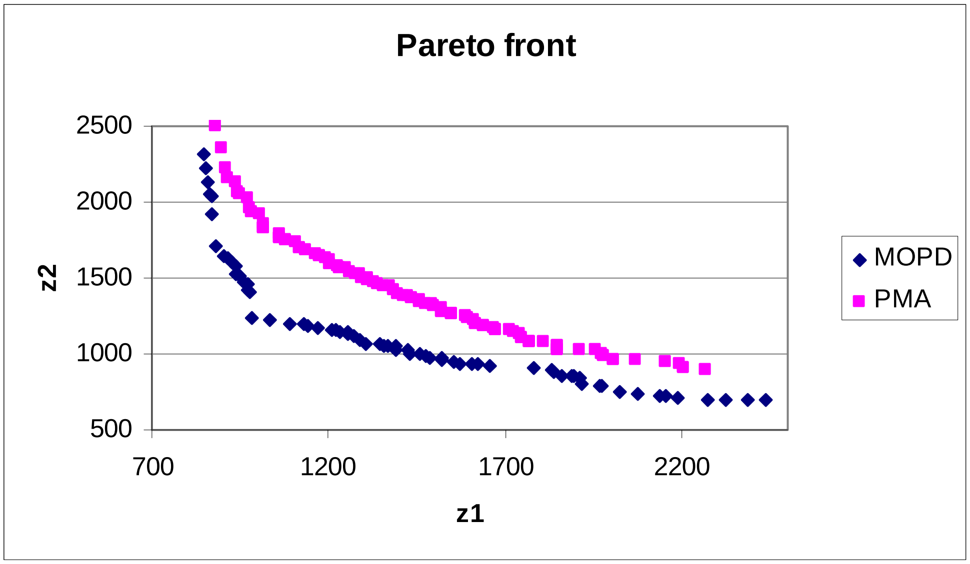


Figure 2.4 Comparison between MOPD and PMA (reported in Jaszkiewicz, 2004).

Design of a hollow shaft

This example involves the design of a hollow steel shaft (Figure 2.5). Design parameters are d and t. The objectives are to minimize 1) the weight of the structure and 2) its vertical deflection. These objectives generally conflict each other. Designs have to satisfy constraints related to maximum stress as well as local buckling.

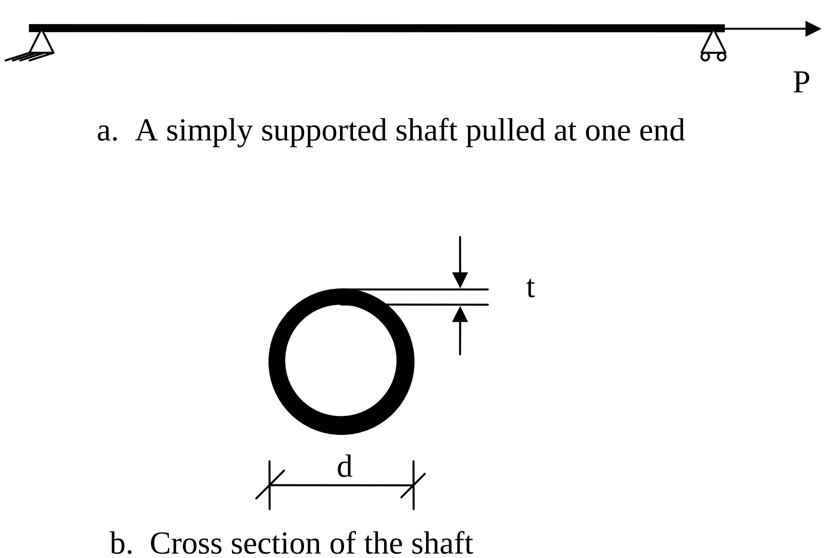


Figure 2.5. Design of a shaft under tension and bending due to self weight.

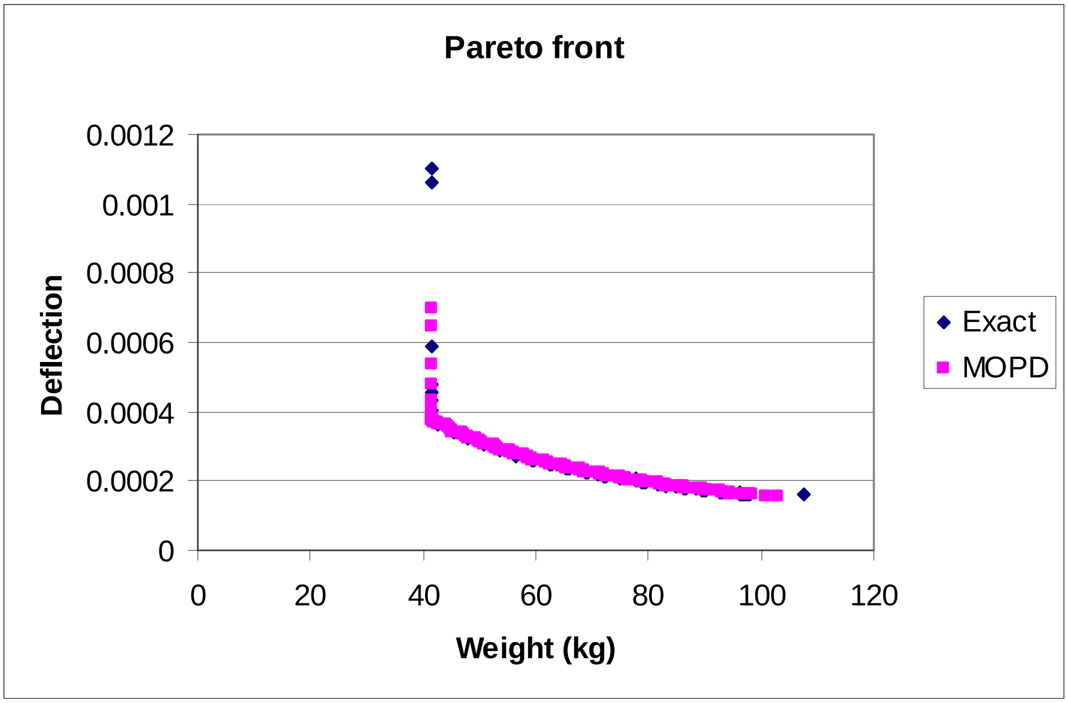


Figure 2.6. Pareto front obtained by MOPD for the shaft design problem overlaps the exact curve obtained through exhaustive search.

The diameter of the shaft d, was varied between 5 mm and 1000 mm. The thickness t was varied between 1 mm and 100 mm. Both variables were treated as discrete and were allowed to vary at intervals of 1 mm. Since there were only two design variables, the exact Pareto front could be calculated by evaluating all possible combinations of values. This is compared with the solution obtained by MOPD. Both curves were identical. It shows that MOPD is able to generate the exact front with good distribution of points.

The values of d and t of Pareto optimal solutions are shown in Figure 2.7. The curve is discontinuous. For example, there are no solutions between d=386 and d=518. This is confirmed by exhaustive search. Within this interval, both the weight and deflection could be simultaneously reduced by decreasing the diameter and increasing the thickness or increasing the diameter and decreasing the thickness. Above d=593, local buckling governs, so the thickness has to be increased along with the depth. This causes the weight to increase while the deflection decreases. The set of solutions to the right side of the kink belong to this case. By examining Figures 2.6 and 2.7 together, good values of d and t might be chosen that represent reasonable trade-offs between deflection and weight.

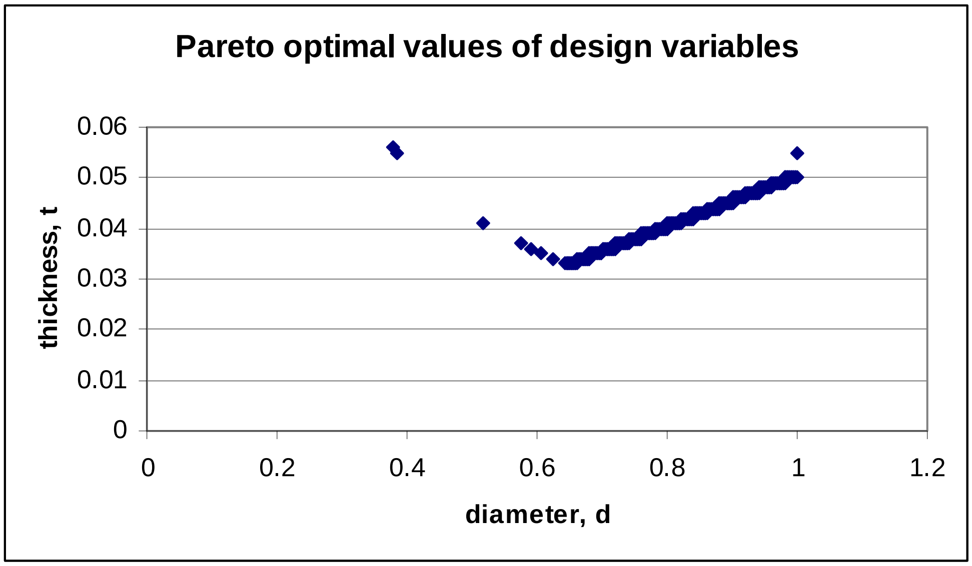


Figure 2.7. Shaft design: Pareto optimal solutions.

2.3.2 Selecting a single solution based on multi-criteria optimality

A Pareto front is useful for engineers for multi-criteria decision making. However, it is not useful for automatic control since this requires the identification of a single best solution. For this purpose, a new algorithm has been developed. The algorithm is based on the following principles:

- Some objectives are more important than the others (objectives have to be ordered according to their importance)
- Small differences in the value of an objective are not important (sensitivity of objectives should be taken into account)

To illustrate the first principle, user preference should be given higher priority compared to energy savings. However, it may not be possible to quantify the relative importance in the form of weight factors. Therefore, the algorithm should not attempt to combine multiple objectives into a single objective using weight factors.

To illustrate the second principle, consider this situation. Two solutions differ by less than 1% in terms of energy consumption. Since the difference is not significant, both solutions may be considered to be equivalent with respect to this objective. From among these two solutions, a better solution might be chosen by using other objectives.

The newly developed algorithm, called, Relaxed-Restricted Pareto Optimal Selection (RR-PARETO) is described next.

2.3.2.1 Relaxed-Restricted Pareto Optimal Selection

In this algorithm, the objectives are ordered (ranked) by the users according to their importance. The user also inputs the sensitivity of each objective in terms of a percentage. All the points lying within the specified percentage are considered to be equivalent with respect to that objective. The algorithm uses the definition of an optimal class as follows: Within a set of points, all the points that are equivalent to the best point according to a specific objective belong to the optimal class of that objective.

A global search algorithm such as PGSL (Raphael and Smith, 2003b) is used to generate solutions that move towards the global optimum for the primary objective function (highest ranking objective). During the search, each point is evaluated using all the objective functions. A set of points known as Relaxed-Restricted Pareto set (RR-Pareto Set) is maintained by the algorithm. A point is added to this set if it satisfies the following conditions

a. The point should satisfy the Pareto optimality criterion (that is, it should be non-dominated)

b. The point should have a primary objective function value which is within the specified sensitivity with respect to the best solution known so far (the point belongs to the optimal class of the primary objective)

c. At least one secondary objective function value is within the specified sensitivity with respect to the best solution according to this objective (the point belongs to the optimal class of the at least one secondary objective)

At the end of the search, the RR-Pareto set is sequentially filtered according to each objective in the order of their importance. During each filtering, all the points that do not belong to the optimal class of the objective are removed. This process is terminated when there is a single point remaining in the set. This point is chosen as the best solution. If there are multiple points left after the filtering, the best point according to the primary objective is chosen.

In order to understand how the algorithm works, an example is shown in Figure 2.8. The figure shows the energy required for lighting and air conditioning for various positions of a window blind. The first curve shows the energy required for lighting alone. The second curve shows the total energy for lighting and air conditioning (taking only the power consumed by the chiller for illustration). The window blind position varies from 0% to 100% and is shown on the x-axis. The blind position 0% represents completely closed and the position 100% represents completely open. The energy required for lighting and cooling are obtained from simulations and are described in more detail in Section 2.5. As the blind opens, the energy required for lighting decreases, while the energy required for cooling increases. The Pareto front for this problem is shown in Figure 2.8. Here the objectives are lighting energy and the total energy. Even though, normally we would like to minimize total energy, in some situations we might want to minimize lighting energy because this increases natural daylight in the room and is preferred by users.

In this example, the blind positions from 0% to 95% are on the Pareto front. With pure Pareto filtering, it is not possible to select a single best blind position.

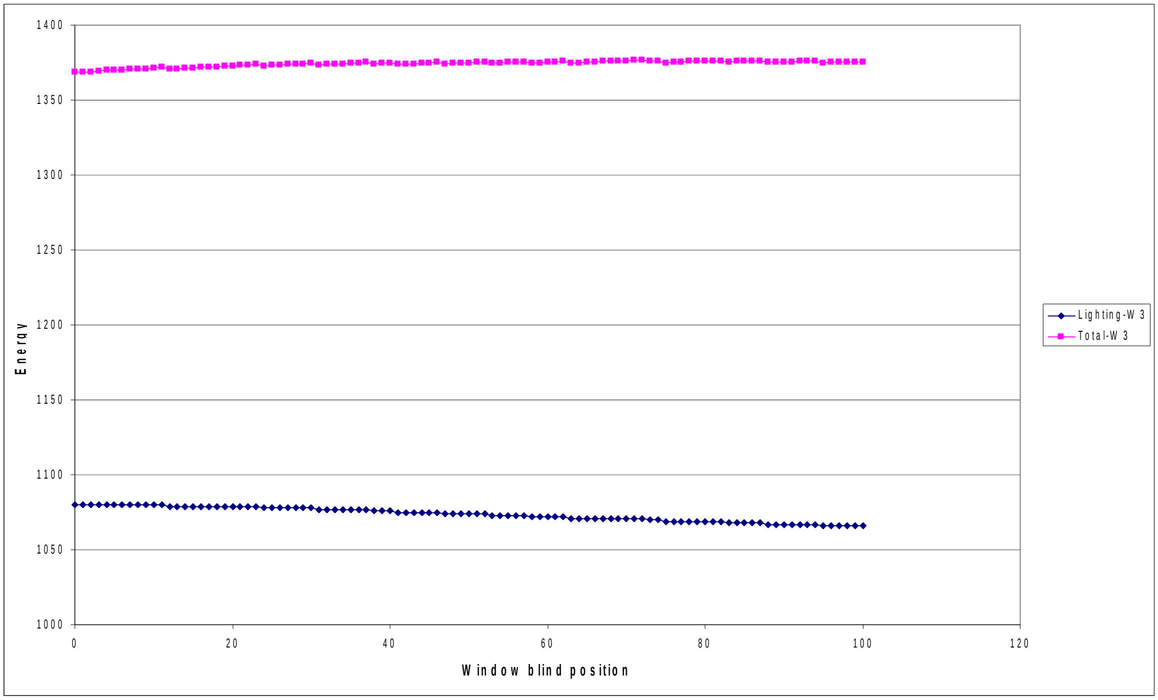


Figure 2.8. Lighting and cooling energy for various positions of a window blind

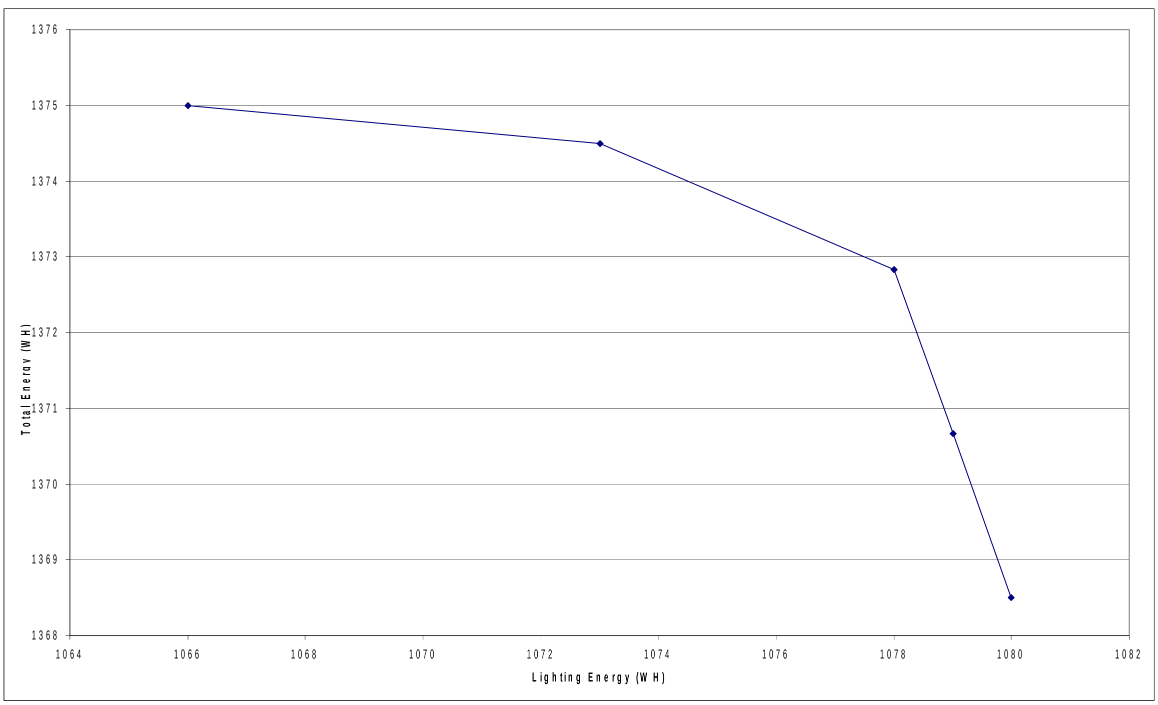


Figure 2.9. Pareto front for a window blind control problem

The points that belong to the Pareto front are shown in Table 2.1.

Blind position	Lighting Energy (WH)	Total Energy (WH)
0	1080	1368.50
12	1079	1370.67
24	1078	1372.83
53	1073	1374.50
95	1066	1375.00

Table 2.1 Points on the Pareto front

In order to illustrate the new algorithm, two scenarios are considered. In the first scenario, the sensitivity of both the objectives is specified as 2% and the total energy has higher priority over lighting energy. The best point in Table 2.1 according to the objective of total energy has a value of 1368.5 WH. All the points lie within 2% of this value, and therefore, all the points shown above are initially added to the RR-Pareto Set. The best point according to the objective of lighting energy has a value of 1066 WH. All the points shown in the figure have a lighting energy within 2% of this value. Therefore, at the end of filtering all the points still remain in the RR-Pareto set. In this case, the best point according to the primary objective is chosen. This results in the blinds getting completely closed because the second objective of lighting energy is not able to discriminate between the solution points.

In the second scenario, the sensitivity of both objectives is specified as 1%. In this case, the chosen solution will be different even though all the points have a total energy within one percent of the best point. Only two points, that is, blind positions 53% and 95% are in the optimal class with respect to the lighting energy objective. Therefore, during the filtering process all the other points are removed from the RR-Pareto set. The best point with respect to the primary objective from the resulting RR-Pareto set is the blind position 53%.

The selection process can be interpreted as follows. All the points are equivalent with respect to the objective of minimizing total energy since all the solutions lie within the sensitivity limit. However, some of the points are not good with respect to the second objective and therefore, these are removed from the set. The best point with respect to the primary objective is selected from the remaining set. This point represents a good trade off between the two conflicting objectives.

2.3.2.2 Evaluation of RR-Pareto algorithm

The example given in the previous section provides empirical evidence that the new algorithm is able to select good solutions that make reasonable trade-offs among conflicting objectives. Users are able to control the selection of the optimal point by specifying sensitivities of objectives. More detailed evaluation of the algorithm is provided in later chapters.

2.4 Predictive control

The objective function that is used in global optimization for identifying the best control action computes the difference between predicted and desired responses. The control action can be selected correctly only if the responses can be predicted accurately. This is not critical in the case of closed loop feedback control since the control action is selected by trial and error. That is, if the action does not produce the anticipated effect, the action is reversed in the next step. However, the accuracy of prediction is quite important in global optimization. Otherwise, it loses its advantages over conventional techniques and takes significantly longer to converge.

Most building systems exhibit complex behavior and it is difficult to develop simple models that are accurate enough. The most reliable method of prediction is using simulation models. There are well tested and validated simulation software that can predict thermal load and lighting levels. These software could be used as black boxes and executed as external programs after developing a complete simulation model consisting of material properties, geometric properties and other data. Results from the simulation software are usually available as text files which can be read by the optimization routine for computing the difference between desired and predicted responses. Routines for interfacing with simulation software that have been developed in this project are described in Section 2.5.

The main drawback of executing simulation software as external programs is that they consume much CPU time. A single lighting simulation can take any where from a couple of seconds to a few hours depending on the complexity of the model. If the simulation software takes too long, it cannot be used for real time control. In this case, we need empirical models that are trained offline in order to produce quick results in real time. Machine learning and statistical techniques such as regression are commonly used for this task. Case Based Reasoning (CBR) is used in this project. This is described in more detail in Section 2.6.

2.5 Simulation Module: NUS-ILES

The prototype applications that have been developed in this project involve lighting simulation and thermal simulation. Radiance software (Ward and Shakespeare, 1995) is used for lighting simulation and EnergyPlus (Crawley et al., 2001) is used for thermal simulation. A simulation module called NUS-ILES (Integrated Lighting and Energy Simulation) has been developed in this project for interfacing between these two programs.

Radiance (Ward and Shakespeare, 1995) is a lighting simulation software that employs backward ray-tracing. This software is being used as the underlying simulation engine in many popular design software and has been extensively tested and validated. For examples, see Mardaljevic (1995); Freewan et al. (2008); Greenup and Edmonds (2004); Ochoa and Capeluto (2006); and Tsangrassoulis et al. (1996). These studies have generally concluded that Radiance simulation results are found to be in good agreement with physical models.

The following are the data required for running lighting simulation:

- Sky conditions
- Geometric model of the building including the configuration of shading devices
- Material properties such as visible transmittance of glass, reflectivity of surfaces, etc.

With the above data, the software is capable of computing the illumination level at any desired point.

The most important data required for predicting the daylight in the room interior is the position of the sun and the sky brightness. Even though solar tracking devices are available, they are expensive and not recommended here. Instead, the position of the sun is obtained from weather data files which are found in sources such as the US department of Energy website2. These data files provide the solar azimuth and altitude for any time and day of the year for a given geographical location.

EnergyPlus is an energy analysis and thermal load simulation program. Based on a simplified 3D model of a building and associated mechanical systems, EnergyPlus calculates the heating and cooling loads necessary to maintain thermal control set point conditions. This is one of the most extensively tested and validated energy simulation programs (Crawley et al., 2005) and is the official simulation program of the US Department of Energy. It was developed at the University of Illinois and the University of California through the Lawrence Berkeley National Laboratory.

Since Radiance and EnergyPlus are software that have been developed independently, their input data formats are different and it is not possible to use the same building model for both simulations. The new interface module NUS-ILES was developed to address this problem. This module uses a shared building model and generates the data files for Radiance and EnergyPlus using the same model. The model is defined in XML (Extensible Markup Language) which is the emerging standard in the representation of information. An XML document is a text file in which tags are defined to separate data into logical parts. The XML model of an existing office building in Singapore is shown in Figure 2.10a-e. It can be seen that data required for lighting simulation such as reflectivity of surfaces as well as data required for energy simulation such as thermal conductivity of materials are stored in the model.

---

[2] http://www.eere.energy.gov/buildings/energyplus/cfm/weather_data.cfm

Representation of the general characteristics and dimensions of a building is shown in Figure 2.10a. The walls of the building are grouped into four parts, west façade, east façade, north façade and south façade. These need not be perfectly aligned in the principal directions. It could be inclined with respect to these orientations in which case, the angle is specified in the model. A façade consists of a group of walls as shown in Figures 2.10a and 2.10b. A wall might contain multiple windows and overhangs. The representation of roofs and ground are shown in Figure 2.10c. The interior space is divided into a number of zones as shown in Figures 2.10c. A zone might have several internal partitions as shown in Figure 2.10d. The representation of a grid of lamps in a zone is shown in Figure 2.10e. The grid on which lighting levels are computed is shown in Figure 2.10e. Detailed description of the attributes and the format is given in Appendix A. In summary, the XML representation is a comprehensive building model for performing lighting and energy simulations.

```xml
<?xml version="1.0" encoding="UTF-8"?>


<building>

<!--
coordinates x,y,z store the position of components relative to the containing
object. dimensions length, width and height of components are in the local
(natural) coordinates of the component

        the global coordinates are oriented like this:
                x-axis in the west - > east direction
                y-axis in the south - >north direction
                z-axis along the height of the building

the local coordinates for a facade are like this
        x-axis along the length of the facade increasing to the RHS when viewing
from exterior. y-axis perpendicular to the length of the facade increasing towards
the interior
-->

  <!-- Total dimensions of the building -->
  <!-- distance in the x direction  between the walls on the east and west -->
  <length>             18.0    </length>

  <!-- distance in the y direction between the walls on the north and south -->
  <width>              36      </width>

  <height>             10.8    </height>

  <westfacade>
    <wall id="1">

      <description>  Wall on the west facade between grids 3 and 5 </description>

      <!-- Origin shift - position of the wall relative to the west edge -->
      <x>                       0       </x>
      <y>                       0       </y>
      <z>                       0       </z>

      <length>                  36.0    </length>
      <height>                  10.8    </height>
      <thickness>               0.15    </thickness>

      <U>                       0.23    </U>
      <!-- Thermal conductivity  -->
      <reflectivity>            0.6     </reflectivity>
```

Figure 2.10a. XML representation of an office building

```
    <window id="W03-02-100-1">
      <description>  West facade window between grids 3 and 4</description>

      <!-- window position relative to the wall  -->
      <x>            1.825   </x>
      <!-- distance from the corner of the wall  -->
      <z>            0.3     </z>
      <length>       6.8     </length>
      <height>       2.6     </height>

      <U>            2.2     </U>
      <!-- Thermal conductivity  -->
      <SC>           0.39    </SC>
      <!-- Shading coefficient   -->
      <VT>           0.5     </VT>
      <!-- Visible Transmittance   -->
    </window>

     … omitted for brevity

    <overhang>

      <description>
             shading below light shelf between grids 3 and 4
     </description>

      <x>            1.825   </x>
      <z>            1.585   </z>
      <length>       6.6     </length>
      <width>                1.05    </width>
      <thickness>    0.07    </thickness>
      <reflectivity> 0.8     </reflectivity>
    </overhang>

     … omitted for brevity

  </wall>
</westfacade>

<eastfacade>
  <wall id="2">

    <description>  Wall on the east facade between grids 3 and 5 </description>

     … omitted for brevity

</wall>

</eastfacade>
```

Figure 2.10b (continued from 2.10a). XML representation of an office building

```xml
<roof>
  <x>                       -1      </x>
  <y>                       0       </y>
  <z>                       10.8    </z>

  <length>           22.0    </length>
  <width>            39      </width>
  <thickness>               0.15    </thickness>

  <U>                       0.23    </U>
  <!-- Thermal conductivity  -->
  <reflectivity>            0.1     </reflectivity>

</roof>


<ground>
  <reflectivity>            0.01    </reflectivity>
</ground>


<zone id="Z21a">

  <description> floor 2 office between grids 3 to 5 </description>

  <!--  Origin of the zone         -->
  <x>                       0       </x>
  <y>                       18      </y>
  <z>                       0       </z>

  <!--  Zone dimensions     -->

  <!-- distance in the x direction  -->
  <length>                  18      </length>

  <!-- distance in the y direction  -->
  <width>                   18      </width>

  <!-- total height  -->
  <height>                  3.6     </height>


  <ceiling>
    <z>                     2.7     </z>
    <reflectivity>          0.8     </reflectivity>
    <thickness>             0.1     </thickness>
  </ceiling>
```

Figure 2.10c (continued from 2.10b). XML representation of an office building

```xml
<partitions>

   <partition>
     <description>  Partitions for Rooms along grid 3  </description>


     <x>                     0       </x>
     <y>                     15      </y>
     <z>                     0       </z>

     <dx>                    18      </dx>
     <dy>                    0       </dy>
     <height>                2.7     </height>
     <thickness>             0.1     </thickness>

     <reflectivity>          0.8     </reflectivity>
   </partition>

   <partition>
     <description>  Partitions for Rooms along grid 3 on the west </description>

     <x>                     4.5     </x>
     <y>                     15      </y>
     <z>                     0       </z>

     <dx>                    0       </dx>
     <dy>                    3       </dy>
     <height>                2.7     </height>
     <thickness>             0.1     </thickness>

     <reflectivity>          0.8     </reflectivity>
   </partition>

    … omitted for brevity

</partitions>


 <floor>
   <reflectivity>            0.01    </reflectivity>
 </floor>
```

Figure 2.10d (continued from 2.10c). XML representation of an office building

```xml
  <lamps>

    <grid>

      <description>  Lamps in the offices near grid 5      </description>

      <x>            1.5     </x>
      <y>            1.5     </y>
      <dx>           2       </dx>
      <dy>           3       </dy>
      <nx>           4       </nx>
      <ny>           1       </ny>

      <lamp>
        <watt>               56      </watt>

        <!-- Ballast and other losses -->
        <loss>    4    </loss>
        <lux>                300     </lux>
        <!-- min lux provided by the lamp within its radius of influence  -->
        <target>     300     </target>
        <!-- target lighting level  -->
        <influence>  3       </influence>
        <!-- radius of influence of the lamp  -->
        <dimmable>   yes     </dimmable>
      </lamp>
    </grid>

      … omitted for brevity

  </lamps>

 <grid id="1">
    <x>              0.5     </x>
    <y>              0.5     </y>
    <!-- Height of workplane  -->
    <z>              0.8     </z>
    <dx>             1       </dx>
    <dy>             1       </dy>
    <nx>             18      </nx>
    <ny>             18      </ny>
  </grid>


</zone>
<!-- end of zone z21a  -->
```

Figure 2.10e (continued from 2.10d). XML representation of an office building

2.5.1 Thermal simulation

EnergyPlus does not have a user friendly Graphical User Interface (GUI). However, this is not critical since the simulations are done automatically without any user interaction. The input is read from a text file and the output is produced in the form of tab separated text files. NUS-ILES automatically prepares the input files needed to run EnergyPlus. A sample text file generated by NUS-ILES is shown in Figure 2.11.

Even though a GUI is not needed for automatic control applications, it is useful for model validation during the design stage. EnergyPlus generates autocad drawing files for verifying models visually. A sample drawing file is shown in Figure 2.12.

Any required output can be produced from EnergyPlus by configuring the reports. Since hourly simulations are performed in the prototype applications developed in this project, NUS-ILES has configured EnergyPlus to produce report in the format shown in Table 2.2.

Date/Time	Environment:Outdoor Dry Bulb [C](Hourly)	ZONE ONE:Zone/Sys Sensible Cooling Energy[J](Hourly)
12/15 07:00:00	25.00	637960.224
12/15 08:00:00	25.00	814470.465
12/15 09:00:00	25.00	1673793.56
12/15 10:00:00	25.00	2374380.62
12/15 11:00:00	26.56	3805098.62
12/15 12:00:00	25.31	3911285.34
12/15 13:00:00	24.63	2518328.43
12/15 14:00:00	25.31	2026692.34
12/15 15:00:00	25.81	2883253.92
12/15 16:00:00	26.63	3214843.21
12/15 17:00:00	27.81	3430621.1
12/15 18:00:00	28.74	3896838.06
12/15 19:00:00	27.75	2920221.17

Table 2.2 Sample output produced by EnergyPlus

The output which is a tab separated text file, is read by NUS-ILES for further processing to be used in optimization routines.

```
!  Generated by NUS-ILES Copyright (c) Benny Raphael (2008)
!  The template for generating this file was taken from an IDF generated input file

RunPeriod,
  12,                    !- Begin Month
  15,                    !- Begin Day Of Month
  12,                    !- End Month
  15,                    !- End Day Of Month
  UseWeatherFile,             !- Day Of Week For Start Day
  Yes,                   !- Use WeatherFile Holidays/Special Days
  Yes,                   !- Use WeatherFile DaylightSavingPeriod
  No,                    !- Apply Weekend Holiday Rule
  Yes,                   !- Use WeatherFile Rain Indicators
  Yes,                   !- Use WeatherFile Snow Indicators
  1;                     !- Number of times runperiod to be done

MATERIAL:WINDOWGLASS,
  CLEAR 3MM,                  !- Name
  SpectralAverage,            !- Optical Data Type
  ,                      !- Name of Window Glass Spectral Data Set
  0.003,                 !- Thickness {m}
  0.837,                 !- Solar Transmittance at Normal Incidence
  0.075,                 !- Solar Reflectance at Normal Incidence: Front Side
  0.075,                 !- Solar Reflectance at Normal Incidence: Back Side
  0.898,                 !- Visible Transmittance at Normal Incidence
  0.081,                 !- Visible Reflectance at Normal Incidence: Front Side
  0.081,                 !- Visible Reflectance at Normal Incidence: Back Side
  0.0,                   !- IR Transmittance at Normal Incidence
  0.84,                  !- IR Hemispherical Emissivity: Front Side
  0.84,                  !- IR Hemispherical Emissivity: Back Side
  0.9;                   !- Conductivity {W/m-K}

CONSTRUCTION,
  SingleGlazing6,             !- Name
  CLEAR 6MM;                  !- Outside Layer

Surface:HeatTransfer:Sub,
  W1,                    !- User Supplied Surface Name
  WINDOW,                     !- Surface Type
  SingleGlazing6,             !- Construction Name of the Surface
  ZONE SURFACE WEST,               !- Base Surface Name
  ,                      !- OutsideFaceEnvironment Object
  autocalculate,              !- View Factor to Ground
  ,                      !- Name of shading control
  ,                      !- WindowFrameAndDivider Name
  1,                     !- Multiplier
  4,                     !- Number of Surface Vertex Groups -- Number of (X,Y,Z) groups in this surface
  0,                     !- X,Y,Z  1 {m}
   5.5,                  !- X,Y,Z  1 {m}
  0.601,                 !- X,Y,Z  1 {m}
  0,                     !- X,Y,Z  2 {m}
   5.5,                  !- X,Y,Z  2 {m}
   0.6,                  !- X,Y,Z  2 {m}
  0,                     !- X,Y,Z  3 {m}
   0.5,                  !- X,Y,Z  3 {m}
   0.6,                  !- X,Y,Z  3 {m}
  0,                     !- X,Y,Z  4 {m}
   0.5,                  !- X,Y,Z  4 {m}
  0.601;                 !- X,Y,Z  4 {m}
```

Figure 2.11 A sample EnergyPlus input file generated by NUS-ILES

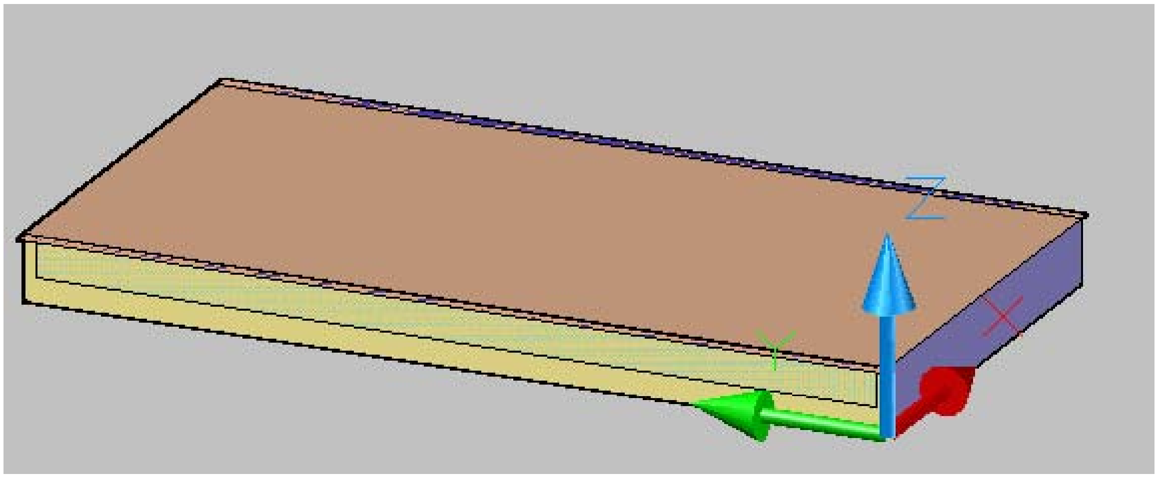


Figure 2.12 A sample drawing file generated by EnergyPlus for verifying the input model

2.5.2 Lighting simulation

The primary objective of running lighting simulation in this project is to compute the energy required for artificial lighting. Computation of lighting energy consumption using the simulation software Radiance involves the following steps:

Step 1: Model creation: Building geometry, material properties and weather data are used to create a simulation model which is compiled into an octree representation. The position of the sun (depends on the hour of the day that is simulated) and the sky parameters are part of this model.

Step 2: The illuminance at various points in the interior of the building is evaluated using the octree model. This is typically done along a grid of spacing 1 meter or less. In Radiance, a tool called rtrace reads points one by one and outputs the illuminance values sequentially.

Step 3: The average illuminance value that is measured by a light sensor is predicted by taking the average of all the points within the area influenced by the sensor.

Step 4: Using the predicted average illuminance value, the level to which the lights can be dimmed are calculated. The power consumption of the lamp is calculated by assuming linear relationship between dim level and the power.

Step 5: The total power consumption of all the lamps at the particular hour is calculated.

Step 6: Steps 1 to 5 are repeated for different hours of the day (simulation hours) in order to obtain hourly power consumption. The total energy consumption is calculated by summing up the energy for each simulation hour.

The above steps are summarized in the form of a flow chart in Figure 2.13. There are two nested loops. The outer loop repeats for all the hours of simulation, typically from 8 am to 7 pm when day light is available. The inner loop repeats for each grid point where the illuminance level is to be computed.

NUS-ILES prepares the data files needed for running Radiance simulation automatically using the models defined in XML format. Radiance input consists of text files in a particular format that is defined by the software. The text files contain the definition of the 3D model as well as material properties such as reflectivity and visible transmittance.

Similar to EnergyPlus, Radiance produces output mainly in the form of text files. This is read by NUS-ILES for further processing to be used in optimization. In addition to the text file output, image files can also be generated by Radiance. These are useful for model verification. A sample rendering produced by Radiance is shown in Figure 2.14. The results of the simulation of this building are shown in Figure 2.15 for illustration.

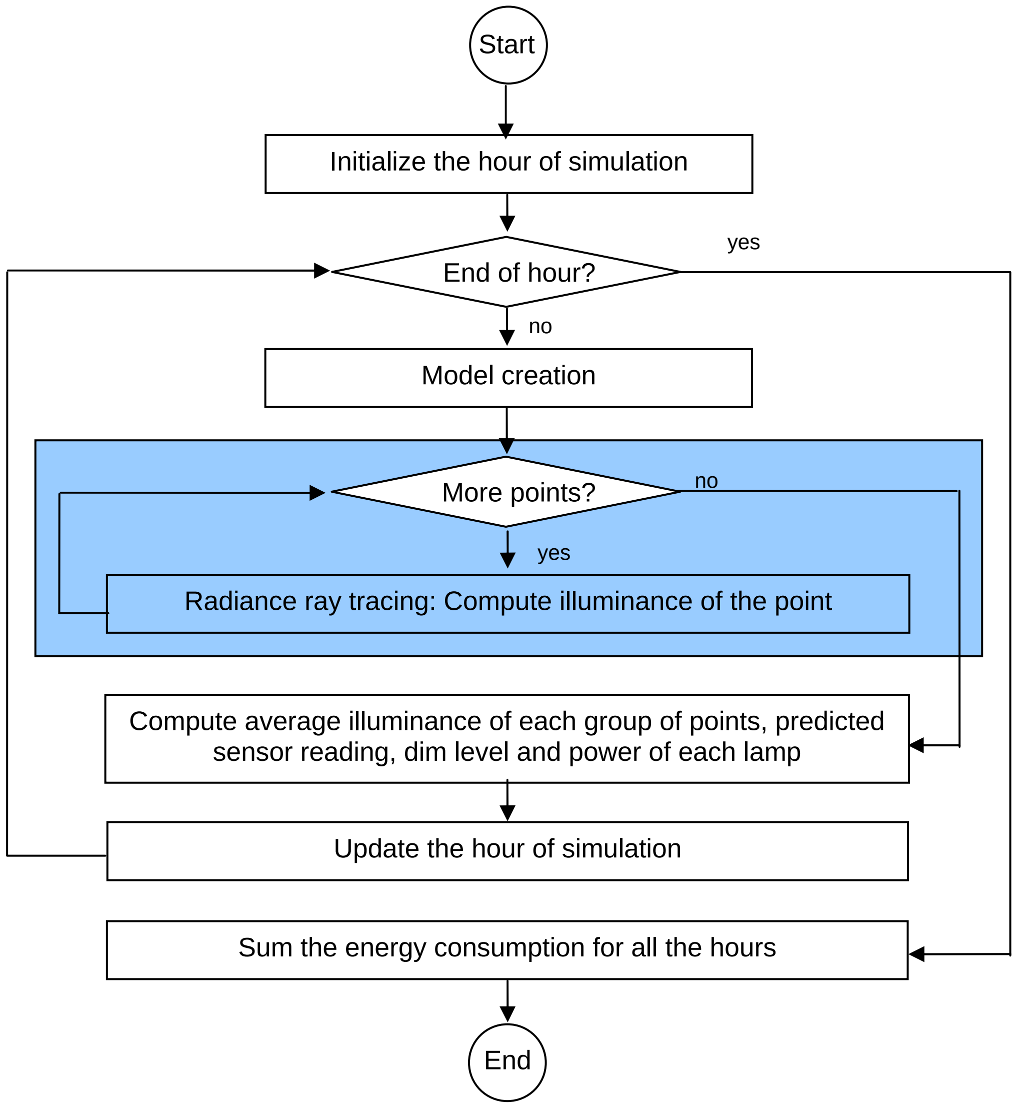


Figure 2.13. Algorithm for computing the energy consumption of artificial lights.

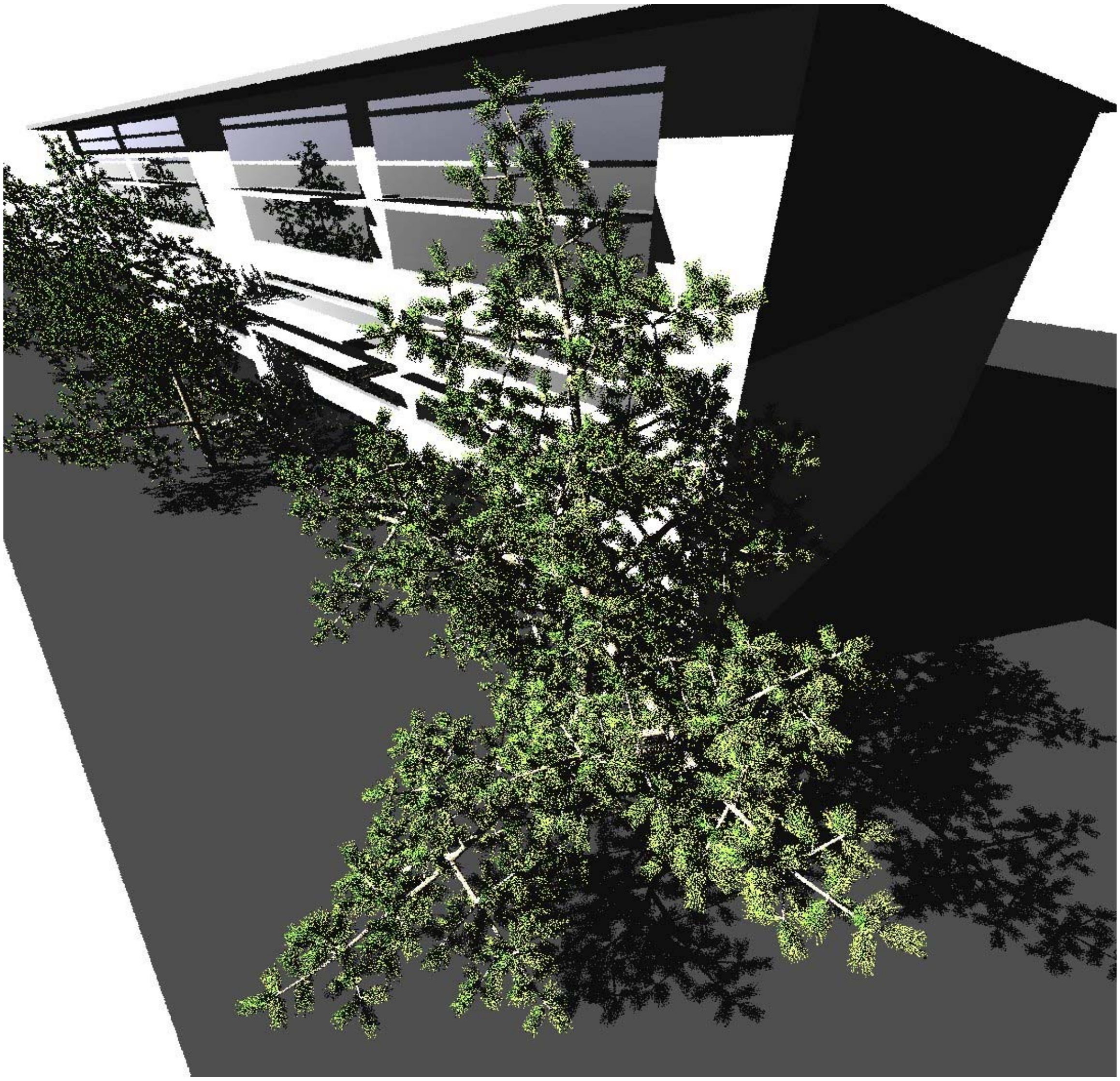


Figure 2.14. Rendering of the model of an office Building by the Radiance simulation software

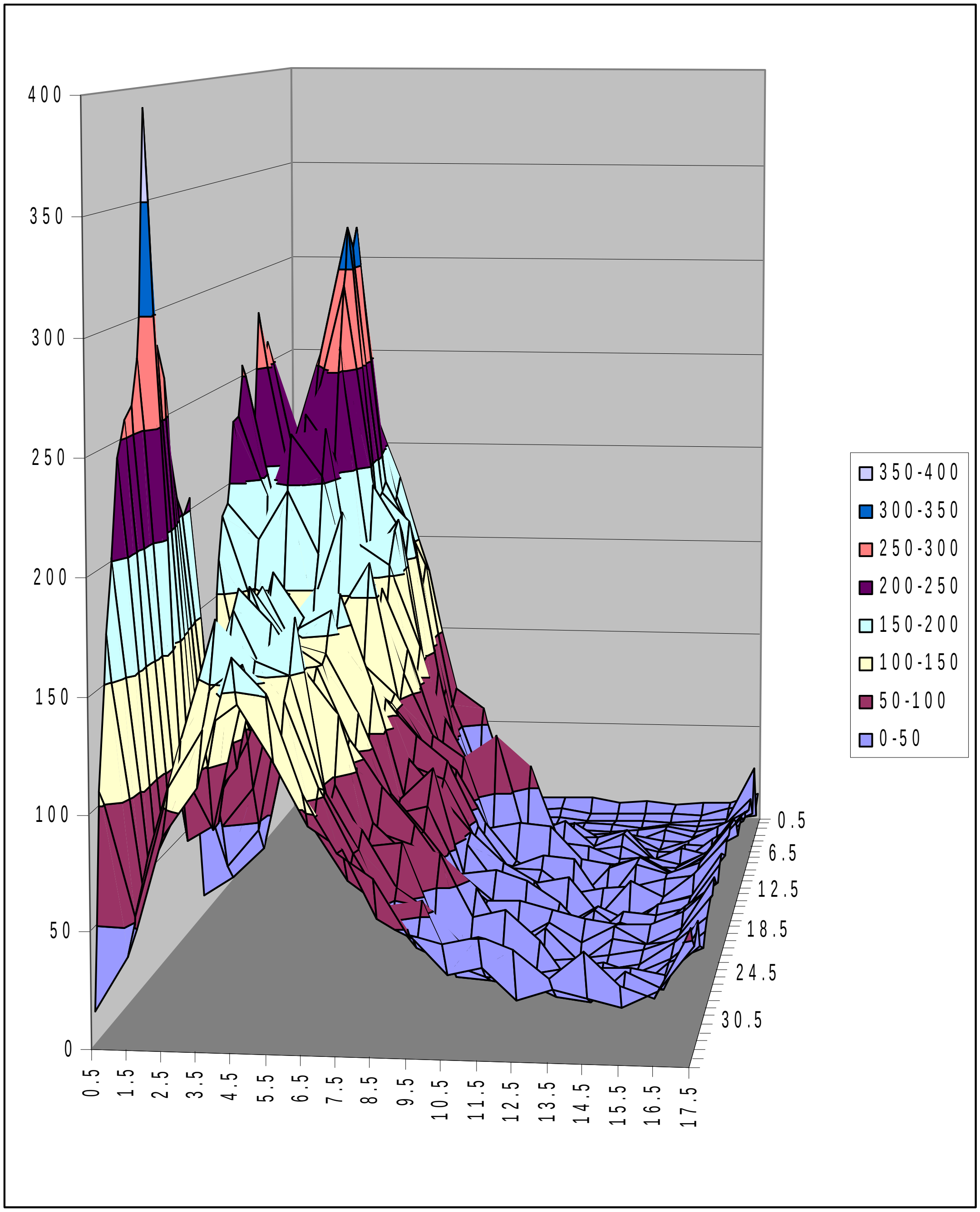


Figure 2.15. Surface plot of brightness (lux) at every grid point at 14:00 hours in an office building. Points on the west side have higher illuminance because the sun is on the west side at this time.

2.6 Learning strategy

The objective of machine learning in this project is to increase the speed of simulations and to develop empirical models when models based on physical principles are not available. The outcome of machine learning is a predictive model which can be used to compute output variables for the given values of input variables. The predictive model is developed using training data which could be based on simulation results or actual measurements using sensors.

The machine learning program developed in this project is called NUS-CBRFIT and is based on Case Based Reasoning (Raphael and Smith, 2003a). Case-based reasoning involves finding solutions to new tasks through reusing good solutions to old tasks. Three important tasks in CBR are representation, retrieval and adaptation. Representation refers to storing existing solutions in the case base in an appropriate form. Retrieval refers to the selection of the most relevant cases to the new problem. Adaptation involves modifying existing solutions to create a new solution.

In NUS-CBRFIT, cases are numerical solutions obtained through simulations or measurements. A case consists of a set of input variables and a set of output variables. For example, in the prediction of cooling load for a set of blind positions, the input variables are the blind positions and the output variable is the cooling load. Past solutions obtained through simulations are stored in the case base.

For a new problem in which input variables are known and the output variable needs to be predicted, similar cases are retrieved from the case base. NUS-CBRFIT selects similar cases according to the following algorithm:

Loop 1: Repeat for each input variable

> Step 1: Select the case *A* in which the value of the input variable is lesser than or equal to the value in the new problem and the difference between the two values is the minimum. If the case *A* is not already present in the list of selected cases, add case *A* to the selection, otherwise select another case with the next minimum distance. If no case is found satisfying the condition, ignore this step.
>
> Step 2: Select the case *B* in which the value of the input variable is greater than the value in the new problem and the difference between the two values is the minimum. If the case *B* is not already present in the list of selected cases, add case *B* to the selection, otherwise select another case with the next minimum distance. If no case is found satisfying the condition, ignore this step.

End of Loop1

Case adaptation in NUS-CBRFIT is done using all the selected cases through the statistical technique of multiple linear regression. This is done using the following algorithm.

Step 1: Compute pair-wise correlations between the output variable and each input variable using the data from selected cases.

Step 2: If all the variables have the same value (within a specified precision) for any variable, that variable is removed from the predictive model.

Step 3: If the number of cases selected in Loop 1 is less than the number of input variables, remove the input variables with the least correlation with the output variable from the predictive model, until the number of cases is at least one more than the number of input variables.

Step 4: Perform multiple linear regression using the selected input variables and the data from the selected cases. This gives a linear relationship between the input and output variables. This relationship is used to predict the value of the output variable.

There are several novel features in this learning algorithm, which include,

- The combination of CBR and statistical techniques
- The use of multiple cases for adaptation
- The selection of cases in such a way that the interpolation is most reliable

In order to test the efficiency and accuracy of the algorithm a set of 575 points were generated for a blind control task involving two window blinds. The points were generated by using random blind positions and performing thermal load simulations. The generated points are added to the casebase. Test data is created by generating blind positions along a grid consisting of 20 values for each blind. The cooling load was predicted using NUS-CBRFIT for each combination of blind position. The predicted cooling load is plotted in Figure 2.16. The x-axis is the first blind position. The y-axis is the second blind position. The z-axis is the predicted cooling load. It can be seen that the input-output relationship is not linear. The average prediction error was less than 2% and the computation time was a fraction of a second for each prediction. This shows that the machine learning module is both accurate and efficient and suitable for online control.

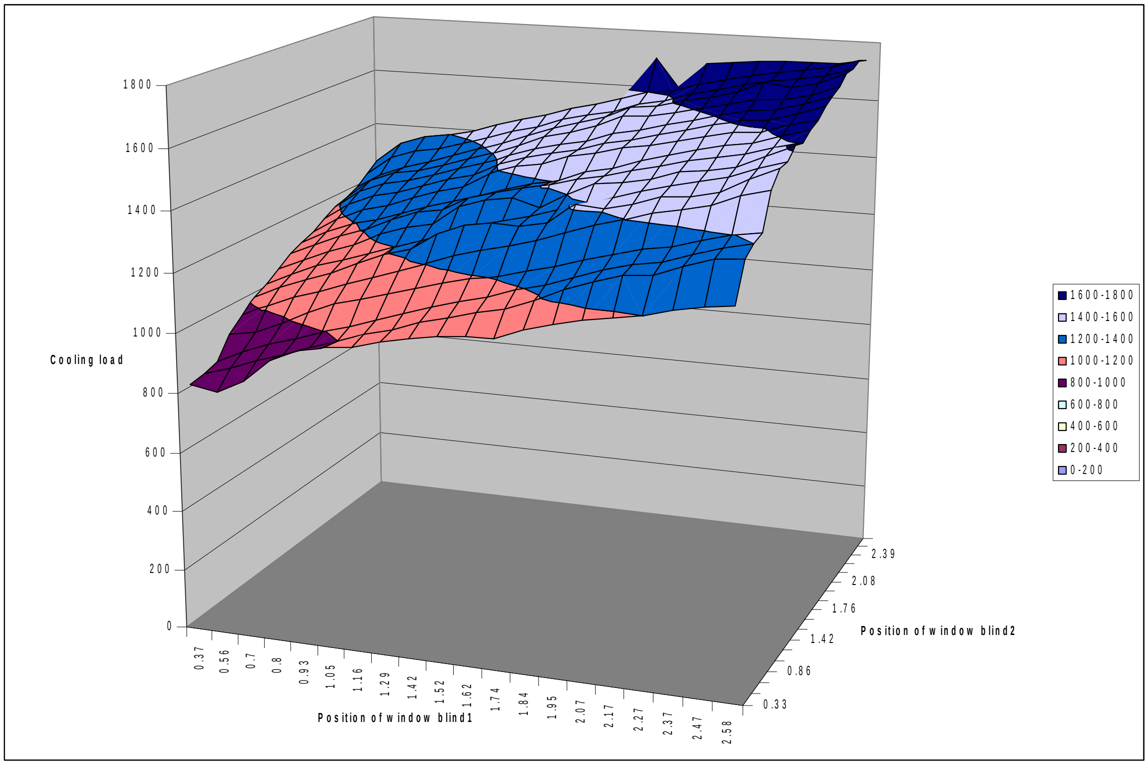


Figure 2.16. Cooling load predicted by the learning algorithm NUS-CBRFIT for different positions of two window blinds. The average prediction error is less than 2%.

2.7 Conclusions

A framework for the integrated control of IEQ was developed in this project. The framework uses multi-objective global optimization for selecting the best control action. Important components in the framework are Pareto optimization, predictions using simulation models and predictions using machine learning models. These components have been implemented in the form of generic software program modules. The modules have been tested using a number of test cases which prove their performance in terms of efficiency and accuracy.

References

Adam, B. and Smith (2007). I.F.C. Tensegrity Active Control: a Multi-Objective Approach, J Computing in Civil Engineering, Volume 21, Issue 1, pp. 3-10.

Alcala, R.; Casillas, J.; Cordon, O.; Gonzalez, A. & Herrera, F. (2005). A genetic rule weighting and selection process for fuzzy control of heating, ventilating and air conditioning systems. Engineering Applications of Artificial Intelligence, Vol. 18, No. 3, April 2005, pp. 279-296

Chen T., (2001). Real-time predictive supervisory operation of building thermal systems with thermal mass, Energy and Buildings, 33(2), pp. 141-150.

Chow T., Zhang G., Lin Z., and Song C., (2002). Global optimization of absorption chiller system by genetic algorithm and neural network, Energy and Buildings, 34(1), pp. 103-109.

Christina Diakaki, Evangelos Grigoroudis, Dionyssia Kolokotsa, (2008). Towards a multi-objective optimization approach for improving energy efficiency in buildings, Energy and Buildings 40 1747–1754.

Crawley, D.B., Lawrie, L.K., Winkelmann, F.C., Buhl, W.F., Huang, Y.J., Pedersen, C.O., Strand, R.K., Glazer, J., (2001). EnergyPlus: Creating a new-generation building energy simulation program, Energy and Buildings, 33 (4), pp. 319-331.

Crawley, D.B., Hand, J.W., Kummert, M., Griffith, B.T. (2005). Contrasting the capabilities of building energy performance simulation programs Ninth International IBPSA Conference, pp. 231-238.

Czyzak P. and A. Jaszkiewicz. (1998). “Pareto Simulated Annealing – A Metaheuristic Technique for Multiple-Objective Combinatorial Optimisation.” Journal of Multi-Criteria Decision Analysis 7, 34–47.

Deb K. and Tiwari S., (2005). Omni-optimizer: A Procedure for Single and Multi-objective Optimization, Coello et al. (Eds.): EMO 2005, LNCS 3410, pp. 47–61.

Grierson D. E., and Khajehpour S., (2002). A method for conceptual design applied to office buildings, ASCE Journal of Computing in Civil Engineering, Vol. 16, No. 2, April 2002, pp 83-103.

Guillemin A., and Morel N., (2002). Experimental results of a self-adaptive integrated control system in buildings: a pilot study, Solar Energy, 72(5), pp. 397-403.

Haralampidis Y., Papadimitriou C. and Pavlidou M., (2005). Multi-objective framework for structural model identification, Earthquake Engng Struct. Dyn,. 34, pp. 665–685.

Jaszkiewicz, A., (2004). “A comparative study of multiple-objective metaheuristics on the Bi-Objective set covering problem and Pareto Memetic Algorithm.” Annals of Operational Research 131, pp.135-158.

Kim I. Y., Weck O. L., (2006). Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation, Struct Multidisc Optim, 31, pp. 105–116.

Kolokotsa, D. (2006). Artificial Intelligence in Buildings: A Review on the Application of Fuzzy Logic, Advances in Building Energy Research, Vol.1, pp. 29-54, James and James Publisher.

Kummert M., Andre P., and Nicolas J., (2000). Optimal heating control in a passive solar commercial building, In proceedings of the eurosun 2000 Congress, Copenhagen, Denmark.

Mahdavi, A., Unzeitig, U., (2005). Occupancy implications of spatial, indoor-environmental, and organizational features of office spaces, Building and Environment 40 (1), pp. 113-123.

Morel N., Bauer M., El-Khoury M., and Krauss J., (2001). Neurobat, a predictive and adaptive heating control system using artificial neural networks, International journal of solar energy, 21, pp. 161-201.

Raphael B. and Smith I.F.C., (2003a). Fundamentals of computer aided engineering, John Wiley and sons, Ltd, (306 pages).

Raphael B. and Smith I.F.C., (2003b). A direct stochastic algorithm for global search, J of Applied Mathematics and Computation, Vol 146, No 2-3, pp 729-758.

Raphael, B., (2006). "A new approach to multi-objective optimization". IT solutions for the design and managment of infrastructure construction projects , ed. B. Kumar and P.R. Swarup CIDC. (ICITE/ITCSED, 15 - 17 Nov 2006, New Delhi 2006.

Robert-Nicoud Y., Raphael B. and Smith I.F.C., (2005). System identification through model composition and stochastic search, ASCE journal of Computing in Civil Engineering, (JCCE), Vol 19, No 3, pp. 239-247.

Suter, G., Brunner, K., Mahdavi, (2007). A., Building model reconstruction based on sensed object location information, Automation in Construction 16 (1), pp. 2-12.

Tan K.K., Wang Q.G., Hang C.C., (1999). Advances in PID Control. London, UK: Springer-Verlag,.

Ulungu, E.L., J. Teghem, Ph. Fortemps, and D. Tuyttens. (1999). “MOSA Method: A Tool for Solving Multiobjective Combinatorial Optimization Problems.” Journal of Multi-Criteria Decision Analysis 8, 221–236.

Ziva Kristl, Mitja Kosir, Mateja Trobec Lah, Ales Krainer, (2008). Fuzzy control system for thermal and visual comfort in building, Renewable Energy 33 694–702.

3 Implementation of an Integrated Control System

Summary

This chapter describes the hardware and software details related to the implementation of an integrated control system framework. Hardware components needed for the control such as sensors, actuators and controllers are discussed briefly. The rationale for the choice of communication protocol is also discussed. Finally, programming details related to the communication using the BACNET protocol are presented.

3.1 Introduction

A framework for the integrated control of IEQ that was developed in this project is described in Chapter 2. This chapter describes the implementation details related to the hardware and software.

In the present control system framework, IEQ parameters are evaluated through data from sensors. This data is used by the multi-objective optimization module to select the best control actions that reasonably satisfy objectives and constraints. The selected actions are applied by sending commands to appropriate actuators. Responses to these actions are recorded by sensors and are input into the optimization module to complete the feedback loop.

The important pieces of hardware required to implement this framework are

- Controllers
- Operator workstations
- Sensors
- Actuators
- Communication network

In addition to the hardware, communication protocols are critical for the successful implementation of the system. This chapter discusses the hardware and software options available for the implementation of the control system.

3.2 Control systems hardware and protocol

Several communication protocols have been developed by control system vendors over the years. These include Profibus, Modbus, Lonworks, etc. In the recent years, there is a shift in trend away from proprietary systems towards open systems and international standards. Two important candidates for the implementation of the control system that were evaluated in this project are EIB and BACNET. These are described below.

3.2.1 EIB (KNX)

European Installation Bus (EIB) is a building automation system founded in the late eighties with the support of a number of major European manufacturers such as Siemens, Gira, Jung, Merten, and ABB. It is being widely used in Europe and is increasingly being used in other parts of the world. Even though it is currently used mainly for controlling home appliances such as lighting, water heaters and security cameras, the flexibility offered by this technology makes it possible to control any type of electrical device. The EIB system consist a bus control cable installed in addition to and parallel to the power supply cable. EIB devices such as switches and sensors are connected to the bus. The bus is the common communication medium. Readings from sensors can be obtained through a computer connected to EIB. Control commands to actuators can also be sent from the computer through the bus. This makes it possible to develop customised control applications that operate on these devices.

EIB is now renamed as KNX and is an international standard and is administered by Konnex Association3. The standard has been officially published and is available from the website of Konnex Association. However, software tools have to be purchased from certified vendors. There are a limited number of companies dealing with these in the local context and getting support from them was found to be difficult. Preliminary evaluation of the software tools also indicated that they are more expensive than the BACNET system.

3.2.2 BACNET

BACNET stands for Building Automation and Control Network4. It is a data communication protocol developed by the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). It is now an ISO global standard. BACnet is a true non-proprietary open protocol communication standard. There are many companies dealing with BACNET systems and the availability of products and services is good.

One important consideration in the selection of the protocol for the control system was the availability of function libraries for communication with controllers from standard desktop computers. BACNET communication software is available for reasonable price and it is possible to develop custom programs using this software. This is available in the form of a dynamic link library (DLL) which can be called from commonly used programming languages such as VB.NET and C. This makes it possible to develop custom software that can be executed stand-alone without having to interface with

[3] http://www.knx.org/knx-standard/knx-specifications/

[4] http://www.bacnet.org

proprietary software systems. Due to this reason, BACNET protocol was selected for the implementation of the prototype.

3.3 Control system architecture and components

A typical architecture of a BACNET control system is shown in Figure 3.1. BACNET controllers (Direct Digital Controllers) are connected to a local area network (LAN) through the use of BACNET routers. Operator workstations which are ordinary desktop or laptop computers on the LAN are able to communicate with the controllers using the standard TCP/IP protocol.

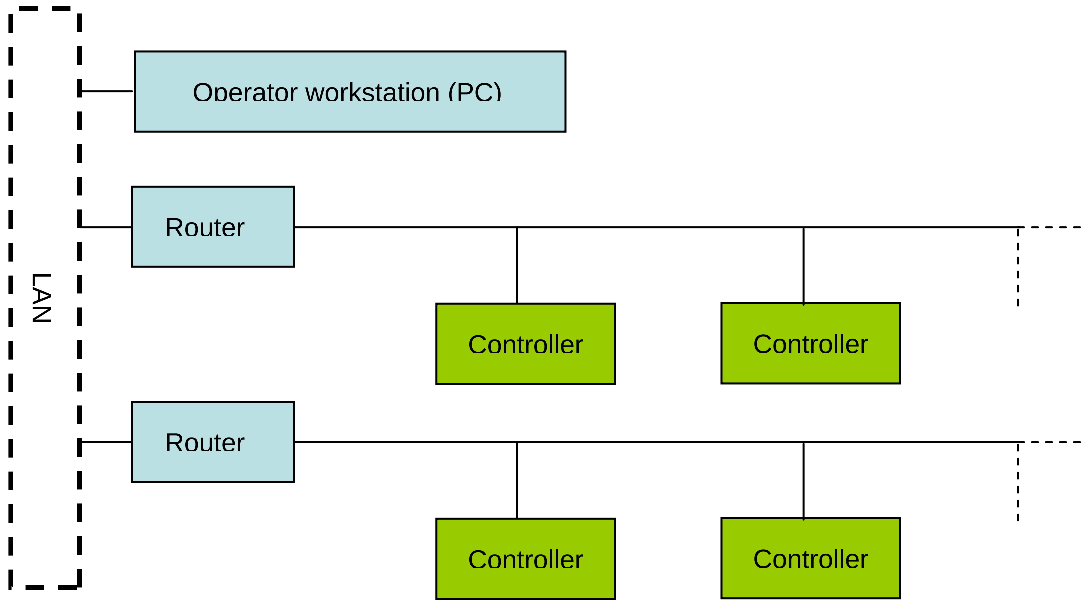


Figure 3.1 A simple BACNET control system architecture

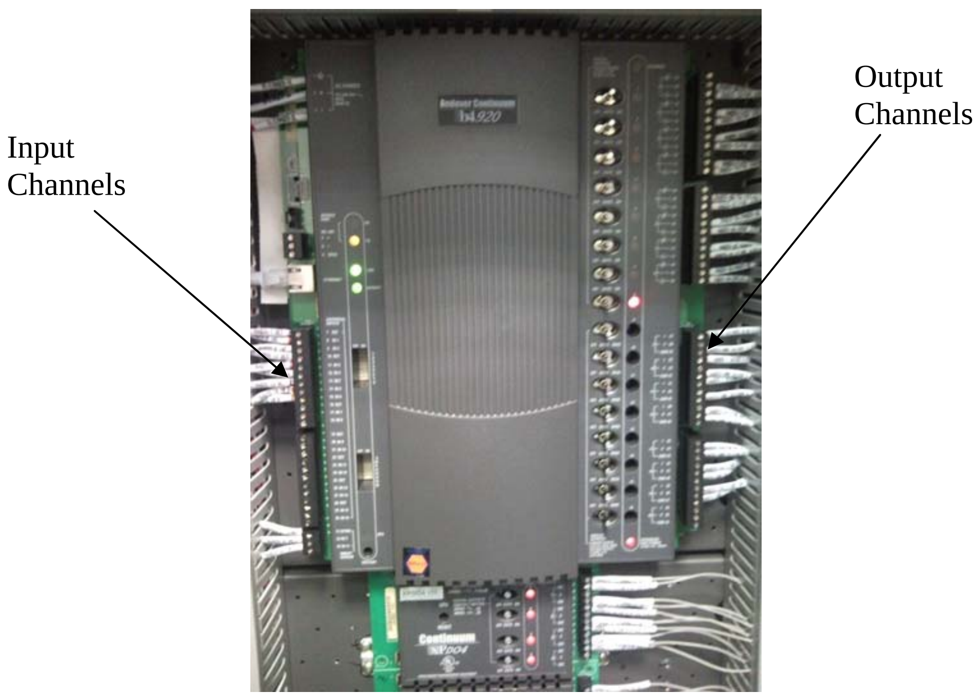


Figure 3.2 A BACNET Direct Digital Controller

A Direct Digital Controller (DDC) is a small computer with its own microprocessor, memory, etc. It has a set of input and output channels to which sensors and actuators are connected. Figure 3.2 shows the picture of a BACNET DDC. Input channels can be seen on the left side and the output channels on the right side. Sensors are connected to the input channels. Actuators are connected to the output channels.

Sensors produce either voltage or current output. This can be read by the controller and the sensor reading can be obtained from the operator workstation by sending a BACNET request to the controller. Different types of sensors are used in this project. These include

- Temperature sensors
- Motion sensors for detecting the presence of occupants
- Photo sensors for measuring the level of lighting
- Heat flux sensors for measuring the amount of radiant heat
- Air velocity sensors for measuring the air movement through ducts
- CO_2 sensors for measuring the level of CO_2 in ppm

Pictures of some of the sensors used in this project are shown in Figure 3.3.

Output channels are of two types, binary and analogue. Binary output can be used for operating relays which open or close a circuit. The picture of a damper actuator that takes a binary command input is shown in Figure 3.4. When a current is applied to the command input line of the actuator, the motor starts rotating opening or closing the valve of an air conditioning duct. The motor stops when the current is stopped. The controller can be commanded to apply or stop the current at any of its binary output channels.

Analogue output channels can be used to apply a varying voltage in a specific range such as 0-10 V DC. Analogue output can be used to operate actuators which operate in a continuous range. The amount of movement is proportional to the applied voltage. Figure 3.5 shows a damper actuator that takes command in the form of an analogue voltage. The degree of rotation of the damper valve is proportional to the applied voltage.

In addition to reading input from sensors and producing output to operate actuators, the DDC can also perform computations by running programs on its microprocessor. The specific model of BACNET controllers that have been used in this project permit writing programs in a language known as Plain English Programming Language. This is similar to Visual Basic. However, the program development environment is not as user friendly as Microsoft Visual Studio. The environment is not conducive for developing complex programs. The memory of the controller is also limited. Therefore, it was decided to execute the program on personal computers and use the controllers only for reading sensor data and operate actuators.

a) Air velocity sensor

b) CO_2 sensor

d) Temperature sensor

c) Light sensors

e) Heat flux sensor

f) Motion sensor

Figure 3.3 Different types of sensors used for control

Binary
Command
Input

Figure 3.4 A damper actuator which takes a binary command input

Figure 3.5 A damper actuator which takes an analogue command input

3.4 Communication routines

As mentioned earlier, communication with the controllers is achieved through the use of a DLL (Dynamic Link Library) that implements the BACNET protocol. This particular library was developed by an automation company which supplied the BACNET control system hardware5. The communication routines are encapsulated in a class called BACNet. An instance of this class is created using the following VB.NET code:

```
Dim bac As BACnet
bac = New BACnet()
```

The parameters required to connect to the controller are specified by setting the attributes of the bacnet object as follows:

```
bac.Net = networkID
bac.Dest = networkDestination
bac.Host = hostAddress
```

The primary communication routines needed are reading values from input channels and producing output on the output channels of the controller. These are achieved by calling the methods defined in the BACNet class. These are illustrated using the following code:

```
 ' Writing a value to an output channel
 'whose identifier is specified by objectid
bac.WriteProperty(objectid, PropertyPresentValue, value)

 ' Reading a value from an input channel
 ' whose identifier is specified by objectid
bac.ReadProperty(objectid, PropertyPresentValue, value)
```

It can be seen that basic communication can be achieved using a fairly simple API (application program interface). The more difficult part is the configuration of the controllers which involves setting up the network parameters, configuring the input and output channels, etc.

3.5 Control Logic

The control logic is coded in the programming language VB.NET. VB.NET is a modern object oriented programming language. It is a general purpose and powerful programming language which is being widely adopted for developing applications on the Microsoft Windows platform. It is very efficient and provides a rich user interface for fast prototyping. Therefore, VB.NET was chosen as the platform for the implementation of the control system.

[5] http://www.cna.com.sg/

All the algorithms described in Chapter 2 were coded in VB.NET. This includes the optimization and machine learning algorithms. The simulation programs Radiance and EnergyPlus were called as external programs from VB.NET. After the control actions are determined through optimization, these are communicated to the controllers using the routines described in Section 3.4.

4 Prototype implementation: Blind Control

Summary

This chapter describes the implementation of the multi-objective framework presented in Chapter 2. A software system for the control of window blinds has been developed. This software was tested using a laboratory prototype involving motorized window blinds. Solar radiation was simulated using an array of light bulbs which could be dimmed to different levels. The laboratory prototype demonstrates that the integrated control strategy is feasible. Tests show that the system performs according to expected patterns. Energy saving potential of the control strategy is evaluated using the case study of a hypothetical office building. Energy savings of up to 26% is possible for the selected case study.

4.1 Introduction

Introducing natural daylight into buildings is an essential part of improving energy efficiency. Apart from direct savings of electricity used by artificial lights, daylighting also reduces air conditioning energy due to the reduction in heat dissipated by electric lights. It is reported that substantial savings can be achieved through integrated daylighting because electric lighting accounts for 25–40% of a commercial building's energy requirements; Energy savings as high as 52% have been reported in some cases (Leslie 2003). However, overall energy savings could be achieved only through meticulous design which involves the evaluation of not only the lighting performance, but also other aspects such as solar heat gain and the HVAC system. For example, large windows permit more daylight into the building, but increases the energy required for air conditioning in tropical climates. Some form of optimization is needed to make the balance between daylighting and cooling load.

Traditionally, the balance between daylighting and cooling load is achieved through passive design using rules of thumb. Recently, the use of simulation software has become more wide spread. The availability of reliable simulation software (Section 2.5) permits the systematic evaluation of design options through approaches such as optimization and search (Raphael and Smith, 2003). Recently, a number of papers that discuss the optimization of building performance have been published, for example, Diakaki et al. (2008), Wright et al. (2002) etc. In particular, muti-criteria optimization has been recently shown to have much potential, even though, the idea was originally proposed more than a decade ago (Gero et al. 1983). While significant improvement in performance can be achieved through passive design optimization, the potential for energy savings through active control is even more. One application of active control which has attracted the attention of many researchers is the control of window blinds. Lee et al. (1998) studied the performance of a dynamic venetian blind designed to optimize daylight admission and solar heat gain rejection in real-time, and concluded that energy savings of up to 32%

could be achieved. Other studies involving the control of daylighting and shading devices include Kristl et al. (2008); Park et al. (2005) and Guillemin and Morel (2001).

Most control systems for window blinds reported in the literature operate using simple rules such as a limit on the maximum direct solar radiation coming through the window (typically 100 W/m2) or estimated glare discomfort. Where optimization is performed, only one optimization criterion is used at a time, for example, energy optimization or visual optimization as in Guillemin and Morel (2001). Such optimizations use simplified procedures instead of full-scale simulations. In Guillemin and Morel (2001), a parameter called illuminance ratio is used to correlate the indoor horizontal illuminance to the outside vertical illuminance. This parameter can be used to predict indoor daylight level at a given point once the daylight in the exterior of the building is measured. However, a number of light sensors are needed for calibrating the parameters for all the required points in the room in order to control the artificial lights in a large office building. Even though, sensors are becoming cheaper, the cost of sensors is still very high compared to that of conventional lighting.

This chapter presents a control strategy for window blinds that employs the multi-objective framework presented in Chaper 2. This strategy requires very few sensors. Lighting and energy simulations are used to predict daylight levels and cooling load instead of using real time measurements using sensors.

4.2 Control of window blinds as an optimization problem

The type of window blinds used in this study is external blinds as shown in Figure 4.1. However key aspects of the methodology are equally applicable for the control of other types such as glazing with internal blinds, venetian blinds etc. In the chosen type, the blind which is installed in the exterior of the window can be opened or closed to any level by operating a motor. Raising the blind causes more daylight to enter into the room through the window. However, this might also increase the solar heat transmission and cause glare when the sun is at certain positions. Setting the blind at an appropriate position could achieve a balance between lighting requirements and solar heat gain.

Computing the optimal blind position involves prediction of interior daylight levels for each position of the blind under given environmental conditions. This can be done by daylighting simulation, provided an accurate model of the building and the environment is created. Building model parameters include reflectivity of surfaces and visible transmittance of window glass. Environmental parameters include the position of the sun and sky conditions. The lux levels at any point in the interior or exterior of the room are obtained using simulation software. Using the lux levels it is possible to calculate the power required for artificial lighting in order to achieve a target illumination, for example, 500 lux.

The cooling load can be computed using software such as EnergyPlus. This can be used to obtain a rough estimate of the energy required for air conditioning by assuming a value for the coefficient of performance (COP) of the system. The COP can be computed in real time if all the required data are recorded by the system. If detailed data related to the energy consumption of the chillers and AHUs are obtained, a more accurate energy model might be used for predicting the total energy consumption.

The total energy required is obtained by summing up the energy for artificial lighting and air conditioning as well as other equipment loads. Theoretically, the optimal blind position can be calculated by minimizing the total energy. This can be done using stochastic search as shown in Figure 4.2. In the simplest form, this is a single-objective optimization problem in which the optimization variable is the blind position and the objective function is the total energy.

Even though, minimizing the total energy is attractive from the point of view of environmental sustainability, it might result in the window blind being closed most of the time which is not preferred by users. If user preference is also included in the optimization model it becomes a multi-objective optimization problem. The RR-Pareto algorithm presented in Section 2.3 can be used to find solutions to the multi-objective optimization problem. The two objectives used in the optimization model are:

1. Minimization of total energy
2. Maximization of day light

The second objective is activated if the user prefers to have the blinds open, but is willing to make compromises in order to achieve energy efficiency. This objective is equivalent to the minimization of lighting energy and is used in this form in the present study.

Any of the two objectives might be chosen as the primary objective by the user. The sensitivity of the objectives should also be defined by the user. For illustration 1% is used as the sensitivity of both objectives. Using these parameter settings, blind positions that achieve appropriate trade-offs between the two objectives are identified by the RR-Pareto algorithm.

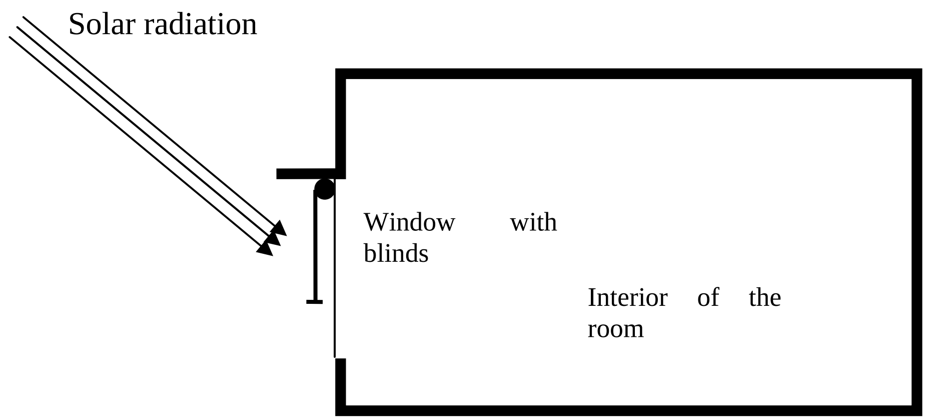


Figure 4.1. Schematic of an external window blind

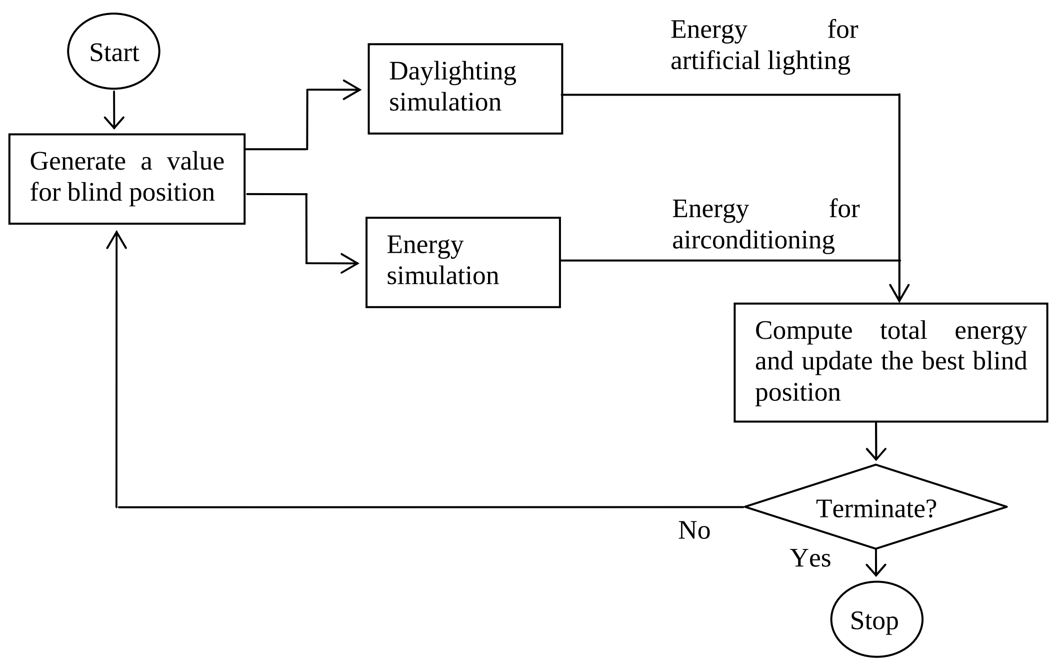


Figure 4.2. Stochastic search for computing optimal blind position

4.3 Improving the speed of optimization

The speed of optimization depends critically on the time taken to complete lighting and energy simulations. Ways to improve these are discussed in this section.

4.3.1 Lighting simulation

Lighting simulation takes considerable time and is not practical to do this on-line for real-time control. Determining the optimal control action might involve hundreds of lighting simulations and these might take anywhere from a few minutes to a few hours depending on the complexity of the model. A method of improving the speed of optimization is to reuse the results of simulations that have been executed apriori. That is, perform simulations in advance, store the results and reuse them during the optimization. However, this is not straight forward since there are many combinations of input conditions and it requires considerable amount of memory. This is evident from the discussion below.

Different combinations of blind positions and environmental parameters create different daylighting levels in the interior. A separate simulation has to be performed for each combination. Lux levels are typically calculated along a grid of spacing 0.5 m within the room in order to have reasonable accuracy in the computation of artificial lighting required. For an office space of 6m x 18m, this results in 432 grid points where lux values have to be stored for each simulation. When combined with different sun positions and sky conditions, it results in unacceptable overhead of computational resources for the control of a small office space.

In any case, the sky brightness varies depending on local conditions and should be measured on site. A simple method to measure sky brightness indirectly using outdoor light sensors is described below. Outdoor light sensors are typically installed on the roof pointing towards northern sky.

Two daylighting simulations are performed for each time of the day in order to obtain the predicted lux value at the point where the light sensor is installed; one for clear sky condition and the other for overcast condition. The values obtained from the sensor readings are compared with the predicted values in order to determine whether clear sky or overcast sky conditions should be used at any given time. After this, the sky brightness used in the selected simulation is proportionately scaled to produce a match between predicted and measured values. This procedure works well because indoor illuminance is found to be reasonably correlated with outdoor illuminance which is evident from the use of simple techniques such as daylight factors in many applications (Guillemin and Morel 2001)).

The scaling of the indoor lux levels according to outdoor brightness levels permit reducing the memory requirement considerably. The number of simulation results that need to be stored can be reduced further by noting that the results do not vary significantly between different days of the same month. The results for a typical day of the season might be used without much error. Therefore, different simulations are performed by varying the following:

- Day of the season (one for each season)
- Sky conditions (two values, clear sky, overcast sky)
- Time of the day (typically 13 hours starting from 7 am to 7 pm)
- Blind position (typically 100 values at intervals of 1%)

If there are multiple blinds in the room, one simulation is performed for each blind keeping all the other blinds completely closed. The lux levels for the combination of blind positions are generated by using the NUS-CBRFIT algorithm presented in Section 2.6.

During real time control, the nearest day of the season and the time of the day is located from the stored results using the system clock time. The sky conditions are determined using sensor measurements as explained above and the results are scaled according to the values obtained from light sensors. Thus the predicted distribution of lux values within the room is obtained for each combination of blind positions.

4.3.2 Energy simulation speed-up

Similar to the strategy for lighting simulation presented in Section 4.3.1, the speed of thermal load computation is increased by storing the simulation results a-priori. Since energy simulations do not take as much time as lighting simulation, a simulation is performed for each day of the year using weather data files, for various positions of the blinds. The predicted thermal loads for combinations of blind positions are obtained using the NUS-CBRFIT algorithm presented in Section 2.6.

Since the actual heat transmission is difficult to measure accurately using sensors, there are errors introduced through the use of simulation results. It should be mentioned that the corrections for actual outdoor temperature and humidity have not been incorporated in the present energy model. These are enhancements planned for the future.

The solar heat transmitted through the windows is corrected using a procedure similar to that followed in the case of lighting energy. The outdoor solar radiation is measured using a heat flux sensor. This is divided by the direct solar radiation (in W/m2) reported by the simulation software (EnergyPlus) for the hourly simulation for the day, in order to obtain the radiation correction factor. The solar radiation from the exterior window is estimated by subtracting the thermal load for a given blind position from the thermal load for the fully closed blind position. This value is multiplied by the solar correction factor in order to obtain a more accurate value for the solar heat transmitted by the window. This value is added to the thermal load for the fully closed position to obtain the predicted load for a given blind position.

4.4 Software implementation: NUS-IntBlind - An integrated control software for window blinds

A prototype implementation of a control system based on the framework described in Chapter 2 is presented in this section. The software has been developed such that the core modules containing control strategies are generic, but components such as chillers, motors, etc. have to be customized to individual implementations. The software is implemented in VB.NET programming language.

4.4.1 Control objectives

As described in Section 4.2, two control objectives have been implemented in the software prototype implementation. These are total power consumption and the lighting power consumption. Even though, lighting power forms part of the total power, it is considered as a separate objective since it represents the user preference for maximizing daylighting. It should be noted that minimizing lighting energy might result in higher total energy in certain cases.

The total power consumption consists of the following parts:

- Lighting
- Chiller system
    - Chiller
    - Chiller pumps
    - Condenser pumps
    - Cooling tower
- AHU fans
- Plug loads (electrical equipments)

The predicted total power consumption of each of the above components is computed as described in the following subsections. All the components except the plug loads depend on the positions of the blinds.

4.4.1.1 Lighting power

The lighting prediction module calculates the estimated power consumption of electrical lights for a specified combination of blind positions. This is described in detail in Sections 2.5.2 and 4.3.1.

4.4.1.2 Chiller system power

The blind position influences the chiller system power through the cooling load. In order to predict the change in power consumption of the chiller system, the following procedure is followed.

Step 1 (performed offline): The blind positions are discretized into 100 levels (at 1% increments). Energy simulation is performed for the zone affected by the window for each blind position as described in Section 4.3.2.

Step 2 (performed online): The difference between the solar heat gain of the current blind position and the proposed new position is calculated using stored simulation results for the time of the day (after applying the correction for the solar radiation).

Step 3: The predicted cooling load for each AHU for the new blind position is obtained by adding the increment in the solar heat gain obtained through Step 2.

Step 4: The total chiller load is calculated by summing up the cooling load of all the AHUs.

Step 5: Using the part-load performance curve of the chiller, the power consumption of the chiller is calculated.

Step 6: If the condenser and chiller pumps use variable speed drive, (VSD), the part-load of the chiller is used to calculate their power consumption. This is done by assuming a cubic relationship between part-load and power consumption. The power consumptions of cooling tower fans are also similarly calculated.

4.4.1.3 AHU fans

As a first approximation, the fan power is interpolated using two points, the current value and the maximum power. The fan power corresponding to the predicted cooling load for the AHU is calculated using these two points.

4.4.2 User interface

Figures 4.3a-b show parts of the user interface of the application. In the main screen (Figure 4.3a) the user can select the day and hour of the simulation results that can be used for control. By default, the simulation result for the nearest day is taken according to the system calendar. The closest simulation hour matching the current system time is also chosen.

The building model in the form of an XML file is read and the locations of controllable window blinds are determined. The names of these windows are populated in the list boxes in the main screen. The total energy consumption is computed and the optimal blind positions are determined. In this user interface, users are also provided the option to change the blind positions manually instead of applying the optimal blind positions.

The user interface shows variables that are important to the user such as the solar gain, cooling load, lighting power and total energy for the hour. The user is able to explore the effects of changing the blind position and compare the energy consumption for various subsystems. By changing parameters in a setup file, these options can be explored without actually applying the actions by sending the commands to the controller.

The system data screen can be invoked by clicking on a button in the main screen. In the system data screen, user can input parameters such as chiller part load efficiencies, power of pumps, ventilation rate, etc. There is an option to read real time load data, if the system is interfaced with a building management system.

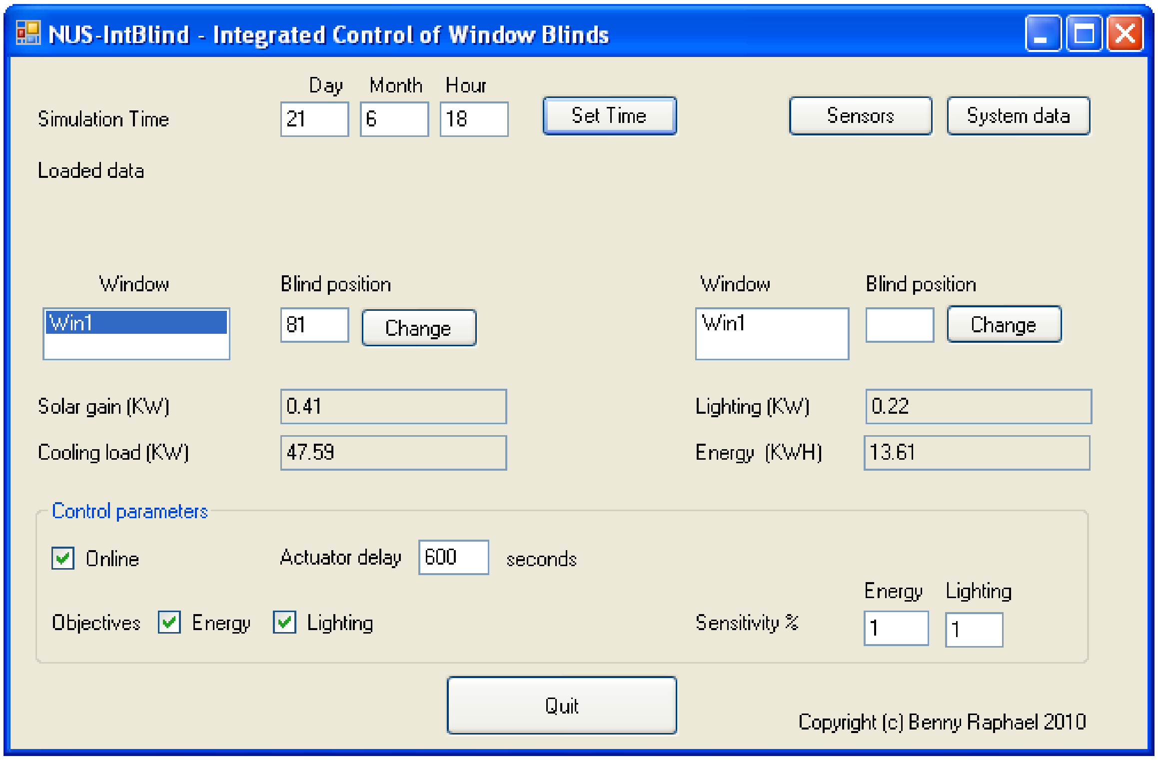


Figure 4.3a. Screen shot of Window Blind Control Application: Main screen

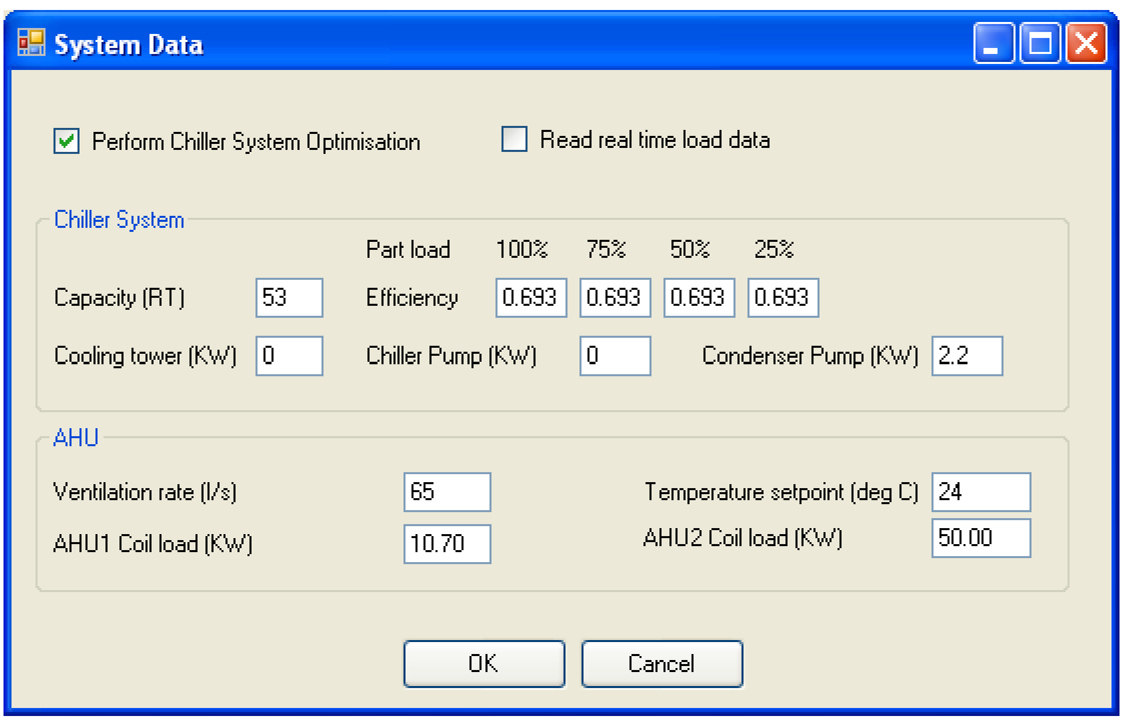


Figure 4.3b. Screen shot of Window Blind Control Application: System data

4.5 Hardware implementation: Laboratory prototype

A laboratory prototype of a motorized window blind has been implemented in order to demonstrate important aspects of the control strategy. The laboratory prototype does not fully represent a realistic scenario because of the limitations of space in the laboratory and other factors. However, it demonstrates that a control system based on global optimization is feasible.

A picture of the laboratory prototype of the system is shown in Figure 4.4a. The motor for raising or lowering the blind is activated by sending a command in the form of a direct current (DC) voltage. Desired position of the blind is obtained by activating the motor for the required duration. The motor that is currently being used does not provide feedback related to the actual number of rotations performed. It is also not possible to specify the number of rotations to be performed. The motor can only be turned on or turned off. The number of rotations of the motor and the eventual position of the blind are estimated by measuring the time taken for complete movement from the top to the bottom during the calibration stage.

The motorized blind is connected to a direct digital controller (DDC) which accepts commands from a desk top computer using the BACNET protocol. This is achieved using the BACNET library routines (Section 3.4).

Figure 4.4a Laboratory prototype of a motorized window blind

Figure 4.4b An array of light bulbs to simulate solar radiation

4.5.1 Limitations

The laboratory prototype has several practical limitations and does not closely resemble a full fledged building system. First of all, the window blind is installed in the interior of the building and not on an external window. Since the external windows of this room are shaded and not much solar heat comes through, solar heat and light are simulated using an array of light bulbs mounted on a metal frame (Figure 4.4b).

Secondly, existing air conditioning system could not be interfaced with the new control system. This is because of concerns related to warranty and the possibility that original equipment suppliers might refuse to perform repairs if their equipments are altered. Since the laboratory facility is being used by other projects, it was not advisable to make any changes to these. It was not possible to interface with existing Building Management System (BMS) programmatically because the system was proprietary and the supplier did not provide any application program interface (API). It was not even possible to get sensor readings from the system in real time. The readings could only be obtained by using the graphical user interface of the BMS (Building Management System) and the data had to be exported manually. It was not possible to use these for real time control.

The generic control software was customized to take into account the limitations of the laboratory prototype. Some of the changes that had to be made in order to overcome the limitations of the laboratory prototype are explained in the following sections.

4.5.1.1 Prediction of cooling load

The heat component of the power output of the bulbs for different dim levels of the light bulb array cannot be calculated theoretically. Therefore, the simulation results are replaced with actual measurements and these are used in the control software. The heat load from the bulbs is measured using a heat flux sensor placed outside the window blinds. The heat flux value is multiplied by the window area in order to estimate the radiant heat transmitted.

4.5.1.2 Prediction of lighting

The light from the bulb array comes roughly in the same direction and does not depend on the time of the day. The light levels cannot be easily predicted through simulation since the characteristics of the bulb are not known. Therefore, the simulation results are replaced with actual measurements. The lux levels at two points are measured using light sensors for different positions of the blind under two ambient light conditions (to simulate clear sky and cloudy sky). Real time measurements from these sensors are used to scale the indoor light levels for various dim levels of the bulb array. The actual light level in the laboratory depends on background light coming through the external window and other electrical lights. The prediction using two sensors provide a reasonable approximation.

4.5.1.3 Actual power consumption

Since the control system was not interfaced with the existing air conditioning and lighting system, the actual power consumption could not be obtained in real time. However, in order to simulate a real building system the following steps were taken.

1. The number of electrical lights in the room and their maximum power are input by the user. It is assumed that all the lights are equipped with dimmable ballasts and light sensors. Linear reduction in power through dimming is assumed.
2. Electrical equipment loads are input by the user.
3. The number of occupants in the room is input by the user. Heat load of occupants are calculated using a standard value of 130 W/person (combined latent and sensible heat load).
4. A constant fresh air supply of 10 l/s/person was assumed. If total fresh air supply is less than 0.13 l/s/area this is chosen as the minimum fresh air supply.
5. The cooling load of the AHU is calculated by summing up the external heat gain, occupant load, ventilation load, electrical equipment load and lighting load.
6. In order to study the effect of the load on other AHUs served by the same chiller, the user inputs the cooling load of other AHUs.
7. The part-load performance curve of the chiller was input using data from a manufacturer. This does not correspond to the real chiller used in the building, whose data was not available.

4.6 Testing and validation

4.6.1 Tests using the laboratory prototype

The laboratory prototype was tested under different conditions of ambient lighting and dim levels of the bulb array. The control system worked correctly in all the cases and the blinds were raised or lowered automatically according to expected patterns. The following observations summarize the experimental tests

- When the bulb array was turned on at full power and the chiller was loaded at about half the maximum load, the window blind closed completely. This was because the increase in cooling load was very high when the blind opened and the brightness was much higher than permitted values
- When the bulb array was switched off, the window blind was completely opened. This is because there is no appreciable heat transmitted through the window and the daylight was not adequate
- When the bulb array was turned on at low power and the chiller was below minimum part load, the window blind was completely opened. This was because the heat transmitted did not cause any increase in the chiller power, while it improved the lighting

- When the bulb array was turned on at medium power level and the chiller was around 50% part load, the window blind was partially opened. This was because the control system allowed increased heat transfer up to a point that was permitted according to the trade-off computed using the sensitivity parameter for the energy objective.

4.6.2 Hypothetical test case for evaluation

In order to bring out the advantages of the integrated control in a realistic situation, a case study of an office building is taken. Two control strategies are compared. The first one is the multi-objective control strategy that was developed in this project. The second strategy is similar to the one that is commonly used in window blind control. This is based on limiting the maximum brightness in the room. In this strategy, the window blind is left completely open as long as the lux values in the room are below recommended values. If the lux values are higher than the limiting value, the window blind is completely closed. 600 lux is taken as the limiting value in this study.

In the multi-objective control strategy, the optimal blind positions for all the hours of a typical summer day from 8 am to 6 pm are computed, using total energy as the primary objective function and the lighting energy as the secondary objective. For each hour, the energy consumption of the optimal control action is computed. This energy is compared with that of the second control strategy. The difference between these two cases gives the energy savings that can be achieved using the integrated control strategy.

The case study involves the first floor of an office building of size 36m x 18m, with the longer side oriented along the north-south direction. There are windows with controllable blinds on the east and west facades, named W1 and W3 respectively. The window height is 2.4m and the ceiling is at a height of 2.7 m from the floor. A 3D rendering of the building using Radiance is shown in Figure 4.5. The XML file containing the complete building model is given in Appendix A.

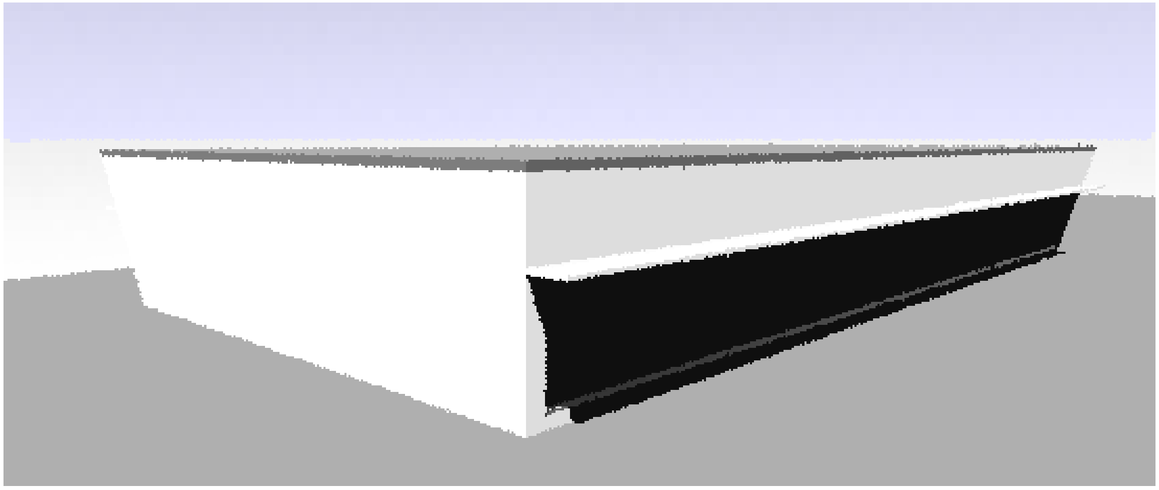


Figure 4.5. Case study of an office building rendered using Radiance

Table 4.1 summarizes the energy computations for all the hours of the day. The second and third columns contain the optimal blind positions determined by the control algorithm. The fourth column gives the lighting power and the fifth column the cooling load. The sixth column gives the total energy for the optimal blind positions. The last column contains the energy for the second control strategy.

Hour	Blind position		Lighting (KW)	Cooling (KW)	Total Energy (KWH)	
	W1	W3			Integrated Control Strategy	Control strategy 2
8	100	63	5.52	11.09	20.57	29.5
9	49	45	5.91	11.48	21.35	30.08
10	26	39	6.06	11.73	21.75	30.47
11	100	40	5.74	12.28	21.97	30.61
12	100	78	5.45	12.59	22	30.52
13	100	100	5.31	12.67	21.95	21.95
14	85	96	5.39	12.7	22.05	30.63
15	49	18	6.02	12.26	22.24	31
16	37	21	6.09	12.13	22.19	31.21
17	41	30	5.99	11.93	21.88	30.83
18	55	76	5.61	11.48	21.05	29.97

Table 4.1. Summary of energy computations for the optimal blind positions

The total savings in energy for the whole day with respect to the second control strategy is 26.87%. A plot of the total energy for the two control strategies is given in Figure 4.6.

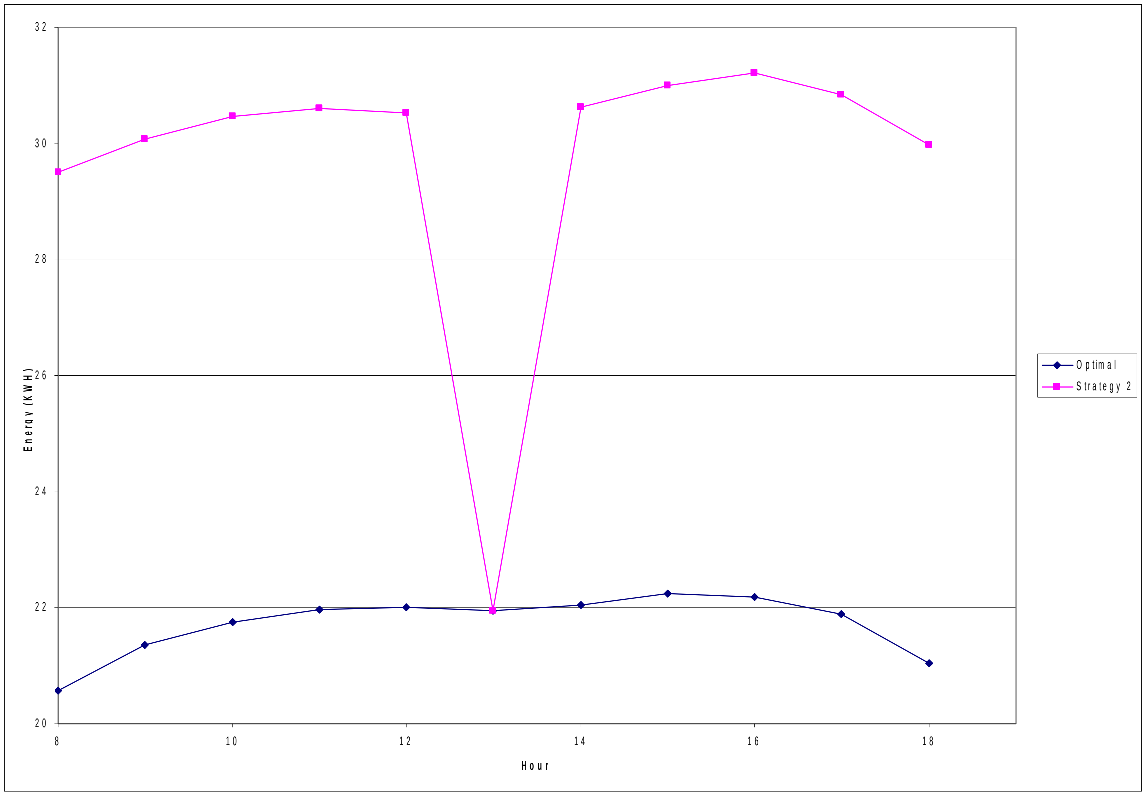


Figure 4.6. Comparison of energy for two different control strategies

It can be seen that for every hour, the optimal control results in lower energy consumption. When the blinds are fully open, either too much solar gain causes the cooling energy to be higher or the excessive brightness causes the blinds to be closed, thereby increasing the lighting energy consumption. The only exception is at 13:00 hours, when the shading is just adequate to prevent excessive heat and light, causing a dip in the energy consumption of the second control strategy. Only at this hour, the external shade of 0.6 m width prevents direct sunlight from entering the room.

It should be noted that the energy savings vary depending on a number of factors such as the building orientation, position and size of windows, types of shading provided, etc. Nevertheless, tests have shown that significant savings can be achieved through integrated control in most cases.

4.7 Conclusions

The integrated control strategy using multi-objective optimization is shown to be feasible through a fully functional laboratory prototype. Energy saving potential of the strategy is evaluated using the case study of a hypothetical office building.

References

Diakaki, C., Grigoroudis, E., Kolokotsa, D., (2008). Towards a multi-objective optimization approach for improving energy efficiency in buildings, Energy and Buildings 40 (9), pp. 1747-1754.

DOE-2.1E (Version-119) , (2002). Building Energy Simulation Code, Lawrence Berkley Laboratories (LBL), Berkley, CA, April 9.

EnergyPlus 1.2.3.023, (2005). A building energy simulation program, www.energyplus.gov.

Freewan A.A., Shao, Riffat S. (2008). Optimizing performance of the lightshelf by modifying ceiling geometry in highly luminous climates, Solar Energy 82 343–353.

Gero J.S., Neville D.C., Radford A.D., (1983). Energy in context: a multicriteria model for building design, Building and Environment 18 (3) 99–107.

Guillemin A., Morel N. (2001). An innovative lighting controller integrated in a self-adaptive building control system, Energy and Buildings 33, 477-487.

Lee E.S., Bartolomeo D.L., Selkowitz S.E., (1998). Thermal and daylighting performance of an automated Venetian blind and lighting system in a full-scale private office, Energy and Buildings 29 47-63.

Leslie R.P., (2003). Capturing the daylight dividend in buildings: why and how? Building and Environment 38, 381–385.

Loutzenhiser, P.G., Manz, H., Moosberger, S., Maxwell, G.M., (2009). An empirical validation of window solar gain models and the associated interactions, *International Journal of Thermal Sciences* 48 (1), pp. 85-95.

Park, K.-W., Athienitis, A.K., (2005). Development and testing of an integrated daylighting control system, ASHRAE Transactions, Volume 111 PART 1, Article number 4762, Pages 218-226.

Ochoa, C.E., Capeluto, I.G., (2006). Evaluating visual comfort and performance of three natural lighting systems for deep office buildings in highly luminous climates. Building and Environment 41, 1128–1135.

Raphael B. and Smith I.F.C, (2003). Fundamentals of computer aided engineering, John Wiley, UK.

Tsangrassoulis, A., Santamouris, M., Asimakopoulos, D., (1996). Theoretical and experimental analysis of daylight performance for various shading systems. Energy and Buildings 24, 223–230.

Ward Larson G, Shakespeare R. (1998). Rendering with Radiance: the art and science of lighting visualization, San Francisco: Morgan Kaufmann.

Wright J.A., Loosemore H.A., Farmani R. (2002). Optimization of building thermal design and control by multi-criterion genetic algorithm, Energy and Buildings 34 (9) 959–972.

Ziva Kristl, Mitja Kosir, Mateja Trobec Lah, Ales Krainer, (2008). Fuzzy control system for thermal and visual comfort in building, Renewable Energy 33 694–702.

5 Control of light shelves [6]

Summary

This chapter describes the application of the control strategy based on global optimization to the configuration of adaptable light shelves. A new type of adaptable light shelves was developed in this research. This light shelf consists of an external part that can be rotated and an internal part that can be moved horizontally. The optimal configuration for the light shelf for a given position of the sun and environmental conditions is determined through global search. Tests that were conducted using simulations show that such adaptable light shelves have the potential to save significant amount of energy.

5.1 Introduction

Over the years, many passive design features have been developed that maximize the daylight in buildings. One such passive design element is a light shelf. A light shelf is a horizontal or inclined projection with a high reflectivity meant to increase the depth of daylight penetration into a room. It operates by reflecting sunlight off to the ceiling from where it is further reflected to the workplane (Figure 5.1).

Several studies have been conducted on the effectiveness of light shelves and other daylighting features. Aghemo et al. (2008) presents a case study for the comparison of lighting performances of different traditional shading devices. Scaled models are tested in a specially designed facility for simulating the building performance under different sky conditions. Carlos and Soler (2002) studied the influence of light shelf parameters and the solar elevation and azimuth on the performance. Other studies include Sarajii and Mistrich (1993); Aizlewwod (1993); Oteiza and Soler (1995); Littlefair et al. (1995); Reinhart and Herkel (2000); and Soler and Oteiza (1997).

Freewan et al. (2008) state that there is a controversy over the potential of light shelves to increase illuminance levels in the rear of a room (For example, see Littlefair et al., 1994). Improvements in illumination conditions and energy savings are not always achievable. Careful design is essential for obtaining adequate performance.

Usually, light shelf design is carried out using general principles and design options are not evaluated systematically through optimization and search (Raphael and Smith, 2003a). Literature on the optimization of light shelves is rare. Freewan et al. (2008) investigated the impact of ceiling geometries on the performance of light shelves through evaluating the illumination levels achieved by different values of parameters. However, they have not formally treated it as a global optimization problem and they have not used

[6] The system described in this chapter is covered by a provisional patent application filed in the USA.

any general purpose optimizing engine. Similar parametric studies are found in Ochoa and Capeluto (2006).

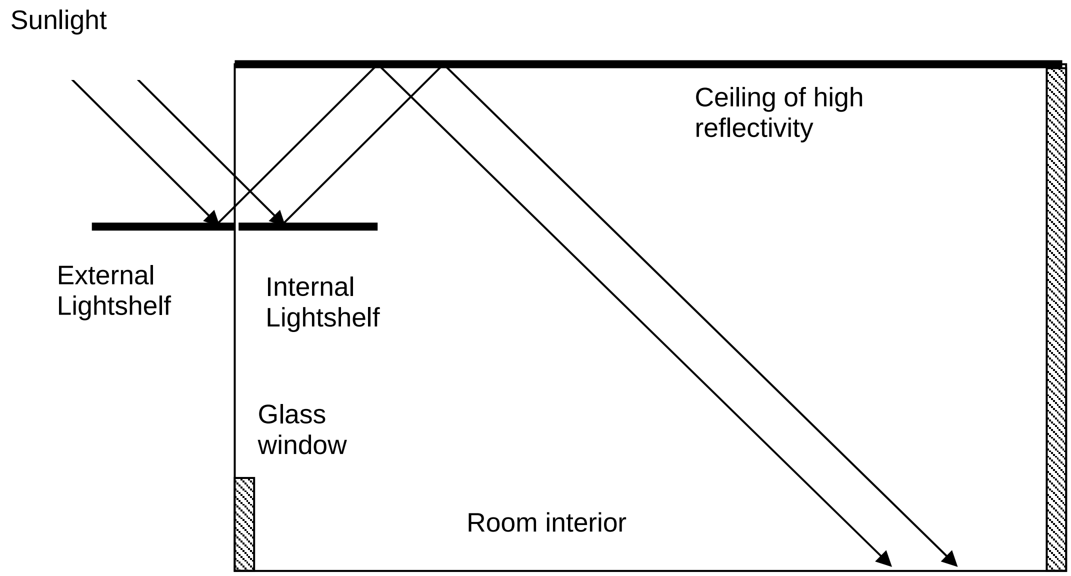


Figure 5.1. Cross section of a room. The light shelf reflects sunlight to the ceiling from where it is further reflected to the workplane.

The controversy related to the effectiveness of the light shelf is probably because it is treated as a passive design element. It is designed to maximise the average distribution of daylight during the operating hours of a building and its geometry is not adapted to the changing conditions during the day. Since the position of the sun as well as the sky conditions change with time, more effective distribution of daylight is possible by changing the geometry of the light shelf dynamically. No publications that report active control of light shelf geometry could be found in the literature. There are discussions related to the control of other devices such as roller shades (for example see, Tzempelikos and Athienitis, 2007; Kristl et al., 2008). Since these are shading devices, they can easily be controlled by setting a limit on variables such as brightness. On the other hand, the control of light shelves requires optimization for maximizing the distribution of daylight.

This chapter presents a methodology for the active control of light shelves. A light shelf system whose geometry can be adapted has been developed. Geometrical parameters of the light shelf are computed for a given time of the day such that the energy required for artificial lighting is minimized.

The organization of the chapter is as follows: Section 5.2 presents the details of a new light shelf system whose geometry can be adapted to the environmental conditions. The control algorithm is described in Section 5.3. Results of applying the control strategy to a hypothetical office building is presented in Section 5.4. Section 5.5 contains conclusions.

5.2 A light shelf system with adaptable geometry

A standard light shelf consists of two parts, a part which is on the exterior of the building called the *external light shelf* and a part which is in the interior of the room known as the *internal light shelf* (Figure 5.1). Together, they serve the dual purposes of providing shading near the window and reflecting daylight deep into the interior of the room. If the light shelves are horizontal, they are able to increase daylight penetration only when the sun is at a low angle. By making the external light shelf rotatable, its angle can be adjusted to the position of the sun to produce deeper daylight penetration (Figure 5.2).

It may not be practical to make the internal light shelf rotatable. Rotating it downwards would reduce the clear height near the window and render the space unusable. Rotating it upwards is not usually beneficial because this would reflect light away from the room. However, in certain cases this is very effective to eliminate glare.

It is found that the width of the internal light shelf has a significant effect when the sun is at certain angles. This is because the internal light shelf plays a crucial role in eliminating excessive brightness near the window and cutting down glare. When there is a possibility of direct sunlight entering the room through the top part of the window, it is better to increase the width of the internal light shelf for eliminating uncomfortably high illuminance. On the other hand, when less daylight is available it is more beneficial to reduce the width of the internal light shelf to maximize the brightness in the room.

From the above considerations, the light shelf configuration shown in Figure 5.3 is adopted in this work. The external light shelf is rotatable using a motor and its angle φ can be varied between $-\frac{\Pi}{2}$ and $+\frac{\Pi}{2}$. The internal light shelf consists of two parts; ***A*** and ***B.*** The bottom part ***B*** is fixed, but the top part ***A*** can be moved horizontally. The width of the internal light shelf ***W*** is varied by moving ***A*** using a motor. More variation in the width is possible by adding more moveable parts like ***A.***

Another possibility is to use rollers similar to roller blinds on windows. The roller blinds can be made of reflective material and can be moved horizontally using motors. As the blinds open up, the width of the internal light shelf is increased. This idea is planned to be evaluated by building a fully functional prototype in a future project.

The external light shelf is similar to a horizontal louvre that can be rotated. Such moveable louvres have been successfully installed in many buildings. Even though practitioners are generally reluctant to adopt moveable parts in the exterior of a building due to maintenance problems, it is expected that careful detailing can eliminate most of these problems.

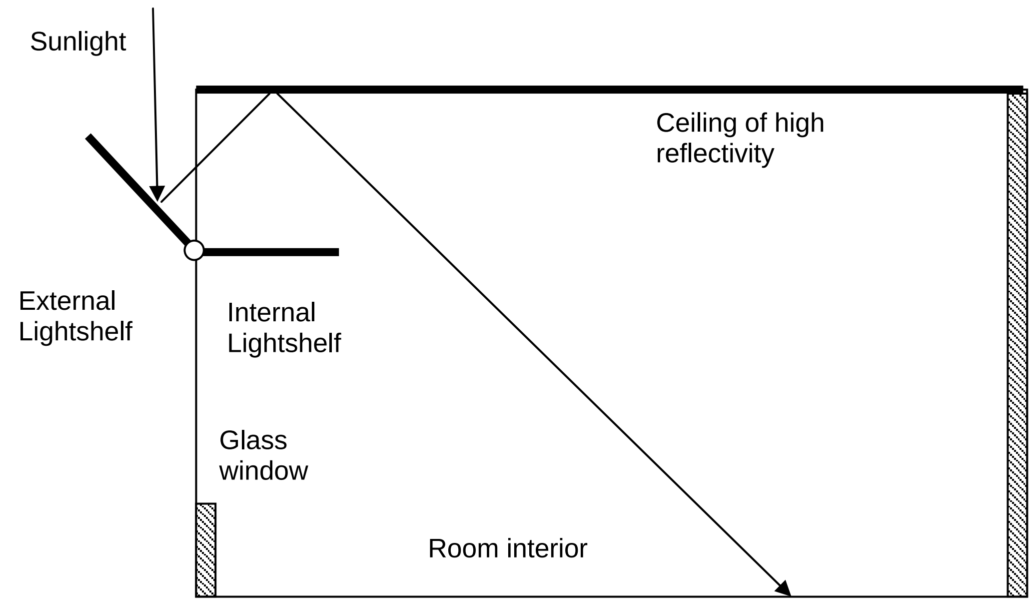


Figure 5.2. Rotatable external light shelf. By adapting the angle of the external light shelf to the position of the sun, deeper penetration of daylight is possible even at high sun angles.

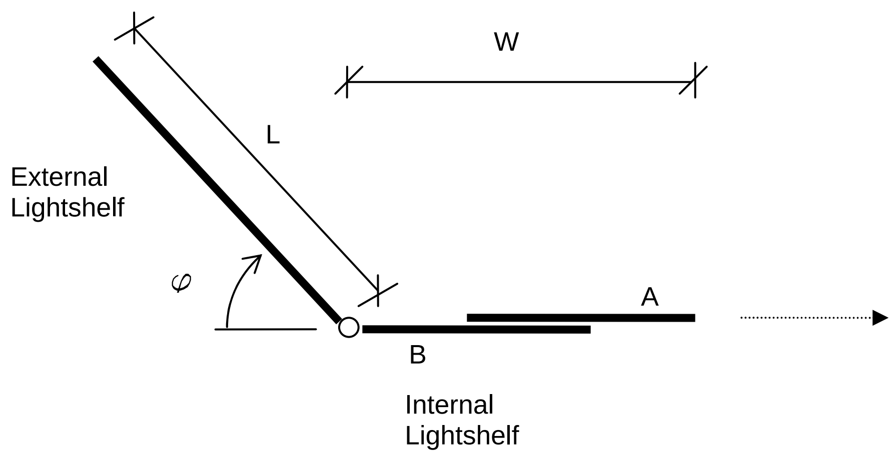


Figure 5.3. An adaptable light shelf. The external light shelf can be rotated. The width of the internal light shelf can be varied by moving the top part A. The bottom part B is fixed.

5.3 An algorithm for the control of light shelf geometry

In traditional closed-loop control systems, actuators are operated using feedback such that the output changes in the required direction. This is not appropriate for the control of light shelf geometry because the effects are not monotonic with respect to control variables. For example, when the external light shelf is rotated in the positive direction, initially there is an increase in daylight penetration and later at greater angles, daylight decreases depending on the angle of the sun. There is an optimal position of the light shelf at which adequate shading is provided near the window and maximum daylight is produced in the interior. The complexity of the control task is further increased by the interaction between the external and internal light shelves.

The right approach is to model this control task as a global optimization problem. In this approach, the goal is to select the right combination of values of control variables in order that a user-defined objective function is minimized (or maximized). Here, the control variables are the angle of the external light shelf and the width of the internal light shelf. If there are *m* light shelves in a room, the total number of control variables *N* will be equal to 2*m. Since light from different shelves interact in a complex manner, each combination of light shelf parameters is a potential solution that needs to be evaluated. The size of the search space is thus *pN* where *p* is the number of discrete values for each light shelf parameter. The size of the search space increases exponentially with respect to the number of control variables.

The objective function to be minimized might be chosen as the energy consumption of artificial lights subject to a set of constraints; other formulations of the objective function are also possible. An example of a constraint is: the lux level near the window should be less than a specified value. Restricting the brightness is necessary not only to avoid glare discomfort, but also to avoid the perception of reduced brightness in the interior because of the contrast.

There are several global optimization algorithms available. The most popular ones are undoubtedly genetic algorithms (Holland 1975) and their variations (Bäck and Schwefel, 1993; Adeli and Cheng, 1994a&b; Sarma and Adeli, 2000a&b, 2001, 2002; Jiang and Adeli, 2008). The global optimization algorithm used in this work is called PGSL (Probabilistic Global Search Lausanne) (Raphael and Smith, 2003b). It works by generating and testing potential solutions through sampling using a probability density function (PDF). The probability density function is dynamically updated in order to improve convergence. It has been shown that PGSL performs as well as or better than genetic algorithms for a wide range of practical tasks and benchmark problems (Raphael and Smith, 2003; Domer et al., 2003, Patil et al., 2004). PGSL demonstrates good convergence characteristics even when the number of variables is increased to 100, which is considered to be a large problem for non-linear optimization. Therefore, PGSL was judged to be an appropriate tool for solving the present optimization problem.

The PGSL algorithm operates in a manner similar to other search methods such as Genetic algorithms. In each iteration, the algorithm proposes a solution which consists of a combination of values of the optimization variables. Each solution is evaluated using the user-defined objective function. PGSL differs from other search methods in the way in which the evaluation of the objective function is used to influence the generation of future points. PGSL explicitly updates the PDF such that there is more probability that values are generated in the neighborhood of good points, without avoiding other regions altogether. This is done in four nested cycles namely, sampling, probability updating, focusing and sub-domain. Each cycle performs a different type of search. For example, the sampling and probability updating cycles perform predominantly uniform search, where as, faster convergence is achieved using focusing cycle. The sub-domain cycle helps to avoid local optima by re-starting the search process. The evolution of the PDF during these cycles is illustrated in Figure 5.4.

In order to solve an optimization problem, users specify the bounds and precision of all the variables, define the objective function and specify the values of algorithm parameters such as the maximum number of evaluations of the objective function. The objective function takes an argument which is the solution point to be evaluated and consists of a set of values for all the optimization variables. All optimization variables are treated as real numbers and their values are generated randomly according to the PDF as explained earlier. Similar to other direct search methods, PGSL treats the objective function as a black-box. That is, the objective function defined by the user is called by the algorithm purely for evaluating potential solutions and is not used to extract mathematical characteristics of the problem such as convexity, gradient, etc. Constraints are incorporated into the objective function by adding a penalty proportional to the degree of constraint violation. More details are available in Raphael and Smith (2003b).

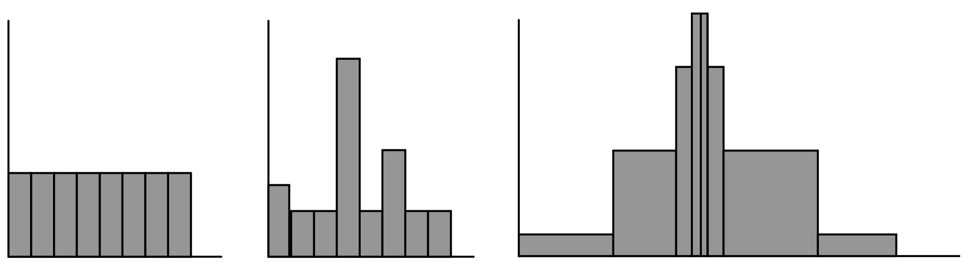


Figure 5.4. PGSL: Evolution of the probability distribution function of a single variable during the search.

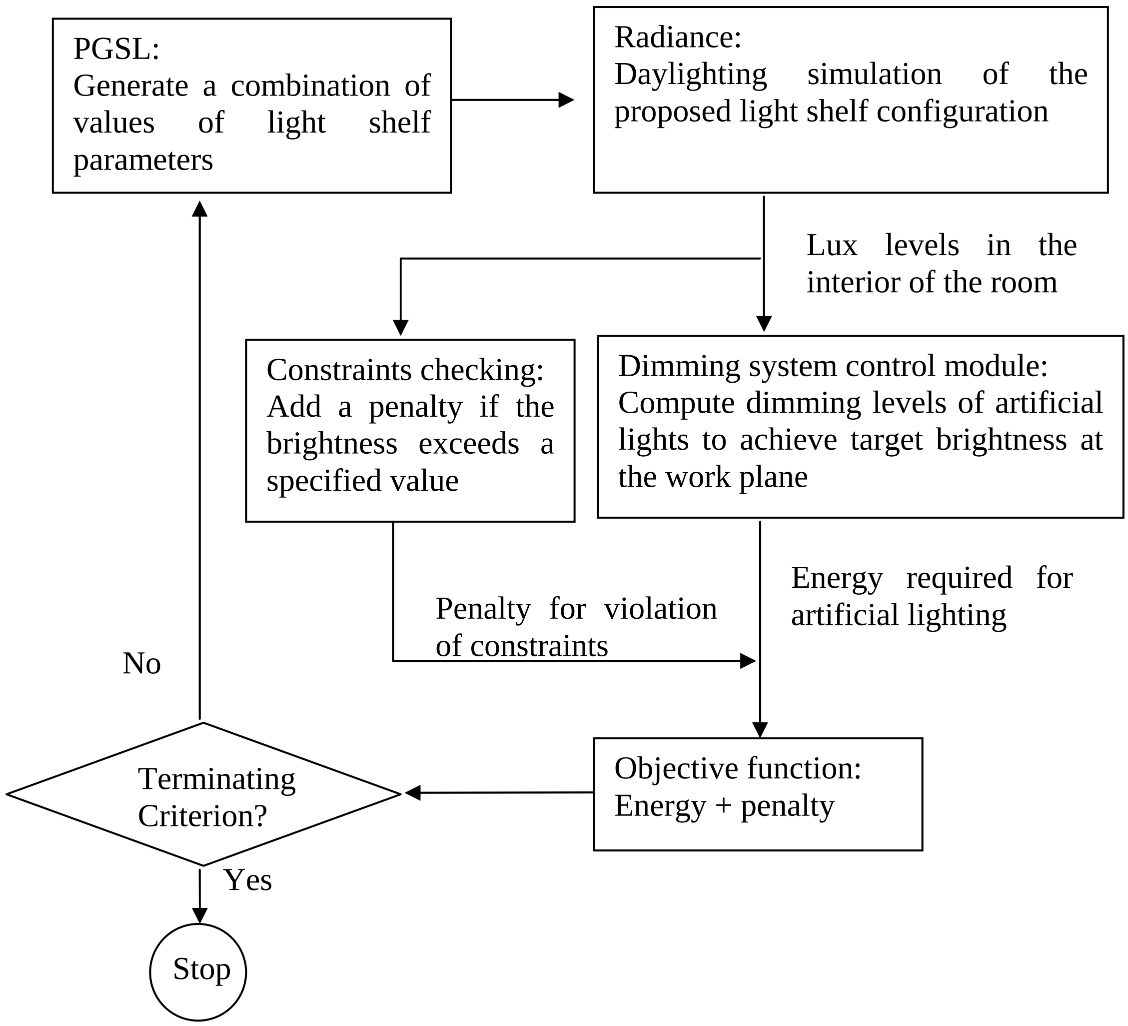


Figure 5.5. Flowchart of the optimization process

In this application, the primary optimization variables are the geometric parameters of the light shelves. The objective function involves the computation of the lux levels in the interior for the proposed light shelf configuration. A lighting simulation software called Radiance is used to perform this step (Section 3.2). After the lux levels are computed, the levels to which the artificial lights can be dimmed are computed. This requires another optimization and is described in Section 3.3. Once the dimming levels of lamps are known, the energy required for artificial lighting can be computed. The overall flow of the optimization procedure is summarized in Figure 5.5.

The following steps are involved in the control strategy:

1. Collect data related to the sun and sky conditions through sensors
2. Perform global optimization to compute the values of control variables

3. Apply the optimal control action
4. Collect feedback through light sensors at selected points within the room and apply minor corrections to the control variables
5. Repeat from step 1

Tasks involved in these steps are explained below.

5.3.1 Sky conditions

The most important data required for predicting the daylight in the room interior is the position of the sun and the sky brightness. Even though solar tracking devices are available, they are expensive and not recommended here. Instead, the position of the sun is obtained from weather data files which are found in sources such as the US department of Energy website7. These data files provide the solar azimuth and altitude for any time and day of the year for a given geographical location. However, the sky brightness varies depending on local conditions and should be measured on site. A simple method to measure sky brightness indirectly using outdoor light sensors is described below. Such sensors are typically installed on the roof pointing towards northern sky.

Two daylighting simulations are performed for each time of the day in order to obtain the predicted lux value at the point where the light sensor is installed; one for clear sky condition and the other for overcast condition. The values obtained from the sensor readings are compared with the predicted values in order to determine whether clear sky or overcast sky conditions should be used. After this, the sky brightness used in the selected simulation is proportionately scaled to produce a match between predicted and measured values.

5.3.2 Daylighting simulation

The lighting simulation software, Radiance (Ward and Shakespeare, 1995) is used in this project. The following are the data required for running lighting simulation:

- Sky conditions
- Geometric model of the building including the configuration of light shelves
- Material properties such as visible transmittance of glass, reflectivity of surfaces, etc.

With the above data, the software is capable of computing the illumination level at any desired point. Since the goal of running this simulation is to determine the amount of artificial lighting needed, the lux values are computed for every point on a grid of equal spacing within the room. The grid spacing is chosen to be 1 m. in both directions. This is adequate for obtaining a reasonable distribution of the variation.

[7] http://www.eere.energy.gov/buildings/energyplus/cfm/weather_data.cfm

5.3.3 Dimming levels of artificial lighting

Lamps fitted with dimmers are an essential component of the control strategy. This makes it possible to reduce energy consumption when more daylight is available. Today commercial lighting units in which power consumption is varied linearly with the brightness level are already available.

The control of dimmers is not straight forward because of the interaction between light from different sources at a point. If light sensors are installed at every grid point, the dimming levels can be adjusted iteratively to produce the required lux at all the points. However, this is expensive and requires many sensors. Commercial dimming systems avoid this problem by installing light sensors on ceilings which measure the average illuminance over a wider area. The energy consumption of such systems is difficult to predict a-priori. A control strategy that does not require any light sensors and is based purely on simulation is presented here. It should be noted that the primary objective here is to obtain a theoretical estimate of the amount of energy required for artificial lighting before performing dimming in the actual system.

The following data are needed:

- Lamp positions and their characteristics such as brightness and power
- Geometric model of the room
- Material properties such as reflectivity of surfaces

One lighting simulation is performed for each lamp under night time conditions (when no daylight is available). The lux value produced by the lamp at each grid point is obtained. After completing all the simulations, the results are compiled in the form of an influence matrix *I* of dimensions ng x nl, where ng is the number of grid points and nl is the number of lamps. In this matrix, a row represents the influence of all the lamps at a grid point. That is, each row of the matrix contains the lux levels at a grid point when individual lamps are turned on, one at a time. Each column of the matrix contains the lux level at all the grid points when only one lamp is turned on. Thus the element of the matrix $I(j,i)$ is the lux level at the grid point j due to the lamp i.

When a lamp is dimmed to a certain level, the lux values at all grid points are proportionately scaled by multiplying the corresponding column of the influence matrix by the dimming level. The combined effect of all the lamps is calculated through linear superposition of the lux values corresponding to each lamp.

Let di represents the dimming level of the i-th lamp in the range [0,1]. A dimming level of 0 indicates that the lamp is completely turned off and a value of 1 indicates maximum brightness. The combined effect of dimming all the lamps at the grid point j is obtained by the equation

$$LampLux_j = \sum d_i \cdot I_{(j,i)} \qquad \text{--- (1)}$$

If Sj is the illuminance due to daylight at this grid point at a particular time, the total illuminance is obtained as

$$Lux_j = S_j + \sum d_i \cdot I_{(j,i)} \qquad \text{--- (2)}$$

This value should be greater than the target illuminance, DesignLux. This condition is expressed as

$$S_j + \sum d_i \cdot I_{(j,i)} \geq DesignLux$$

; for all j from 1 to ng. -- (3)

If Pi is the power consumption of the i-th lamp at full brightness, the total power consumed by all the lamps (assuming linearly varying dimming power) is given by

$$TotalPower = \sum d_i \cdot P_i \qquad \text{--- (4)}$$

Therefore, the following linear programming problem is formulated:

Minimise *TotalPower* given by Equation (4)

Subject to constraints given by Equation (3) --- (5)

By solving the linear programming problem, the required dimming levels of all the lamps are obtained. The total power for this combination of dimming levels is computed by Equation (4).

5.3.4 Objective function

The objective function is called by PGSL with a proposed set of values for the optimization variables (decision variables) namely, geometrical parameters of the light shelves. The objective function cannot be expressed in explicit mathematical form in terms of the decision variables because it is computed using results from lighting simulation. The values of decision variables are used only to prepare the data files needed for the simulation and not for computing the value of the objective function directly. The following are the steps involved in the computation of the objective function:

- The geometric model of the building is updated using proposed values of light shelf parameters.
- The data files needed to run lighting simulation are created. The artificial lights are not included in the simulation, since they are treated separately through the influence matrix.
- The lighting simulation software is executed as an external program and the output in terms of the lux values at all the grid points are read (Section 3.2).
- Lux values at all grid points are examined. If the lux values exceed the limit specified in the constraints, a penalty is computed proportional to the degree of constraint violation.
- The dimming levels of artificial lights are computed as described in Section 3.3.
- The total power consumed by artificial lights is calculated as described in Section 3.3.
- The value of the objective function is calculated as the sum of the penalty and the power consumed by artificial lights.

Even though, currently glare index is not explicitly computed, glare is treated implicitly by specifying a constraint that limits the maximum lux level. However, it is easy to add a new constraint based on the value of glare index.

The value returned by the objective function is used by PGSL to update the probability distribution and this influences the regions where generate future points are generated.

5.4 Results

In order to evaluate the control strategy, a hypothetical office building is considered here. The dimensions and other data are taken from a real office building in Singapore. Only a part of the building consisting of an open office in one floor is modeled here. The orientation and dimensions are shown in Figure 5.6. The building is located at latitude 1.4o and longitude 104o east. The Sun path diagram for this location is shown in Figure 5.7. The following material properties were used:

- Window glass below light shelf: visible transmittance of 0.35
- Window above light shelf: Clear glass with a visible transmittance of 0.86
- Reflectivity of interior surfaces: 0.6
- Reflectivity of ceiling: 0.9
- Reflectivity of light shelves: 0.9

Three system configurations are studied. In the first one, conventional light shelves consisting of horizontal immovable parts are installed on the east and west facades. In the second configuration, there are adaptable light shelves on the east and west facades. However, only the external light shelves are moveable; the internal light shelf is fixed and has a constant width of 1.2 m. This configuration was used to study the effect of making the internal light shelf moveable. In the third configuration, both internal and external light shelves are moveable as described in Section 5.2.

Energy required for artificial lighting for the first configuration is computed by performing lighting simulations for various times of the day from 10 am to 4 pm at one hour intervals. Power consumed at each hour is calculated by solving the linear programming problem described by equation (5). Since light shelves are most effective under bright sunny conditions, the simulations are performed for a clear sunny day using Singapore weather data files.

In the second configuration, there are only two control variables, namely, the angles of the external light shelves on the west and east facades. The best combination of values for these variables are obtained for each hour of the day from 10 am to 4 pm through global optimization as explained in Section 2. The total energy for artificial lighting during the entire simulation period is calculated by summing up the energy for each hour.

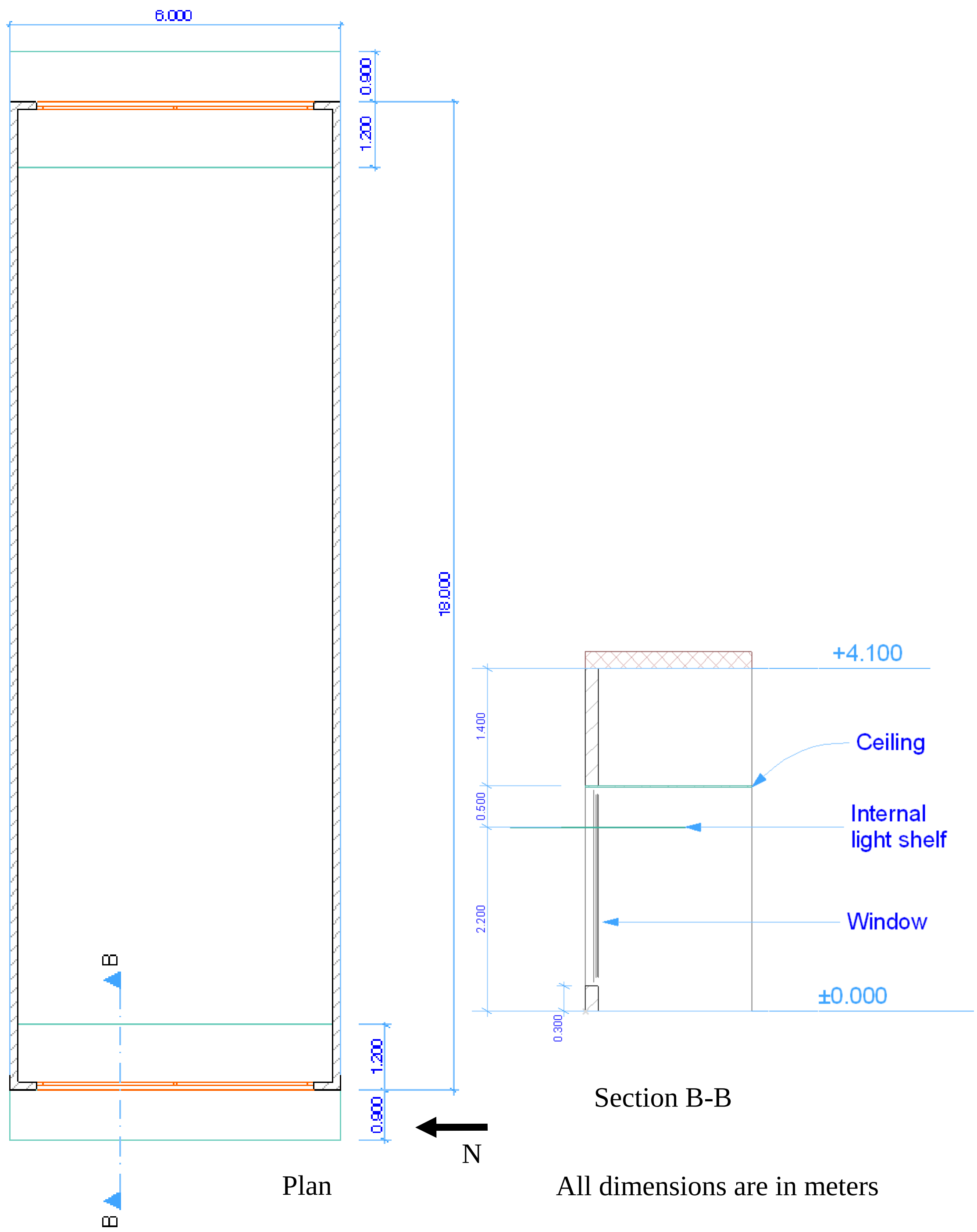


Figure 5.6. Dimensions of the office building.

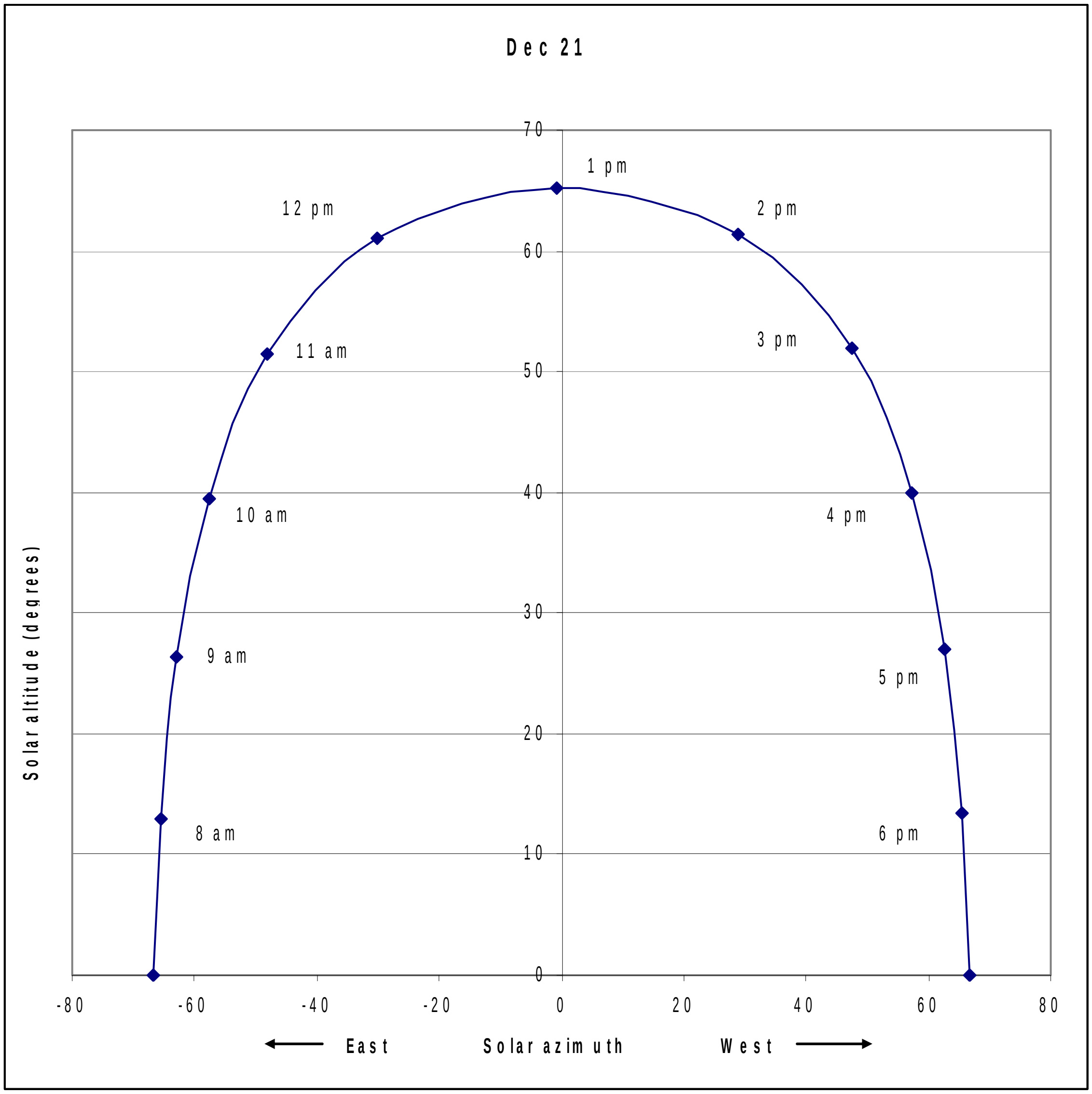


Figure 5.7. Sun path diagram for Singapore on December 21

In the third configuration, there are four control variables, namely, the angle of external light shelf and the width of internal light shelf for each façade. Similar to the second configuration, the optimal values are determined and the total energy for artificial lighting during the entire simulation period is calculated.

The artificial lighting used in all the three cases consists of three rows of luminaires at a uniform spacing of 2 m. Each luminaire consists of two 28 Watt T5 lamps designed to produce an average night time illuminance of around 500 Lux at the workplane. The total light power density (LPD) for the system is 14 W/m2.

Traditional static light shelves

The hourly power consumption of artificial lights after daylight responsive dimming is shown in Table 5.1. Typical distribution of daylight in the interior of the room at 14:00 hours is shown in Figure 5.8. At this time, the sun is on the west and hence, the light intensity is higher near the west window. Reflected light from the light shelf is clearly seen as a peak near a distance of 2 m from the west side. The average lux value for the whole area is 260.

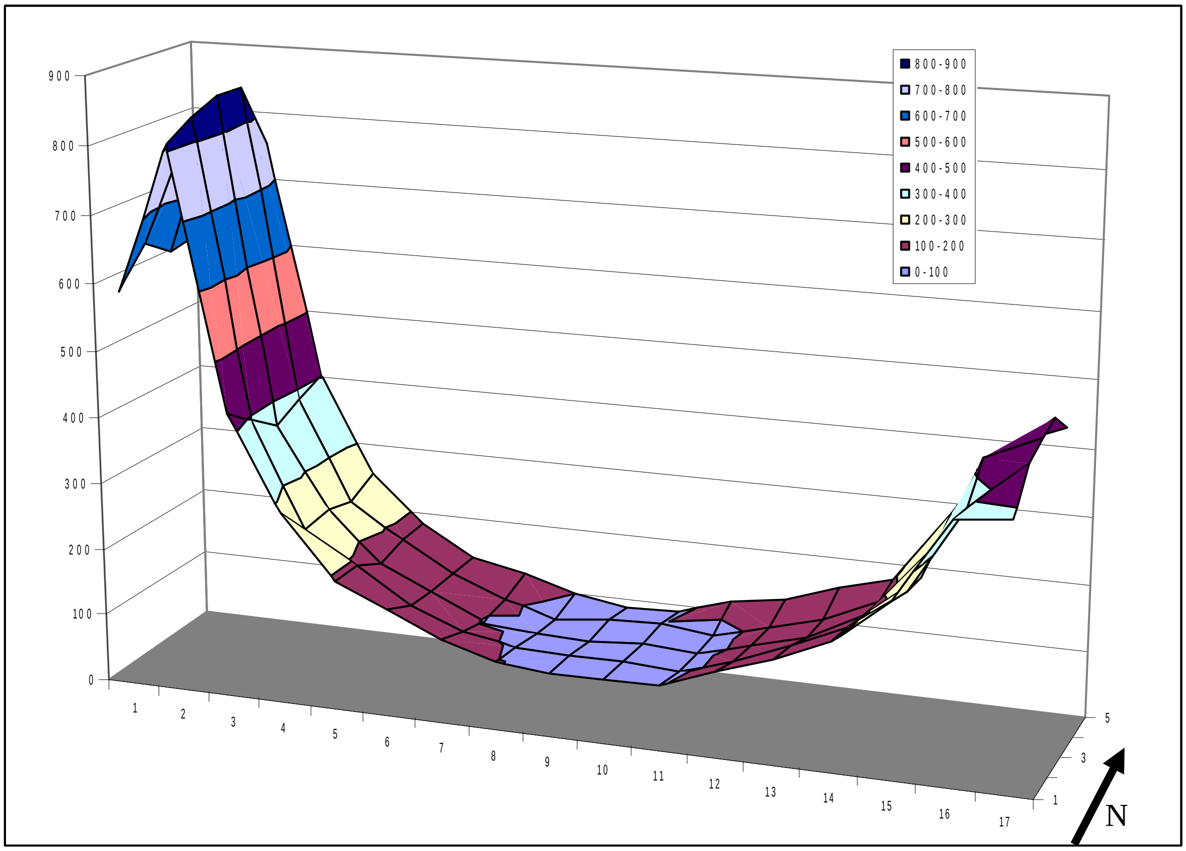


Figure 5.8. Lux levels at the work plane using a static light shelf.

Configuration 2: Rotatable external light shelves

The optimal values for the angles of external light shelves on the west and east are shown in Table 1. In the morning when the sun is on the east, the eastern light shelf is tilted upwards in order to reflect light to the interior of the room. In general, the light shelf angle decreases as the sun angle increases. Small deviations from this general trend are due to the discrete positions of the lamps and local effects such as glare at specific angles. The same trend is noticeable after noon on the west side.

hour	Angle of West Light Shelf	Angle of East Light Shelf
10	-6	49
11	-6	29
12	10	33
13	37	32
14	39	22
15	39	-6
16	54	-5

Table 5.1. Optimal values of light shelf parameters on the west and east

In order to illustrate the nature of the objective function, a sample of random solutions generated during the optimization is shown in Figure 5.9. This optimization was performed for 11 am. Only one optimization variable, angle of east light shelf, is shown here. The variation of the lighting power (y axis) with respect to the angle of the external light shelf on the east (x axis) is shown. Since the second optimization variable (angle of west light shelf) is not shown in this graph, sometimes there is more than one point along a vertical line. The minimum value of the objective function is when the angle of east light shelf is equal to 29o.

The search space in this example is small and it is possible to perform an exhaustive search to find the optimal solution. However, as the number of light shelves increase, the search space grows exponentially and it may not be possible to identify solutions through exhaustive search. Especially, when several light shelves are placed close to each other on one side of the building and when the solar azimuth is non-zero, light from different shelves interact strongly making the search task very complex.

Configuration 3: Moveable external and internal light shelves

The optimal values of light shelf parameters for this configuration is shown in Table 5.2. It can be noticed that in the morning, when the sun is on the east, the east light shelf is tilted upwards. In the afternoon, the west light shelf is tilted upwards. A perspective view of the west façade at 16:00 hours is shown in Figure 5.10.

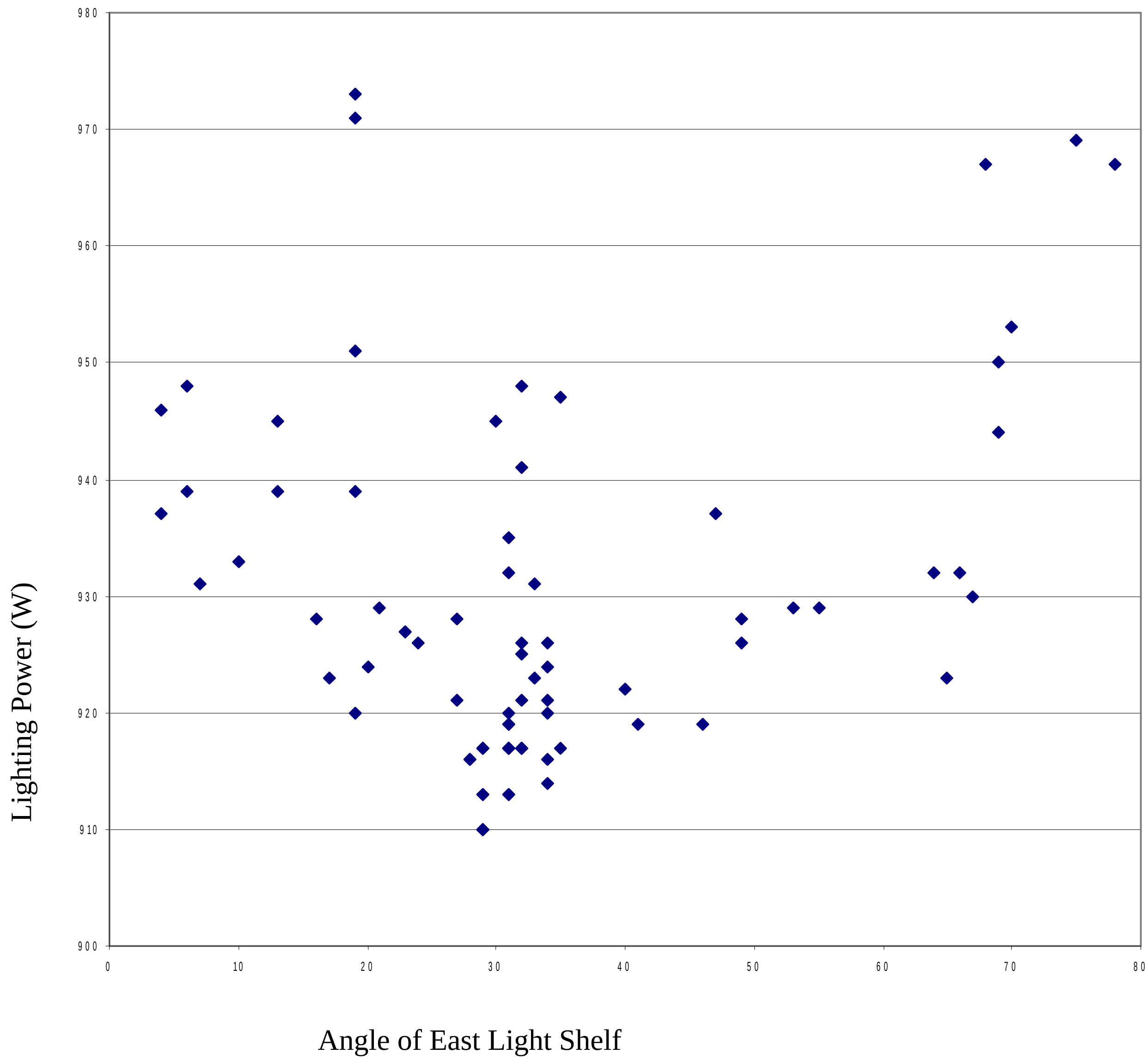


Figure 5.9. A random sample of solution points generated during the optimization for the light shelf configuration-2 at 11 am. Only one optimization variable (angle of east light shelf) is shown.

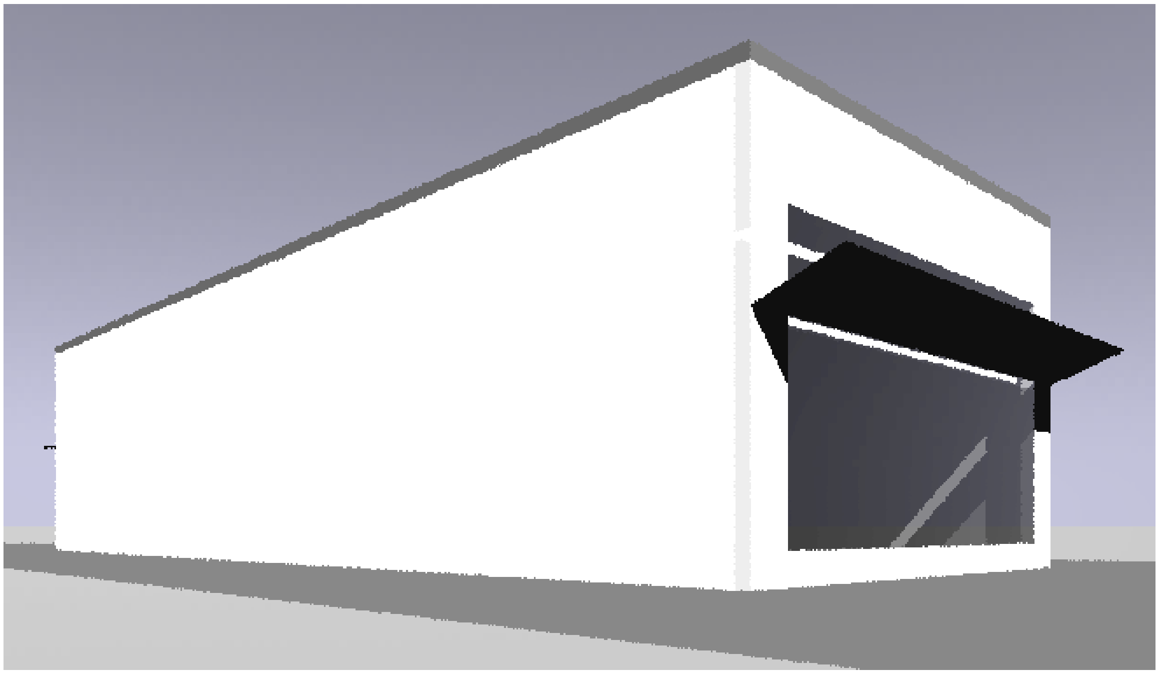


Figure 5.10. View of the west façade at 4 pm rendered using Radiance. At this time, the external light shelf is tilted upwards.

The width of the internal light shelf also changes depending on the position of the sun. When the sun is on the east, the width of the west light shelf is reduced to bring in more light through this window. At the same time, the width of the east light shelf is varied such that there is a balance between the objectives of avoiding direct sunlight and increasing daylight penetration. Usually the width of the east light shelf is near the higher end in order to block direct sunlight entering into the room. This is due to the constraint on glare that is checked during the evaluation of the objective function. The exact reason for the variation in the width of the internal light shelf is specific to the light patterns at the time of the day and is difficult to explain intuitively. Similarly when the sun is on the west in the afternoon, the width of internal light shelf on the west is increased to avoid excessive brightness. On the east it is reduced to increase light levels near the eastern window. A comparison of the light shelf configurations at hours 10:00 and 13:00 are shown in Figure 5.11 for illustration. It should be noted that diffused reflection (not shown here) contributes significantly to the lux levels in the room.

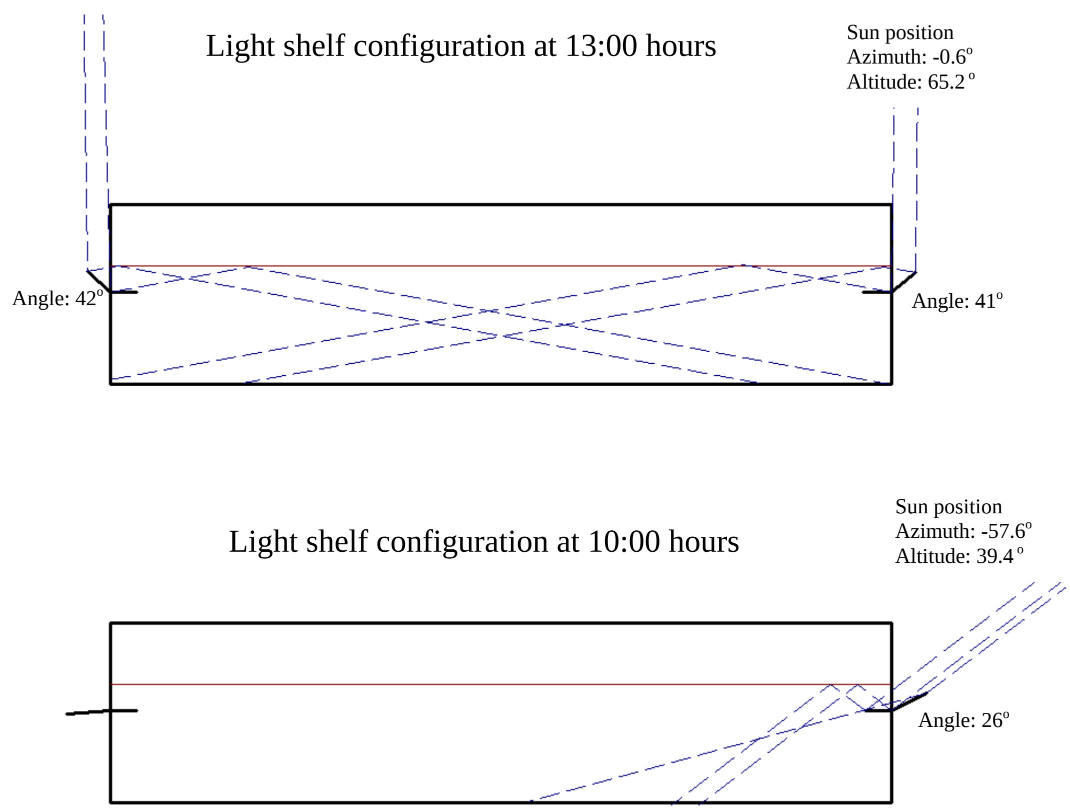


Figure 5.11. Comparison of light shelf configurations at hours 10:00 and 13:00. The dotted line indicates the path of a sunray directly reflected by the light shelf surface.

Hour	Angle of External light shelf		Width of internal light shelf	
	West	East	West	East
10	-5	50	0.63	0.99
11	-4	26	0.64	0.62
12	-2	26	0.61	0.61
13	42	41	0.68	0.65
14	44	4	0.71	0.64
15	38	-10	0.64	0.66
16	21	-1	0.99	0.74

Table 5.2. Optimal values of light shelf parameters for Configuration 3.

The small negative value for the angle of external light shelves when the sun is on the opposite side is a trend that is observed consistently. However, it is noted that the increase in lux levels due to the small tilt is almost insignificant compared to a perfectly horizontal light shelf. This small increase is due to diffused reflections from the sky and the ground and is dependent on the luminance distribution model of the sky used in the simulation software. The decrease in the net energy consumption due to the negative angle might not be significant; the optimization nevertheless converges to the global minimum point.

Overall, there is an increase in the day lighting level in the interior of the room. For example, at 14:00 hours the average lux value is 306, where as in the case of static light shelf it was 260.

A sample of random solutions generated during the optimization for 11 am is shown in Figure 5.12. Only one optimization variable, angle of east light shelf, is shown here. The minimum value of the objective function is when the angle of east light shelf is equal to 26o.

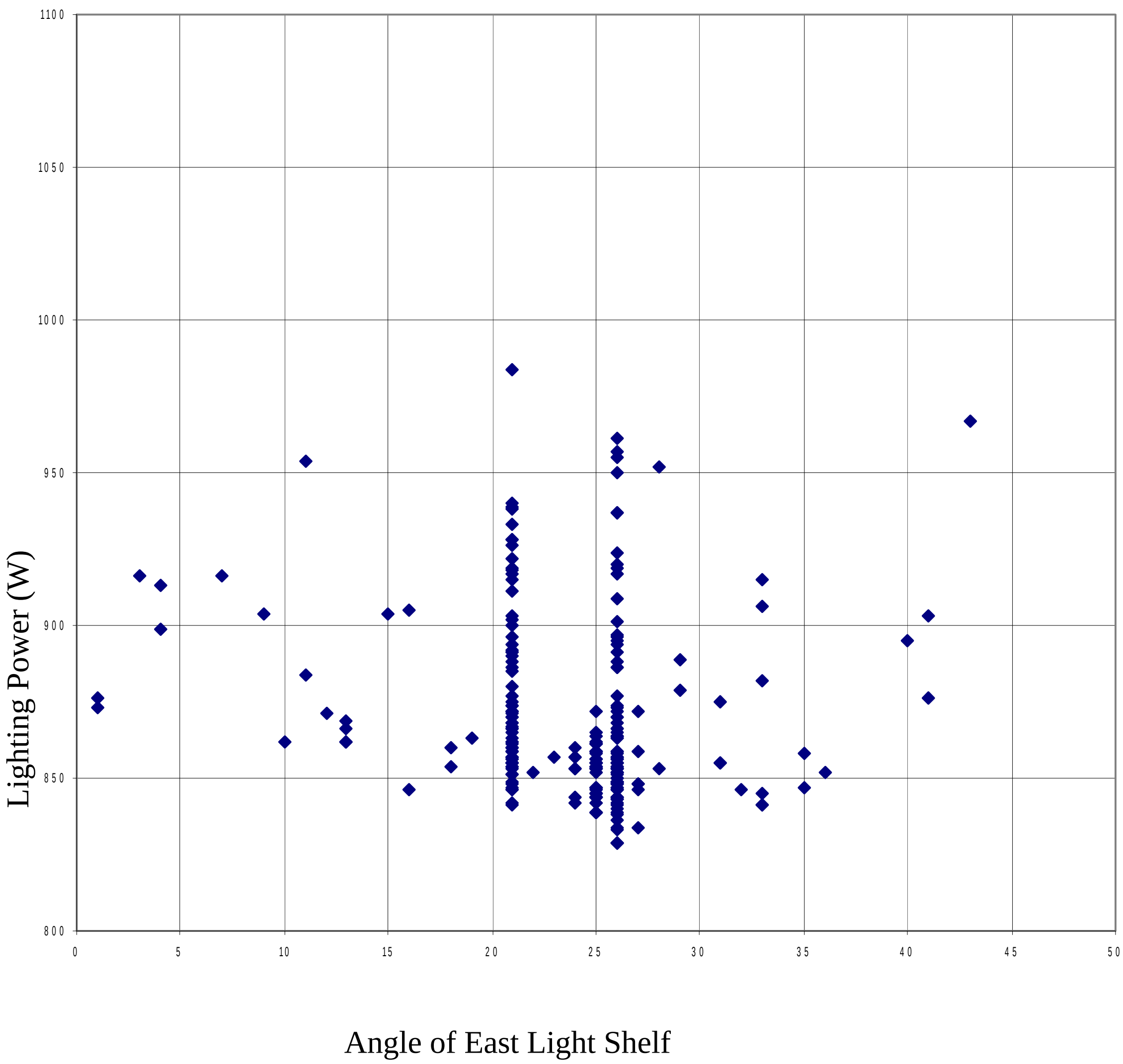


Figure 5.12. A random sample of solution points generated during the optimization for the light shelf configuration-3 at 11 am. Only one optimization variable (angle of east light shelf) is shown.

Comparison between the three cases

In order to evaluate the advantages of the adaptable light shelf configurations, the power required for artificial lighting at different hours of the day are summarized in Table 5.3. The total energy required for lighting during the simulated period is 6.44 KWhr for the static light shelf. This is reduced to 6.05 for the second configuration and 5.67 for the third configuration. The adaptable light shelf design with rotatable external part and moveable internal part produces a net energy savings of 12% compared to static light shelves. This is significant savings for large office buildings. With the adaptable light shelf, the lighting power density reduces to 7.5 W/m2 from 8.5 W/m2 for a static light shelf.

Hour	Power for Artificial Lighting (W)		
	Static Light Shelf	Configuration 2	Configuration 3
10	961	888	858
11	955	910	829
12	913	872	809
13	834	747	686
14	881	848	801
15	953	901	829
16	946	886	859

Table 5.3. Comparison between light shelf configurations

5.5 Concluding remarks

A light shelf configuration whose geometry can be adapted to environmental conditions is proposed in this chapter. The external part of this light shelf can be rotated and the internal part can be moved horizontally. The advantage of the system is evaluated by computing the energy required for artificial lighting. Using an example of hypothetical office building under Singapore weather conditions, it is found that about 12% savings is possible compared to a traditional static light shelf. This savings was possible because the angle of the external light shelf was adjusted to the position of the sun causing maximum daylight penetration into the interior of the room. The width of the internal light shelf was also varied such that direct sunlight was blocked when the sun is on the same side as the window and maximum daylight was permitted when the sun is on the opposite side.

The algorithm is general and can be used for buildings of different orientation and locations. The position of the sun is calculated automatically from the latitude and longitude specified by the user. The orientation of the building is included in the geometric model of the building. The optimization algorithm computes the best configuration of the light shelf for the current position of the sun and sky conditions. For a given building, the parameters of the shelves are predetermined based on month, time and sky conditions. This is a great advantage since the control system simply needs to adjust the geometry based on pre-computed values. Currently, only two sky conditions are used, namely, overcast and sunny. Better results are obtained by considering more variations in sky conditions.

Even though, control of window blinds and other shading devices have been explored before, the use of adaptable light shelves have never been studied. This study demonstrates the potential for savings in energy through such systems and through building automation and control, in general.

Future work involves verifying the predicted energy savings using actual measurements in a real building. Static light shelves have been installed in an office building in Singapore. Measurements taken on this system will help validate the predictions made by the simulations at least for one configuration under different sky conditions. Long term performance of the light shelves will also be part of this study. Evaluation of different types of mechanical systems for adapting the light shelf geometry is being explored. Comparison of horizontally moveable internal light shelf with rotatable internal light shelf (restricted to upward rotation only) is currently being carried out.

References

Adeli, H. and Cheng, N.-T. (1994a), "Augmented Lagrangian Genetic Algorithm for Structural Optimization", *Journal of Aerospace Engineering,* ASCE, Vol. 7, No. 1, pp. 104-118.

Adeli, H. and Cheng, N.-T. (1994b), "Concurrent Genetic Algorithms for Optimization of Large Structures", *Journal of Aerospace Engineering,* ASCE, Vol. 7, No. 3, pp. 276-296.

Aghemo C., Pellegrino A., LoVerso V.R.M. (2008), The approach to daylighting by scale models and sun and sky simulators: A case study for different shading systems, *Building and Environment,* 43, pp. 917–927.

Aizlewood M.E. (1993), Innovative daylighting systems: an experimental evaluation, *The International Journal of Lighting Research and Technology,* 25, pp. 141–52.

Bäck, T., Schwefel, H.P. (1993), An overview of evolutionary algorithms for parameter optimization, *Evolutionary Computation,* 1 (1), pp. 1-23.

Claros S., Solera A. (2002), Indoor daylight climate–influence of light shelf and model reflectance on light shelf performance in Madrid for hours with unit sunshine fraction, *Building and Environment,* 37, pp. 587–598.

Domer B., Raphael B., Shea K. and Smith I.F.C. (2003), "A study of two stochastic search methods for structural control", *Journal of Computing in Civil Engineering,* Vol 17, No 3, pp. 132-141.

Freewan A.A., Shao L., Riffat S. (2008), Optimizing performance of the light shelf by modifying ceiling geometry in highly luminous climates, *Solar Energy*, 82, 343–353.

Greenup, P.J., Edmonds, I.R. (2004), Test room measurements and computer simulations of the micro-light guiding shade daylight redirecting device. *Solar Energy,* 76, pp. 99–109.

Jiang, X. and Adeli, H. (2008), "Neuro-Genetic Algorithm for Nonlinear Active Control of Highrise Buildings, *International Journal for Numerical Methods in Engineering,* Vol. 75, No. 8, pp. 770-786.

Kristl, Z., Košǐr, M., Trobec Lah, M., Krainer A. (2008), Fuzzy control system for thermal and visual comfort in building, *Renewable Energy,* 33 (4), pp. 694-702.

Leslie R.P. (2003), Capturing the daylight dividend in buildings: why and how?, *Building and Environment,* 38, pp. 381–385.

Littlefair, P.J., Aizlewood, M.E., Birtles, A.B. (1994), The performance of innovative daylighting systems, *Renewable Energy,* 5, pp. 920–934.

Mardaljevic, J. (2004), Verification of program accuracy for illuminance modelling: assumptions, methodology and an examination of conflicting findings, *Lighting Research and Technology,* 36, pp. 217–242.

Ochoa C.E., Capeluto I.G. (2006), Evaluating visual comfort and performance of three natural lighting systems for deep office buildings in highly luminous climates, *Building and Environment,* 41, pp. 1128–1135.

Oteiza P, Soler A. (1995), A comparison of the daylighting performanceof different shading devices giving the same solar protection, *Architectural Science Review,* 38, pp. 171–176

Patil A., Raphael B., Rastogi P. (2004), Generalized phase-shifting interferometry using direct stochastic algorithm for global search, *Optics Letters*, 29, pp. 1381-1383.

Raphael B. and Smith I.F.C. (2003a), Fundamentals of computer aided engineering, John Wiley, UK.

Raphael B. and Smith I.F.C. (2003b), A direct stochastic algorithm for global search, *J of Applied Mathematics and Computation*, Vol 146, No 2-3, pp. 729-758.

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

Sarajii R.M., Mistrich R.G. (1993), The development of coefficients of utilization for light shelves. *Journal of the Illuminating Engineering Society*, pp. 139–161.

Sarma, K. and Adeli, H. (2000a), "Fuzzy Genetic Algorithm for Optimization of Steel Structures", *Journal of Structural Engineering*, ASCE, Vol. 126, No. 5, pp. 596-604.

Sarma, K. and Adeli, H. (2000b), "Fuzzy Discrete Multicriteria Cost Optimization of Steel Structures", *Journal of Structural Engineering*, ASCE, Vol. 126, No. 11, pp. 1339-1347.

Sarma, K.C. and Adeli, H. (2001), "Bi-Level Parallel Genetic Algorithms for Optimization of Large Steel Structures", *Computer-Aided Civil and Infrastructure Engineering*, 16 (5), pp. 295-304.

Sarma, K.C. and Adeli, H. (2002), "Life-Cycle Cost Optimization of Steel Structures", *International Journal for Numerical Methods in Engineering*, Vol. 55, No. 12, pp. 1451-1462.

Soler A., Oteiza P. (1997), Light shelf performance in Madrid, *Building and Environment*, 32, pp. 87–93.

Tsangrassoulis, A., Santamouris, M., Asimakopoulos, D. (1996), Theoretical and experimental analysis of daylight performance for various shading systems. *Energy and Buildings*, 24, pp. 223–230.

Tzempelikos, A., Athienitis, A.K. (2007), The impact of shading design and control on building cooling and lighting demand, *Solar Energy*, 81 (3), pp. 369-382.

Ward L. G, Shakespeare R. (1998), Rendering with Radiance: the art and science of lighting visualization, San Francisco: Morgan Kaufmann.

Holland J. (1975), Adaptation in natural artificial systems, University of Michigan Press.

6 Prototype implementation: Control of Personalized Ventilation [8]

Summary

This chapter presents the implementation of an integrated control strategy for personalized ventilation (PV). A full-scale control system involving 16 PV-ATDs (air terminal devices), 16 damper actuators, 16 air velocity sensors and 4 occupancy sensors was installed in the Field Environment Chamber of the Department of Building, National University of Singapore. An automatic control strategy was developed in which the air flow rates preferred by users are maintained constant, irrespective of adjustments to other PV-ATDs. The control algorithm integrates user preferences with other constraints such as maintaining CO2 levels below recommended values.

A series of experiments involving human subjects were conducted in order to evaluate the system. Altogether 10 experimental sessions of two hour durations were conducted. Each session involved about 36 subjects. Different combinations of ambient temperature, PV air temperature and control strategy were tested in these sessions. Two feedback forms were collected in each session. The feedback questionnaire contained questions related to thermal comfort, air quality and other indoor environmental parameters along with the performance of the control system.

Results show that the general acceptance score is high. The most significant finding of the study is that when users are provided individual control over the air flow rate of personalized ventilation, the ambient temperature and PV air temperature do not significantly influence perceived thermal comfort. Users seem to accept warmer temperatures such as 26 deg C. This is significant because there are possibilities for saving energy when the supply air temperature is raised.

Even though, the version of automatic control algorithm tested in the study did not significantly improve the user feedback, it nevertheless maintained reasonably high user acceptance. Some of the negative feedbacks were due to the slow response of the system to the changes made by the user. The automatic control algorithm was adapted after obtaining feedback from the users.

The prototype implementation demonstrates the concept of integrated control. Instead of the conventional strategy of controlling one actuator using one sensor, it is demonstrated that a control algorithm that incorporates multiple aspects such as user preferences, occupancy and air quality is feasible.

[8] The part of this chapter related to the subject study experiments is the work of the Graduate Student Researcher, ChenYixing. For more details, refer to Chen (2010).

6.1 Introduction

Personalized ventilation (PV) (Melikov, 2004; Sekhar et al., 2005; Yang and Sekhar, 2008) is a relatively new concept in air distribution and is yet to enjoy wide spread adoption in office buildings. PV air terminal devices (PV ATD) that are installed on desk tops supply treated fresh air closer to the breathing zone. Figures 6.1a and b show two models of PV-ATDs that have been installed in the Field Environmental Chambers (FEC) of the National University of Singapore, Department of Building.

Personalized Ventilation has the potential to maintain a healthier indoor environment. Unlike conventional ceiling supply, there is less mixing of fresh air with existing contaminated air before it is inhaled by occupants. Human subject studies show that PV has the potential to improve occupants' perceived air quality, thermal comfort (Sekhar et al., 2005; Conceicao et al., 2010; Kaczmarczyk et al., 2004), and decrease the occurrence of effects such as the sick building syndrome (SBS) symptom (Kaczmarczyk et al., 2004).

In addition to health benefits, personalized ventilation is also attractive from the point of view of energy consumption. It has the potential to reduce cooling energy requirements since a higher ambient temperature might be maintained (Sekhar et al., 2005). The energy performance of PV systems has been studied before. Schiavon et al. (2008, 2009) have shown by means of energy simulation that PV has the potential for energy savings in both cold climate and tropics when proper control strategies are applied. However, a problem with these studies is that aspects related to user comfort and acceptance depend to a large extend on the characteristics of the PV ATDs, especially, the flexibility to adjust parameters such as air velocity, direction flow, position related to occupants, etc. Due to these effects, results from studies that are conducted using a particular type of PV ATDs cannot easily be reused for other types.

a. PV ATD fixed on desk

b. Moveable PV ATD

Figure 6.1 Personalised Ventilation Air terminal Devices (PV-ATDs).

A topic that has been largely ignored by previous researchers is the role of the control system in user comfort when PV systems are used. Large differences exist between occupants with respect to physiological and psychological response, clothing insulation, activity, air temperature, and air movement preferences (Melikov, 2004). Therefore, environmental conditions acceptable for most occupants in rooms may only be achieved by providing each occupant with the possibility to generate and control his/her own preferred microenvironment. Thus the control strategy plays an important role in user comfort.

This chapter presents two control strategies for personalized ventilation namely, manual and automatic. The first one involves manual operation of dampers using a software interface. The second involves automatic control in which the air flow rate chosen by the user is maintained at a constant level through closed loop control.

The automatic control strategy has the potential to eliminate some problems associated with manual control such as fluctuations in air flow rate, and non-linear response of dampers and is expected to exhibit more intelligent behaviour. Autonomous behavior is considered to be an essential characteristic of intelligent systems (Raphael and Smith, 2003). The term autonomy refers to a system’s ability to take decisions by itself for progressing towards a well-defined goal. Users are increasingly expecting such intelligent behavior from building systems (Wong and Li, 2006) and do not like to perform frequent manual interventions. For example, people expect electrical lights to be switched off automatically when sufficient daylight is available in commercial and office buildings. However, several studies have indicated that people tend to reject automatic systems when the control algorithm does not respect their personal preferences (Guillemin and Morel, 2001). Incorporating and integrating user preferences in an automatic control algorithm is a complex task and has not been fully addressed in this research. The ideas presented in this chapter are only small steps towards developing a control system framework that accommodates user preferences. Further work is needed to establish a control system that is fully acceptable to a majority of users.

6.2 Incorporating user preferences in automatic control

In general, control is a multi-objective optimization problem in which different objectives might conflict with each other and trade-offs need to be carefully evaluated. For example, increasing the supply of treated fresh air is good for indoor air quality, but not attractive with respect to minimizing energy consumption. In some applications, such conflicts between objectives are resolved by combining multiple objectives using weight factors. In some other applications, objectives are transformed into constraints by limiting the values of parameters based on codes and recommendations of professional bodies. However, these approaches are not effective when user preferences need to be incorporated in the model. This is because it is not easy to quantify user preferences. Because of this reason, the multi-objective framework presented in Chapter 2 cannot be easily adopted for the control of personalized ventilation. Furthermore, energy and other parameters related to personalized ventilation cannot be reliably predicted with the current state of research in this area. Therefore, accommodating user preferences is given the primary focus in this study.

In the traditional control of VAV (Variable Air Volume) systems, user preferences are captured in the form of set points of parameters such as the temperature. The control of PV-ATD faces some difficulties in this respect since ordinary users have no idea about the range of air velocity values and are not familiar with its units. Users are not able to specify what values are acceptable to them in terms of absolute numbers. Therefore, user preferences have to be captured indirectly.

As far as users are concerned, the two most important parameters related to PV-ATD are air temperature and velocity. Since air is supplied to all the PV-ATDs from the same AHU, it is not easy to provide personalized control over air temperature. Air velocity can be adjusted by opening or closing the dampers. However, users may not be able to specify precisely the air velocity that they are most comfortable with. Therefore in this study, users are allowed to choose their preferred air velocity by trial and error. This requires the development of a convenient human-machine interface that encourages users to experiment and explore. Finally, the air flow rate chosen by the user is translated into an appropriate damper setting.

Automatic control attempts to solve several technical difficulties associated with the manual control of personalized ventilation. Firstly, since a group of PV ATDs receive air supply from a common ventilation duct, the air flow rate changes when other users open or close their diffusers. Users have to frequently adjust the dampers in order to maintain their preferred air flow. Automatic control might be used to solve this problem. In automatic control, the control objective is chosen such that a constant air flow rate through each PV-ATD is maintained according to the preference of the user.

Secondly, the input-output relationship is non-linear. Figure 6.2 shows the relationship between the air velocity through the diffuser and the damper opening for a constant air pressure at the air handling unit (AHU). The relationship is quite complex. In the beginning, a small opening causes a large increase in the air velocity. After a certain stage, the change in air velocity is not appreciable because the resistance to the flow is negligible. After about 30% opening, the curve is nearly flat and the slope is close to zero. The flat portion of the curve is problematic for traditional closed loop feedback control systems since the direction of movement cannot be determined using small changes in the actuator states. Most intriguing is the drop in velocity when the damper is opened above 50%, even though this can be explained using fluid mechanics theory.

Even for experts in the domain, the input-output relationship shown in Figure 6.2 is not intuitive. Ordinary users may never expect that the air velocity might decrease when the damper is opened beyond a certain level. The initial sharp increase in velocity followed by a relatively flat region might also create problems for users in adjusting the velocity to their liking. The user might want to increase the air velocity and by intuition, might open the damper, only to find that the air velocity has decreased. This could be quite confusing. Therefore, the automatic control strategy was designed such that users obtain a consistent performance from the system according to their expectations. Users expect a monotonic behaviour, that is, when the damper is opened there should be an increase in air velocity and it should be linearly proportional to the amount of opening. In the automatic control, the user action of opening the damper is interpreted as a request to increase the air velocity. The correct position of the damper corresponding to the proportional increase is calculated by global search.

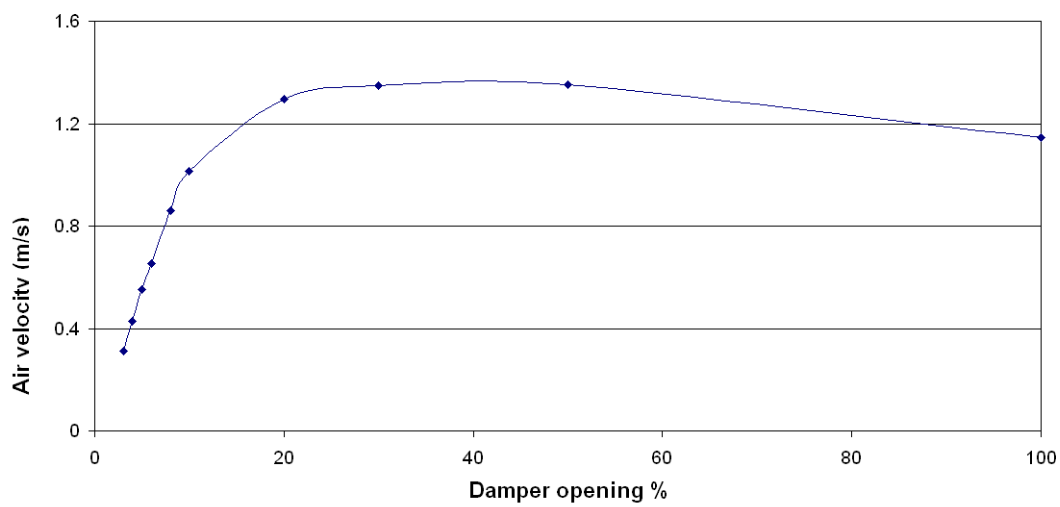


Figure 6.2 Damper characteristics. The input-output relationship is non-linear

6.3 The automatic control algorithm

Traditional HVAC control systems use simple algorithms such as PID (Proportional, Integral Derivative control) for maintaining set points. In such systems, the set points are defined by the user and the actual control parameters are directly measured by sensors. It is also assumed that the direction in which actuators should be moved in order to attain the goal is known. For example, if the temperature is below the set point, the damper actuators have to be opened in order to bring in more cold air. In such systems the control task is simpler because small temporary changes are not easily detected by the users and thermal sensation is affected only by reasonably long exposure. However, PV-ATDs are installed closer to the occupant's face and airflow fluctuations are easily detected by users. Also, the relationship between damper opening and air velocity need not be monotonic as explained in Section 6.2.

The curve shown in Figure 6.2 was plotted by opening only one damper, keeping the dampers of all other PV-ATDs unchanged. The characteristics of the curve change when other dampers are opened or the air pressure at the AHU is changed. In a real system, the curve is not smooth and contains several peaks and troughs because of these interactions.

In general, it is not possible to predict the airflow without performing a complex CFD simulation in which the entire system is accurately modelled. Since it is not practical to perform CFD simulation within the control loop, empirical models that predict the input-output relationship is used. Piecewise linear relationships are used in these models which are learnt and calibrated using real time data during the process of adjusting the dampers.

As explained in the previous section, in automatic control, the main control objective is to maintain a steady air velocity through the PV-ATDs according to the preference of the user. The user preference is captured by examining the direction in which users adjust the damper openings and the relative magnitude of changes made by them. The absolute values of air flow or damper openings are not used. The assumption is that users are not familiar with the units and they are only able to indicate their relative preference. Once the air velocity preferred by the user is captured, the control algorithm ensures that the same value is maintained until the user changes it again.

The set points of air velocity are calculated indirectly from user actions using the following procedure: When the user requests for a change in the damper opening it is checked whether the new value is greater than the previous value. In that case, it is assumed that the user wants to increase the air velocity. The new air velocity set point is calculated by adding an increment proportional to the magnitude of increase requested by the user. Similarly, if the user requests for a decrease in damper opening, the new air velocity set point is calculated by applying a proportional decrement to currently measured value.

For the computed new set point, a first estimate for the required damper opening is calculated through interpolation using the calibrated input-output relationship. This might result in a damper movement which is opposite in direction to what is requested by the user, depending on the nature of the curve. Fine adjustments to the damper position are made in subsequent iterations of the feedback control loop.

6.3.1 Constraints

In addition to the objective of minimizing the difference between the set point and measured values, the control algorithm should also satisfy a set of constraints. Constraints are grouped into two categories, hard and soft. Hard constraints have to be satisfied with high priority. Soft constraints have lesser precedence and might be violated in order to satisfy hard constraints.

The main constraint that is chosen in this application is that the CO2 level in the room should be less than prescribed limits. This is a hard constraint that takes high precedence. Solutions that satisfy this constraint are always selected instead of the ones that violate this constraint. If many users set the air flow at very low levels and the CO2 level exceeds prescribed limits, the control system might open some of the dampers disregarding user’s preferred air flow, as this will cause the CO2 level to drop.

Another constraint that is incorporated is the rule that if there is no occupant at a desk, the damper for that PV-ATD should be closed. This is a soft constraint incorporated for reducing energy consumption. However, this constraint has less importance and if the CO2 level rises, the first option that is tried out is to open the dampers of PV-ATDs at stations where users are not present.

The presence or absence of occupants can be detected using Passive Infra-red (PIR) occupancy sensors. However, due to some practical difficulties these sensors were not used in the present study. The main problem was the large field of view of the sensors that were used in this project. Since the desks were not separated into cubicles, it was difficult to restrict the field of view of the sensors to a single desk. Instead, a button was provided in the software interface to indicate that the user is leaving the desk.

6.3.2 Implementation details

A customized search algorithm was used in this application, instead of general purpose algorithms such as PGSL (Raphael and Smith, 2003). This is because it is possible to simplify this control task into a simpler optimization problem in a single variable. The control algorithm involving the custom search procedure is summarized in the following steps:

Step 1: Read the system configuration. Start the control loop,
Step 2: Read the sensors
Step 3: Check the constraints using current sensor values. If hard constraints are violated, identify the best solution that does not violate any hard constraints. Apply the control action and continue with the control loop from Step 2.
Step 4: Repeat for each PV station. Check if there is any change in the occupancy or user preference at that station. If yes, re-compute the optimal damper position for the new setting, using simple grid search in one dimension.
Step 5: Repeat for each PV station. Check if a specified interval of time has elapsed after the last change in the damper opening and there is a significant change in the air velocity at the PV station. If yes, update the calibration model and re-compute the optimal damper position.
Step 6: Repeat from Step 2 after a small time delay

6.4 User acceptance of control strategies

This section describes the experiments that were conducted in order to evaluate user acceptance of different control strategies. Two different strategies for PV control were tested, namely, manual and automatic.

In manual control mode, users interact with the software interface and specify the percentage of opening of the dampers. They increase or decrease the opening until they are satisfied with the air velocity.

In automatic control mode, the algorithm presented in Section 6.3 is used. The input variable for the control algorithm is the damper opening and the output variable is the air velocity which is measured using air velocity sensors installed in the duct leading to the PV-ATDs. The control task involves determining the damper opening required for achieving a specific air velocity through the PV-ATD.

Altogether 10 experimental sessions of two hour durations were conducted. Each session involved about 36 subjects. Different combinations of ambient temperature, PV air temperature and control strategy were tested in these sessions.

6.4.1 Experimental Setup

The experiments were conducted in a Field Environment Chamber (FEC) of size 10m(L) × 7.5m(W) × 2.6m(H), an experimental facility at the National University of Singapore, which is equipped with different types of air distribution systems. In this study, the combination of conventional ceiling supply and PV is used. Users are able to vary the air flow rate through PV-ATDs at individual workstations using a software interface. The air flow is regulated by opening or closing dampers installed in the supply ducts. In addition, users are able to change the direction of flow by turning the diffusers horizontally and vertically. The height and position of the PV ATDs cannot be adjusted in this implementation.

Fresh air for PV and mixing ventilation for ambient cooling are served by separate air handling units in the FEC. Both systems are linked to a common Building Management System (BMS). All the parameters required for computing energy consumption on the air side were recorded through the BMS. This includes air flow rate, fan speed, pressure, supply and return air temperatures, chilled water supply and return temperatures, etc.

6.4.1.1 Hardware and software

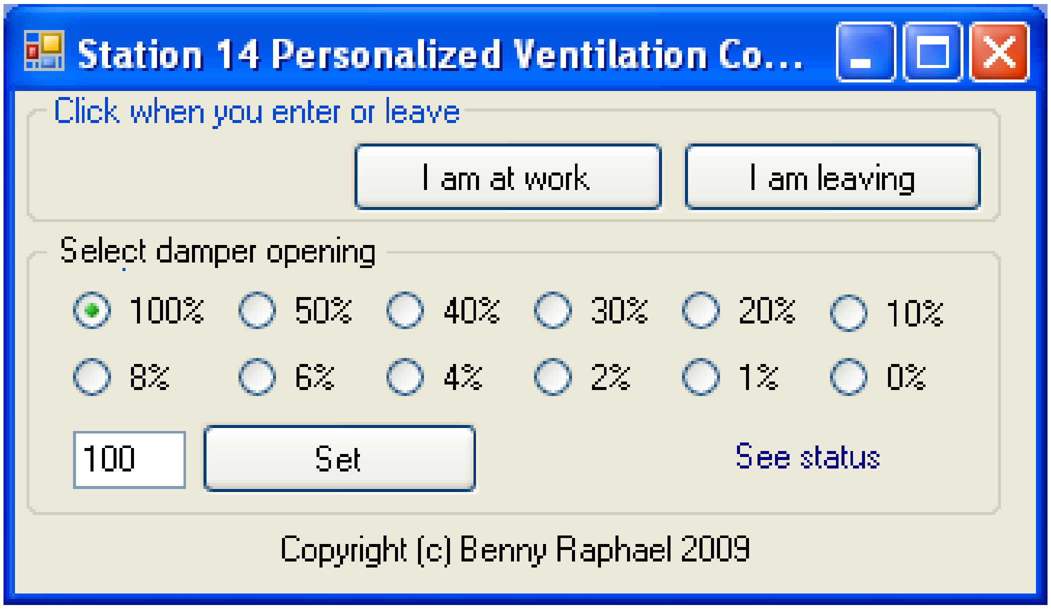
A full-scale control system involving 16 PV-ATDs (air terminal devices), 16 damper actuators, 16 air velocity sensors and 4 occupancy sensors was installed in the FEC. Out of 16 PV-ATDs, only 13 were used in the study since the remaining workstations were not in a usable state.

The model of the PV-ATDs used in the study is shown in Figure 6.1a. These are installed on the desks in the FEC. Initially, these PV-ATDs did not offer any means for regulating the air flow at the workstations. The air flow could be changed only through the use of manually operated dampers installed on the supply ducts as shown in Figure 6.3.

In order to implement automatic control, the manually operated dampers were replaced by motorized dampers as shown in Figure 6.4. These motors take commands in the form of analog DC voltage in the range 0-5 V. The motor rotates proportional to the applied voltage causing the valve to open or close.

6.3 Manual damper to regulate airflow

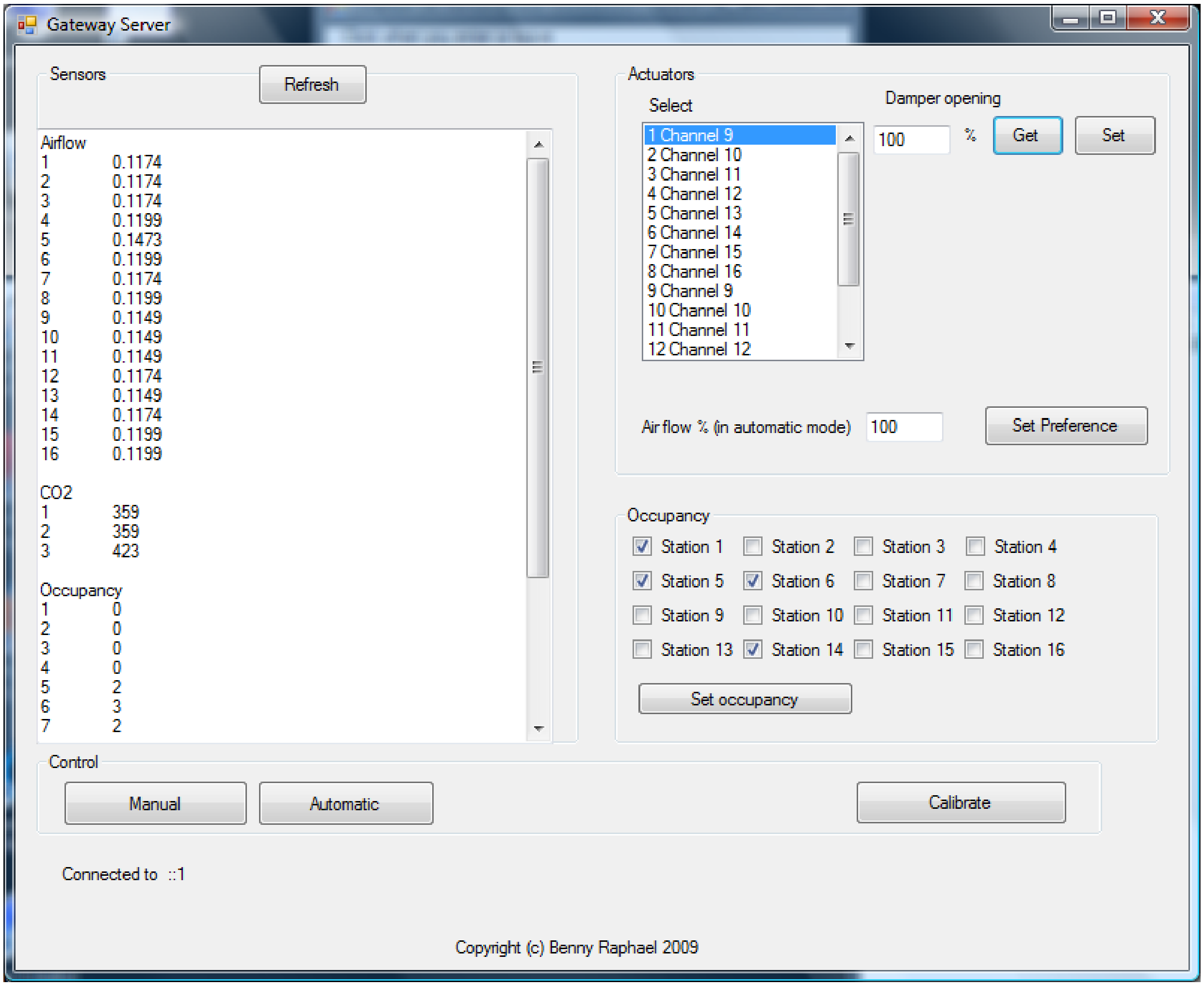
6.4 Motorised damper to regulate airflow



6.5 User interface to regulate airflow

If mechanical controls are provided for regulating the openings of PV-ATDs, additional electronic devices are needed for capturing the preferred settings of users for analysis and other purposes. To avoid this problem, a software interface was provided for regulating the airflow. It is assumed that users working in offices have their computers turned on all the time and it is easier for them to adjust the setting through the software. This approach helps in easily recording any changes made by users and adapting the control strategy according to user preferences. A screen shot of the user interface for regulating the air flow is shown in Figure 6.4. This program is installed on all the computers in the FEC. When the user changes his preferences through this interface, the program communicates with a server using TCP/IP through the local area network (LAN). The server is part of the BACNET network and is connected to the TCP/IP LAN through a router. Upon receiving requests from the client, the server communicates with BACNET controllers which in turn send the analogue DC command signals to operate the damper motors.

The server program has an interface for system administrators. The control strategy (whether manual or automatic) can be selected through this interface. It is also possible to monitor the sensors and operate the actuators through this interface. A screenshot of the interface is shown in Figure 6.6.



6.6 User interface of the server

6.4.1.2 Design of the subject study

Experiments involving human subjects were conducted in the FEC where the environmental parameters such as temperature and humidity are maintained at constant values during a two hour period. Subjects were asked to perform routine tasks in a standard office environment and their feedback is collected. Three groups of about 13 subjects at a time participated in each session. In each session, a certain combination of ambient temperature and PV temperature is maintained. The experimental conditions comprised the combinations of two ambient air temperatures, three personalized air temperatures and two control strategies. All other indoor environment parameters were kept nearly constant. These conditions are listed in Table 6.1. Around 45 subjects took part in the study.

Session	Control	Temperature ^{o}C	
		Ambient	PV
1	Manual	23	20
2	Manual	23	23
3	Manual	26	26
4	Manual	26	23
5	Manual	26	20
6	Automatic	26	20
7	Automatic	26	23
8	Automatic	26	26
9	Automatic	23	23
10	Automatic	23	20

Table 6.1 Experimental conditions

Even though, the original aim was to keep the PV supply air temperature at three discrete values, namely, 20, 23 and 26 deg C, this was not possible for practical reasons. The PV supply air mixes with ambient air and hence the temperature changes within less than a centimeter from the diffuser outlet. In fact, temperature measurements taken using a probe inserted into the holes of the diffuser show that temperature starts to change significantly even before air is released outside the diffuser. The effect is more pronounced at low air velocities. As a result of this, the PV supply air temperature deviated by as much as 1.5 deg C relative to targeted values.

Two sets of feedback are collected in each session, the first after thirty minutes and the second one at the end of the session. The feedback questionnaire is meant to collect information related to user comfort as well as their impression of the personalized ventilation concept. The questionnaire contained questions related to thermal comfort, air quality and other indoor environmental parameters along with the performance of the control system. An acceptability scale (Figure 6.7a) was used to gauge the acceptability of air quality, thermal sensation and air movement. On the 7-point ASHRAE scale (Figure 6.7b) subjects rated their thermal sensation separately for each part of the body and for the whole body. The questionnaire contained 86 questions in which users either had to select options from multiple choices or select a value from a numerical scale.

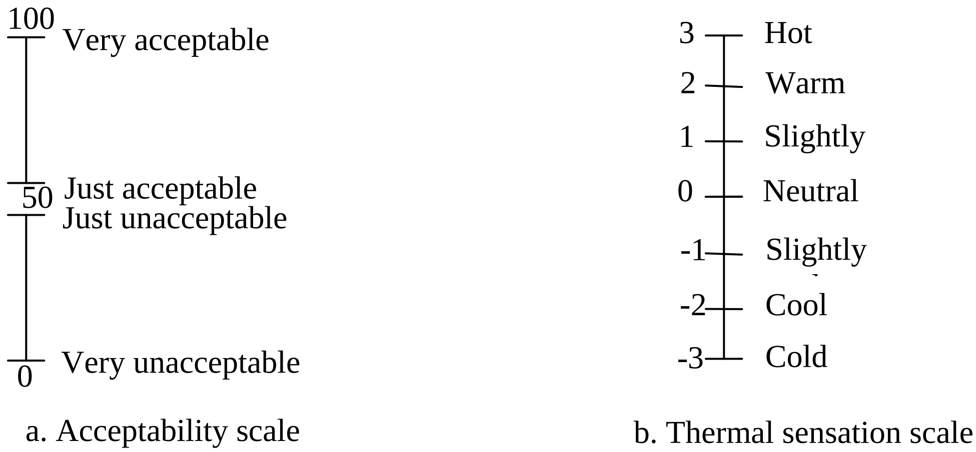


Fig. 6.7 Scales used in subjective evaluation

Manual control was used in five sessions and automatic control in the remaining. In manual control mode, users interact with the software interface to specify the percentage of opening of the dampers. They increase or decrease the opening until they are satisfied with the air flow. The disadvantage of this mode is that the flow might change when other users adjust their dampers since the air is supplied to all the diffusers through a common duct. Users might have to adjust the damper openings from time to time in order to keep the air flow at preferred levels. In the automatic mode, the control objective is to maintain a steady air velocity through the PV-ATDs irrespective of the operations of other users. Users specify the percentage of air flow rate instead of opening of the damper through the same software interface. An Airflow sensor is installed in each duct, which takes about one minute to record a stable air flow after a change in the damper opening.

Users were not informed about the type of control used in each session. They used the same software interface to adjust the air flow rate in both control modes. Changes in the air flow rate initiated by users were recorded by the control software. By analyzing these records, the air flow rates preferred by users under different temperature conditions are obtained. This information is evaluated in conjunction with subjective feedback on perceived air quality.

Altogether 736 user feedback forms were collected in all the experimental sessions amounting to a total of 736 man hours of user evaluation.

6.4.2 Analysis of user feedback

6.4.2.1 User comfort

Only a few important questions related to thermal sensation and air movement are discussed here. These questions are summarized in Table 6.2. The average score for these questions are summarized in Table 6.3. More detailed analysis can be found in Chen (2010).

ID	**Question**	**Scale**
Q1	Acceptability of thermal comfort condition	See Figure 2a
Q2	Thermal sensation on face	See Figure 2b
Q3	Thermal sensation on neck	See Figure 2b
Q5	Acceptability of air movement on face	See Figure 2a

Table 6.2. Selected questions from the feedback questionnaire

Session	**Control**	**Temperature ^{oC}**		**Q1**	**Q2**	**Q3**	**Q4**	**Q5**
		Ambient	**PV**					
1	Manual	23	20	64.10	1.50	0.78	-0.71	59.46
2	Manual	23	23	64.31	1.17	0.69	-0.53	61.28
3	Manual	26	26	63.94	0.79	0.26	-0.48	64.08
4	Manual	26	23	66.20	1.05	0.46	-0.60	61.85
5	Manual	26	20	68.25	0.88	0.42	-0.74	62.04
6	Automatic	26	20	64.50	0.84	0.36	-0.53	63.11
7	Automatic	26	23	68.14	0.81	0.44	-0.50	64.74
8	Automatic	26	26	68.61	0.72	0.35	-0.34	69.73
9	Automatic	23	23	65.17	1.07	0.65	-0.47	64.82
10	Automatic	23	20	64.88	0.99	0.70	-0.45	66.15

Table 6.3. Summary of feedback score for selected questions

From Table 6.3, it can be observed that the acceptability of thermal comfort conditions (Q1) does not vary significantly with the ambient temperature or PV temperature. This result differs from similar past studies. A possible explanation for this is that in the current experiments, users have control over the airflow through the PV ATDs and any perceived thermal discomfort due to higher ambient temperature can be overcome by adjusting the air movement.

The thermal sensations on the face (Q2) and neck (Q3) are between neutral and slightly warm. These do not show significant variations with the temperature. The slightly lower value for Q2 and Q3 for the temperature combination (26, 26) (Sessions 3 and 8) is surprising. Again this might be the result of higher air flow at this temperature combination which has been confirmed from air flow measurements.

The acceptability of air movement on face (Q5) is fairly consistent for all temperature combinations. The score is slightly tilted towards the acceptable range, even though the acceptance level is not very high. This is related to the characteristics of the PV-ATDs used in the study. These were fixed on the desk at a height close to occupant's face. The height and position of the PV ATDs cannot be adjusted with this installation. A new design of moveable PV ATDs (Figure 6.1b) is currently being evaluated and is expected to overcome this problem.

All the temperature settings had a reasonably high level of user acceptance as indicated in the feedback form. In fact, there were no significant differences in the user acceptance score for any of the above settings. This is because any thermal discomfort is compensated by the ability to adjust the PV ATDs and obtain comfortable air movement.

In summary, results show that the ambient temperature and PV air temperature do not significantly influence perceived thermal comfort when users are provided individual control over the air flow rate of personalized ventilation. Users seem to accept warmer temperatures such as 26oC since any thermal discomfort might be overcome through increased air movement. Even though the mode of control was not found to have a significant influence on the user feedback, some issues related to control have been identified and have been addressed.

6.4.2.2 Performance of the PV system

Table 6.4 summarises the scores of three questions that evaluated the performance of the control system and the PV-ATDs.

Question	Scale	Score: manual control	Score: Automatic control
How easy is it to control the personalized ventilation air terminal device (PV ATD) ?	0: Very difficult 100: Very easy	81.5	79.3
How frequently did you have to adjust the PV ATD?	0: Very often 100: Seldom	72.8	68.9
How do you rate the performance of the PV ATD?	0: Bad 100: Good	74.8	73.2

Table 6.4. Summary of feedback related to the PV System

The differences in the scores for manual and automatic control modes are negligible. The general opinion is that it is easy to control the PV-ATD and frequent interventions are not necessary. This shows that a software interface for adjusting the flow in the PV-ATDs is accepted and conventional mechanical controls are not necessary. The high score for manual control shows that the fluctuations in air flow are not easily perceptible and are not critical.

Damper characteristics which are described in Section 6.2 also played a role in a less than expected acceptance score. Even for experts in the domain, the input-output relationship shown in Figure 6.2 is not intuitive. Normal users may never expect that the air velocity might decrease when the damper is opened beyond a certain level. The initial sharp increase in velocity followed by a relatively flat region makes it difficult for users in adjusting the velocity to their preference. The following comments by subjects reflect this observation:

- When I adjust the ventilation from 11% to 10%, the rate of airflow becomes almost negligible.
- I find that the PV ATD responds inconsistently to my adjustments. Sometimes it responds very quickly while other times very slowly and I did not know how strong the air flow would be when I first tried to change the settings.
- Change in opening does not make a big difference to air flow
- Difference between the different levels of damper opening is too vast. Makes it difficult to find a satisfactory level.

In the automatic control mode, some of the above problems are eliminated because the damper opening required for an air flow proportional to what users demanded was computed automatically. However, automatic control also had some problems which are reflected in the following comments after the first session in which automatic mode was adopted:

- It is not responding to my settings and seems to be operating on its own. It blows higher than I have set it to be.
- I noticed that the air movement varies, especially when others adjust their PV ATD.

The main problem in the automatic mode was the time taken for adjusting the damper to the required level. In the manual mode, the damper was instantaneously adjusted to the percentage opening specified by the user. In the automatic mode, this was done indirectly in a few iterations. This process took considerable amount of time because it takes more than one minute for the air flow to stabilize after a change in the damper opening. During this process, users felt that the system was behaving erratically. These issues have been addressed in the development of a new version of the control algorithm.

In spite of the above problems related to the control of air flow, there is overall level of acceptance of the PV system is reasonable which is evident from Table 6.4.

6.5 Concluding remarks

The experiments reported in this chapter differ from similar previous ones in terms of control options available to the user. Some earlier experiments that studied the performance of PV have been done in different conditions. In most studies, the environment was maintained constant and the objective was to assess the variation in subject response with respect to environmental parameters. For example, the combinations of ambient air temperatures, personalized air temperatures and PV airflow rate were kept constant and could not be controlled by users within an experimental session (Sekhar et al. 2005; Sun et al., 2007). Aspects related to thermal comfort were studied through questionnaire feedback. However, in each experiment conditions, the environment was maintained constant. Subjects did not have the control of the micro-environment by themselves. In the current set of experiments, subjects were allowed to select their preferred air flow rate and thus adjust their micro-environment.

The most significant finding of the study is that when users are provided individual control over the air flow rate of personalized ventilation, the ambient temperature and PV air temperature do not significantly influence perceived thermal comfort. Users seem to accept warmer temperatures such as 26 deg C. This is significant because there are possibilities for saving energy when the supply air temperature is raised.

Even though, the version of automatic control algorithm tested in the study did not significantly improve the user feedback, it nevertheless maintained reasonably high user acceptance. Some of the negative feedback were due to the slow response of the system to the changes made by the user. Other comments were related to the characteristics of the PV-ATDs and the damper. Inability to adjust the height and position of the PV-ATDs was found to be a major drawback. These issues are being addressed in the installation of a new PV system.

The main problem with the automatic system was found to be the relatively long time required to achieve desired output, during which users thought that the system was behaving erratically. The automatic control algorithm was adapted after obtaining feedback from the users. The main problems were the time taken for adjusting the damper to the required level and the errors in the calibrated model. In the manual mode, the damper was instantaneously adjusted to the percentage opening specified by the user. In the automatic mode, this was done indirectly. First, the air velocity set point was computed as explained in Section 6.3. Then the damper was adjusted in a sequence of steps to achieve this set point. This process took considerable amount of time because it takes more than one minute for the air flow to stabilize after a change in the damper opening. During this process, users felt that the system was behaving erratically. Based on the feedback, the control strategy was modified as follows:

When the user requests for a change in the damper opening, the new set point is computed as explained in Section 6.3. However, instead of employing an iterative procedure to attain the set point, the control action is applied in a single step using the initial estimate of damper opening obtained from the input-output relationship. This step may not produce exactly the same air velocity that is targeted. However, as long as the step is in a direction that is anticipated by the user, it is likely to be acceptable to them and they can make further adjustments if necessary. After a certain period of time when the user is not found to make any more changes, the actual air velocity is measured and it is used as the set point to be maintained. This modified strategy is expected to increase the user acceptance of the system.

The prototype implementation once again demonstrates the concept of integrated control. Instead of the conventional strategy of controlling one actuator using one sensor, it is demonstrated that control actions can be identified and applied through examining the global scenario. A damper is opened or closed not using the air velocity recorded by a single sensor, but by making use of all the information such as the average level of CO2 in the room, occupancy at other workstations, etc. If energy and IAQ parameters can be reliably predicted, multi-objective control strategies can be potentially applied to the control of personalized ventilation as explained in previous chapters.

References

Chen Y., (2010). Thermal Comfort And Energy Performance Of Personalized Ventilation With Individual Control of Air Flow Rate System In the Tropics, MSc thesis, Dept. of building, National University of Singapore.

Conceição, E.Z.E., Lúcio, Ma.M.J.R., Rosa, S.P., Custódio, A.L.V., Andrade, R.L., Meira M.J.P.A., 2010. Evaluation of comfort level in desks equipped with two personalized ventilation systems in slightly warm environments, Building and Environment 45 (3), pp. 601-609.

Grivel, F. and Candas, V., (1991). Ambient temperatures preferred by young European males and females at rest. Ergonomics, 34 (3), pp. 365-378.

Guillemin A., Morel N., (2001). An innovative lighting controller integrated in a self-adaptive building control system, Energy and Buildings, 33 477-487.

Kaczmarczyk, J., Melikov, A., Fanger, P.O. (2004). Human response to personalized ventilation and mixing ventilation. Indoor Air. 14(8): 17-29.

Melikov A.K., Personalized ventilation, (2004) Indoor Air, Supplement, 14 (SUPPL. 7), pp. 157-167.

Melikov A, Cermak R, Mayer M., (2002). Personalized ventilation: evaluation of different air terminal devices. Energy and Buildings 34(8):829–36.

Raphael B. and Smith I.F.C, (2003). Fundamentals of computer aided engineering, John Wiley, UK.

Schiavon S., Melikov A.K., (2008). Energy-saving strategies with personalized ventilation in cold climates, Energy and Buildings,

Schiavon S., Arsen K. Melikov, Sekhar C.S., (2009). Energy analysis of the personalized ventilation system in hot and humid climates, Energy and Buildings.

Sekhar, S.C., Gong, N., Tham, K.W., Cheong, K.W., Melikov, A.K., Wyon, D.P., Fanger, P.O., 2005. Findings of personalized ventilation studies in a hot and humid climate, (2005) HVAC and R Research, 11 (4), pp. 603-620.

Yang, B., Sekhar, S.C., 2008. The influence of evenly distributed ceiling mounted personalized ventilation devices on the indoor environment, International Journal of Ventilation, Volume 7, Issue 2, September 2008, Pages 99-112.

Sun W., Tham K.W., Zhou W., Gong N., (2007). Thermal performance of a personalized ventilation air terminal device at two different turbulence intensities. Building and environment 42 3974 – 3983.

Wong, J., Li H. (2006). Development of a conceptual model for the selection of intelligent building systems, Building and Environment 41 1106–1123.

7 Conclusions

7.1 Summary of research accomplishments

A framework and a methodology for the integrated control of IEQ parameters through multi-objective optimization were developed in this research. A new control strategy that maximizes building performance has been developed. The most important aspect of the strategy is the treatment of the task as a multi-criteria global optimization problem. The best control action is selected by the simultaneous evaluation of multiple objectives. This is a new development with respect to current industry practice. Control strategies that are commonly adopted in the industry apply local actions which affect a single parameter at a time. In this research, global search is employed for selecting solutions that achieve reasonable trade-offs among parameters when there are conflicting objectives. This approach makes it possible to include criteria such as energy consumption, user preferences and indoor environment quality in the control strategy.

Three new algorithms and their software implementations have been developed in this project. These are:

- MOPD – an algorithm for generating Pareto Front for multi-objective optimization
- RR-PARETO – an algorithm for selecting the best solution from a Pareto Front that achieves reasonable trade-offs between conflicting objectives
- NUS-CBRFIT – a machine learning algorithm using case based reasoning and statistical learning

In addition to the above modules, an interface to lighting and energy simulation software has been developed. This module called NUS-ILES permits both simulations to be done using a single building model expressed in the XML format.

The methodology as well as individual components have been tested and evaluated using several test cases involving both full scale working systems and hypothetical examples. Three control applications have been developed using this methodology. The first one is in the domain of control of window blinds. The second application deals with adaptable light shelves. The third application controls personalized ventilation.

7.2 Important conclusions

Prototype implementations of window blind control and personalized ventilation control demonstrate that integrated control using multi-objective optimization is technically feasible. Laboratory prototype of controllable window blind shows that the system performs well and exhibits patterns of behavior that are intuitive. The system is shown to take actions that conserve energy while respecting user preferences. Full-scale implementation of the personalized ventilation system has been rated highly in a series of subject studies that were conducted in this research.

Hypothetical case studies were used to evaluate the energy savings potential of the window blinds control algorithm. Energy savings of 26% was obtained for a selected case study.

The newly developed control system for adaptable light shelves also demonstrates the potential for energy savings. Using the example of a hypothetical office building under Singapore weather conditions, it is found that about 12% savings in lighting energy is possible compared to a traditional static light shelf.

Personalized ventilation control system provides clear indications of possibilities for energy savings. The most significant finding of this study is that when users are provided individual control over the air flow rate of personalized ventilation, the ambient temperature and PV air temperature do not significantly influence perceived thermal comfort. Users seem to accept warmer temperatures such as 26 deg C. This is significant because there are possibilities for saving energy when the supply air temperature is raised.

7.3 Future work

This research project has developed several ideas that need to be explored further. Some suggestions for future work are given below.

The laboratory prototype of window blind control needs to be improved to make it more realistic. In order to do this, the system needs to be installed on the external window of a fully functional building. The control system should be interfaced with the Building Management System (BMS) in order to obtain real time measurements related to the air conditioning system.

The newly developed light shelf system needs to be tested by implementing a full-scale prototype. Different options for improving the performance of the system and reducing the cost of implementation should be explored. Long term performance of the light shelves and aspects such as maintainability should also be studied.

The control strategy for the personalized ventilation system has been modified based on the feedback from the subject study. Another series of experiments need to be conducted to evaluate the new control strategy.

A new model of PV-ATD has been installed in the FEC that overcomes the drawbacks of the earlier model. The performance of this model needs to be compared with those used in the earlier subject study. The PV Control system might include more complex objectives such as energy and user comfort.

Appendix A: Building model used in the case study


```xml
<?xml version="1.0" encoding="UTF-8"?>
<building>

  <!--
      coordinates x,y,z store the position of components relative to
the containing object.
      dimensions length, width and height of components are in the
local (natural) coordinates of the component

      the global coordinates are oriented like this:
            x-axis in the west - > east direction
            y-axis in the south - >north direction
            z-axis along the height of the building

      the local coordinates for a facade are like this
            x-axis along the length of the facade increasing to the RHS
when viewing from exterior
            y-axis perpendicular to the length of the facade increasing
towards the interior

    -->

  <!-- Total dimensions of the building -->
  <!-- distance in the x direction  between the walls on the east and
west -->
    <length>                  18    </length>         <!-- distance in
the x direction  -->

  <!-- distance in the y direction between the walls on the north and
south -->
    <width>             36    </width>          <!-- distance in the y
direction  -->

    <height>                  4.1   </height>         <!-- total height
-->


    <westfacade>
      <wall id="1">

            <description>  Wall on the west facade </description>

                                                      <!-- Origin shift
- position of the wall relative to the west edge -->
            <x>               0     </x>
            <y>               0     </y>
            <z>               0     </z>

            <length>          36.0  </length>
            <height>          4.1   </height>
            <thickness>       0.15  </thickness>
```

```
            <U>                 0.23  </U>              <!-- Thermal
conductivity  -->
            <reflectivity>          0.6   </reflectivity>

            <window id="W1">

                  <description>  West facade window </description>

                                                        <!-- window
position relative to the wall  -->
                  <x>                 0.5   </x>              <!--
distance from the corner of the wall  -->
                  <z>                 0.3   </z>
                  <length>    35    </length>
                  <height>    2.4 </height>

                  <U>         2.2   </U>                <!-- Thermal
conductivity  -->
                  <SC>        0.39  </SC>               <!-- Shading
coefficient   -->
                  <VT>        0.35  </VT>               <!-- Visible
Transmittance   -->

            </window>


            <overhang id="ext1">

                  <description>  Upper Shade on west </description>

                  <x>                 0            </x>
                        <y>                 0            </y>
                        <z>                 2.7          </z>
                  <length>    36          </length>
                  <width>             0.9          </width>
                  <thickness> 0.07  </thickness>
                  <reflectivity>    0.6   </reflectivity>
            </overhang>

      </wall>
    </westfacade>

    <eastfacade>
      <wall id="2">

            <description>  Wall on the east facade </description>

                                                  <!-- Origin shift -
position of the wall relative to the east edge -->
            <x>                 0     </x>
            <y>                 0     </y>
            <z>                 0     </z>

            <length>            36.0  </length>
            <height>            4.1   </height>
            <thickness>               0.15  </thickness>
```

```
            <U>                 0.23  </U>              <!-- Thermal
conductivity  -->
            <reflectivity>          0.6   </reflectivity>

            <window id="W3">

                  <description>  east facade window </description>

                                                        <!-- window
position relative to the wall  -->
                  <x>                 0.5   </x>              <!--
distance from the corner of the wall  -->
                  <z>                 0.3   </z>
                  <length>    35      </length>
                  <height>    2.4 </height>

                  <U>         2.2     </U>              <!-- Thermal
conductivity  -->
                  <SC>        0.39    </SC>             <!-- Shading
coefficient   -->
                  <VT>        0.35    </VT>             <!-- Visible
Transmittance   -->

            </window>

            <overhang>

                  <description>  Upper shade on the east </description>

                  <x>                 0           </x>
                        <y>                 0           </y>
                        <z>                 2.7         </z>
                  <length>    36            </length>
                  <width>           0.9           </width>
                  <thickness> 0.07  </thickness>
                  <reflectivity>    0.6   </reflectivity>
            </overhang>

      </wall>

    </eastfacade>

    <northfacade>
      <wall id="1">

            <description>  Wall on the north facade </description>

                                                  <!-- Origin shift -
position of the wall relative to the west edge -->
            <x>         0     </x>
            <y>         0     </y>
            <z>         0     </z>

            <length>    18    </length>
            <height>    4.1   </height>
            <thickness>       0.15  </thickness>
```

```
            <U>         0.23  </U>             <!-- Thermal
conductivity  -->
            <reflectivity>         0.8   </reflectivity>

      </wall>
    </northfacade>


    <southfacade>
      <wall id="1">

            <description>  Wall on the south facade between grids 3 and
4 </description>

                                                <!-- Origin shift -
position of the wall relative to the west edge -->
            <x>         0     </x>
            <y>         0     </y>
            <z>         0     </z>

            <length>    18    </length>
            <height>    4.1   </height>
            <thickness>       0.15  </thickness>

            <U>         0.23  </U>             <!-- Thermal
conductivity  -->
            <reflectivity>         0.8   </reflectivity>

      </wall>
    </southfacade>


   <roof>
      <x>               0     </x>
      <y>               0     </y>
      <z>               4.1   </z>

      <length>          18    </length>
      <width>           36    </width>
      <thickness>             0.15  </thickness>

      <U>               0.23  </U>             <!-- Thermal
conductivity  -->
      <reflectivity>          0.1   </reflectivity>
   </roof>


   <ground>
      <reflectivity>          0.1   </reflectivity>
   </ground>



 <zone id="floor1">

   <description> floor 1 office </description>
```

```
   <ceiling>
      <z>                      2.7 </z>
      <reflectivity>           0.8   </reflectivity>
      <thickness>              0.1   </thickness>
   </ceiling>


   <floor>
      <reflectivity>           0.1   </reflectivity>
   </floor>


   <lamps>

      <grid>

            <description>      Interior Lamps in the offices
</description>

            <x>                1.0   </x>
            <y>                1.5   </y>
            <dx>         2     </dx>
            <dy>         3     </dy>
            <nx>         9     </nx>
            <ny>         12    </ny>

    <!-- If sensor is not specified, there is assumed to be one sensor
for each lamp -->
    <!--
        <sensor>
          <x>            2     </x>
          <y>            13.5  </y>
          <influence>    1     </influence>
        </sensor>
      -->

            <lamp>
                  <watt>             56    </watt>

                  <!-- Ballast and other losses -->
                      <loss>    4    </loss>

                  <lux>        500   </lux>             <!-- min lux
provided by the lamp within its radius of influence  -->

                  <target>    500   </target>   <!-- target lighting
level  -->
                  <influence> 1.4   </influence>      <!-- radius of
influence of the lamp  -->
                  <dimmable>  yes   </dimmable>
            </lamp>
      </grid>

   </lamps>
```

```
      <grid id="1">
            <x>                 1     </x>
            <y>                 1     </y>
            <z>                 0.8   </z>        <!-- Height of
workplane  -->
            <dx>          0.5   </dx>
            <dy>          0.5   </dy>
            <nx>          33    </nx>
            <ny>          69    </ny>
      </grid>


 </zone>
 <!-- end of zone floor1  -->
 <!--
________________________________________________________________________
___________________________________ -->




</building>
```

Appendix B: System data used in the case study


```
<?xml version="1.0" encoding="UTF-8"?>

<systemdata>                  <!--  Data related to building systems  -->

       <chiller>
              <capacity>
                     <!-- Capacity in ton  -->
                     53
              </capacity>

              <minimumload>
                     <!-- Minimum chiller load in ton  -->
                     10
              </minimumload>

              <power>
                                                       <!-- % of maximum load -->
                     <partload>     100     </partload>
                                                       <!-- power rating in kw/RT  -
->
                     <rating>       0.693   </rating>
              </power>
              <power>
                                                       <!-- % of maximum load -->
                     <partload>     75      </partload>
                                                       <!-- power rating in kw/RT  -
->
                     <rating>       0.693   </rating>
              </power>
              <power>
                                                       <!-- % of maximum load -->
                     <partload>     50      </partload>
                                                       <!-- power rating in kw/RT  -
->
                     <rating>       0.693   </rating>
              </power>
              <power>
                                                       <!-- % of maximum load -->
                     <partload>     25      </partload>
                                                       <!-- power rating in kw/RT  -
->
                     <rating>       0.693   </rating>
              </power>

       </chiller>

       <coolingtower>
              <power>       2.2           </power>
              <VSD>         yes           </VSD>
       </coolingtower>

       <chillerpump>
              <power>       2.2           </power>
              <VSD>         yes           </VSD>
       </chillerpump>

       <condenserpump>
              <power>       2.2           </power>
              <VSD>         no            </VSD>
       </condenserpump>

       <zone id="floor1">
              <plugload>
                     <!-- Electrical load in KW  -->
                     4
              </plugload>
```

```xml
            <occupants>
                    100
            </occupants>
    </zone>

    <zone id="floor2">
            <plugload>
                    <!-- Electrical load in KW  -->
                    4
            </plugload>

            <occupants>
                    100
            </occupants>
            <designcooling>
                    <!-- Design cooling load in KW  -->
                    50
            </designcooling>
    </zone>

</systemdata>
```