Building Simulation 2003

Eighth International IBPSA Conference
Eindhoven, Netherlands
August 11-14, 2003

A REPRESENTATIONAL FRAMEWORK FOR BUILDING SYSTEMS CONTROL

Kristie Bosko Mertz[1] and Ardeshir Mahdavi[2]
[1] Center for Building Performance and Diagnostics,
Carnegie Mellon University, Pittsburgh PA USA
[2] Department of Building Physics and Human Ecology,
Vienna University of Technology, Vienna, Austria

ABSTRACT

We present a representational framework for the integrated operation of multiple environmental systems in buildings. A methodology for the automatic generation of this framework is introduced. The implementation of both rule-based and simulation-based control algorithms in this framework is demonstrated.

INTRODUCTION

Building control systems and approaches currently do not effectively integrate the operation of multiple environmental systems. A contributing factor to this circumstance is the absence of representational systems (control systems models) that support the cooperation between multiple devices in multiple environmental service domains. This paper presents, thus, a representational framework whose hierarchically organized components can encapsulate a variety of decision-making algorithms toward integrated building systems control. Moreover, a rule-based approach is designed and implemented to automatically generate this framework for a variety of building control scenarios. As a proof of concept, an implementation is described which shows a distributed hierarchy of device controllers and higher-level controllers (meta-controllers) that embody rule-based and/or simulation-based decision-making algorithms toward integrated building systems control.

FRAMEWORK

The relationship between device controllers (DCs), sensors, and devices represents the most basic type of control system interaction, as shown in Figure 1. DCs receive zone state information directly from sensors, and, utilizing an algorithm that defines the relationship between the device state and its sensory impact, set the state of the devices. Real world building control problems are, however, much more complex, as they involve the operation of multiple devices for each environmental system domain (e.g., lighting, heating) and multiple environmental system domains. Thus, system features are needed that would integrate the operation of these entities. Such integrative processes can be distributed among multiple higher-level or "meta"-controllers (MCs), which could encapsulate control function in a distributed fashion.

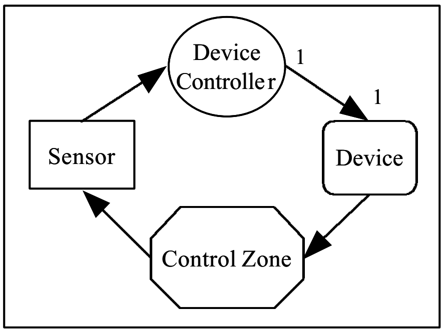

Figure 1: Basic Control System Interaction

Such a meta-controller is required: *i)* to coordinate the operation of identical, separately-controllable devices, *ii)* to enable cooperation between different devices in the same environmental service domain, and *iii)* to aggregate information about the impact of a single device on multiple sensors.

An example of the first case is shown in Figure 2, where an MC is needed to coordinate the operation of two electric lights to achieve interior illuminance goals in a single control zone.

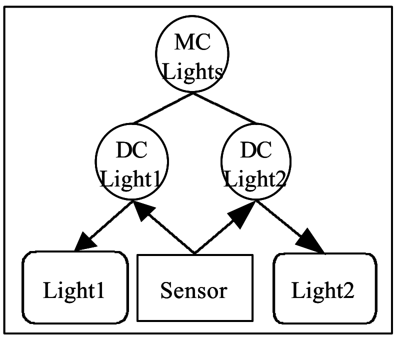

Figure 2: Identical Devices, One Sensor

In the second case (see Figure 3), movable blinds and electric lights are coordinated to integrate daylighting with electric lighting.

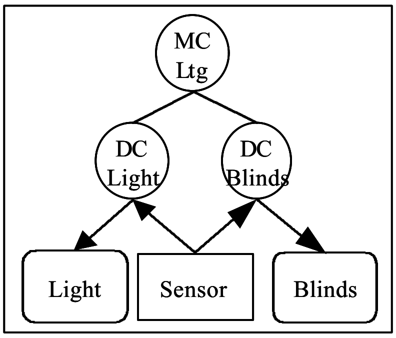

Figure 3: Multiple Domains, One Sensor

Figure 4 depicts case three, in which a single device that impacts the state of more than one space is controlled by aggregating the sensory information in each space.

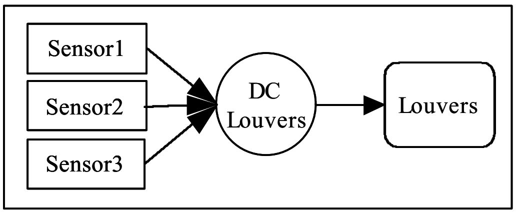

Figure 4: Multiple Sensors, One Device

In actual building control scenarios, many different combinations of the above examples can be found. Thus, the manner in which the control system functionality is distributed among the controllers must be clearly defined. The object model must be created using a logical, coherent, and repeatable method so that it can be used for a diverse set of building controls application. In our solution, a set of constitutive rules allow for the automated generation of the control system model (Mahdavi 2001a, 2001b). Specifically, five model generation rules are applied successively to the control problem, resulting in a unique configuration of nodes that constitute the representational framework for a given control context.

Model Generation Rules

1. Multiple devices of the same type that are differentially controllable and that affect the same sensor need a meta-controller.
2. More than one device of different types that affect the same sensor needs a meta-controller.
3. More than one first-order meta-controller affecting the same device controller needs a second-order (higher-level) meta-controller.
4. Remove any new node whose functionality duplicates that of an existing node.
5. If Rule 4 was applied, re-connect any isolated nodes.

The following example illustrates these Rules and describes the functionality of each node in the resulting object model. The scenario includes two adjacent rooms, each with four luminaires and one local heating valve, which share exterior movable louvers (see Figure 5). Hot water is provided by the central system, which modulates the pump and valve state to achieve the desired water supply temperature. In each space, illuminance and temperature is to be maintained within the setpoint range.

The control zones are defined by describing the relationship between the sensors and devices. It must be determined which devices influence which sensors. In this case, an interior illuminance sensor (E) and a temperature sensor (t) are located in each space. The sensors for Space1 are called E_1 and t_1, and those for Space2 are called E_2 and t_2. In Space1, both the louvers and electric lights can be used to meet the illumination need. As shown in Figure 6, Sensor E_1 is influenced by the louver state, controlled by DC-Lo1, and the state of four electric lights, each controlled by a DC-EL. Similarly, both the local valve state and the louver state influence the temperature in Space1, t_1. Analogous assumptions apply to Space2.

Once the control zones have been defined, the Model Generation Rules can be applied to this scenario. By applying them to the control problem as illustrated in Figure 6, the model of Figure 7 results. A summary of the application of Rules 1, 2, and 3 is shown in Table 1. In the application of Rule 1, four nodes, DC-EL1, EL2, EL3, and EL4 are the same type of device and all impact sensor E_1. Thus, an MC is needed to coordinate their action: MC-EL_1. Similarly, in the application of Rule 2, both DC-Lo1 and DC-Va1 impact the temperature of Space1; thus, MC-Lo_Va_1 is needed to coordinate their action. For Rule 3, four MC nodes are controlling the DC-Lo1 node; thus, their actions must be coordinated by an MC of second order, MC-II EL_Lo_Va_1.

In this example, Rules 1, 2, and 3 were applied to the control problem to construct the object model. With this consistent method, a model of distributed, hierarchical control nodes can be constructed.

In certain cases, the control problem contains characteristics that cause the model not to converge to a single top-level controller. In these cases, Rules 4 and 5 can be applied to ensure convergence.

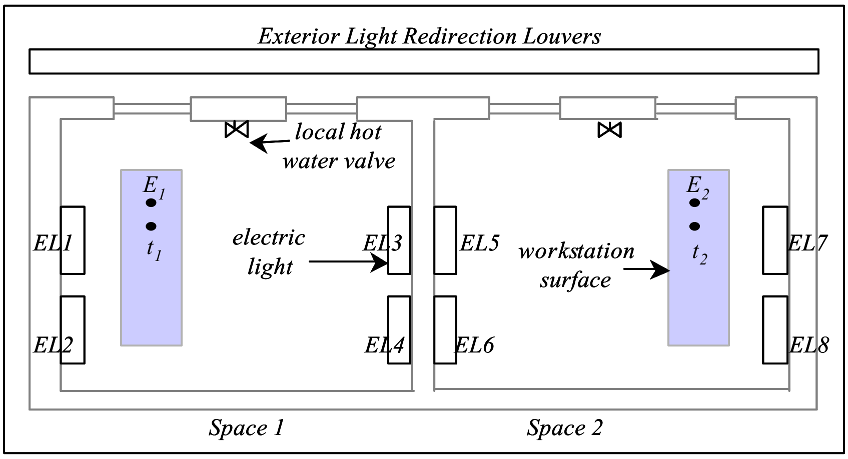

Figure 5: Floor Plan of Spaces for Example

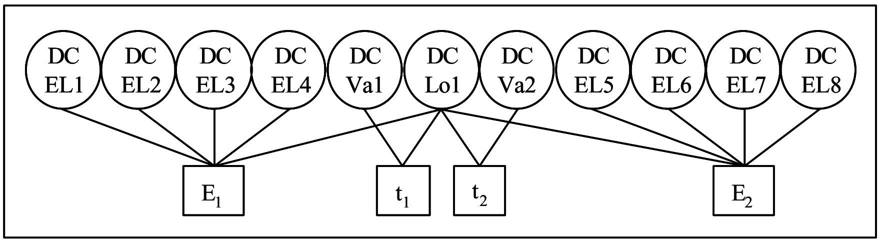

Figure 6: Relationship between Sensors and Devices for Example

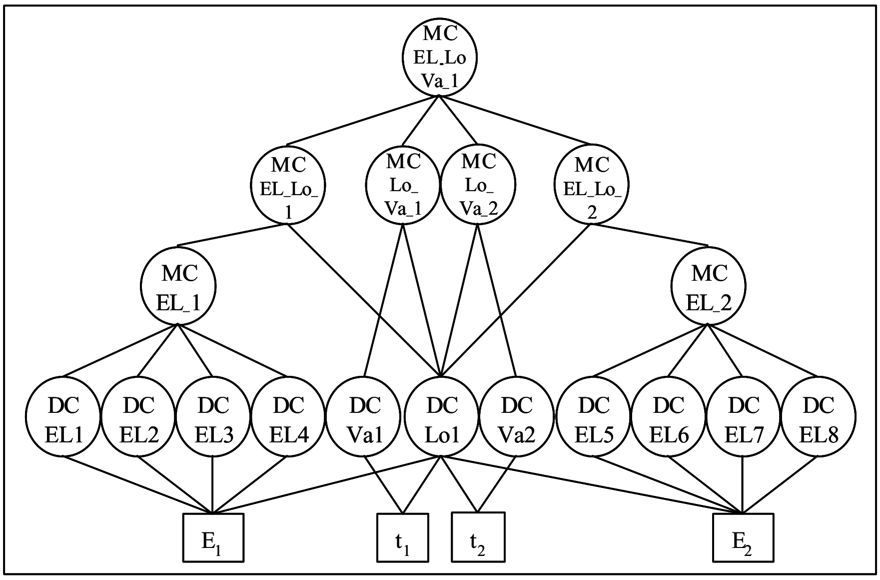

Figure 7: Object Model for Automatic Generation

Table 1: Application of Rules 1, 2, and 3

Multiple Controllers	Affected Sensor	Affected Device	Meta-Controller
Application of Rule 1			
EL1, EL2, EL3, EL4	E_1	N/A	MC-EL_1
EL5, EL6, EL7, EL8	E_2	N/A	MC-EL_2
Application of Rule 2			
Lo1, Va1	t_1	N/A	MC-Lo_Va_1
Lo1, Va2	t_2	N/A	MC-Lo_Va_2
EL_1, Lo1	E_1	N/A	MC-EL_Lo_1
EL_2, Lo1	E_2	N/A	MC-EL_Lo_2
Application of Rule 3			
EL_Lo_1, EL_Lo_2, Lo_Va_1, Lo_Va_2	N/A	Lo1	MC-II EL_Lo_Va_1

The following example illustrates the application of these two rules. Rule 4 is used to ensure that model functionality is not duplicated; the means of detecting a duplicated node is in the node name. Figure 8 shows a model that was constructed using Rules 1, 2, and 3; the application of these rules is summarized in Table 2.

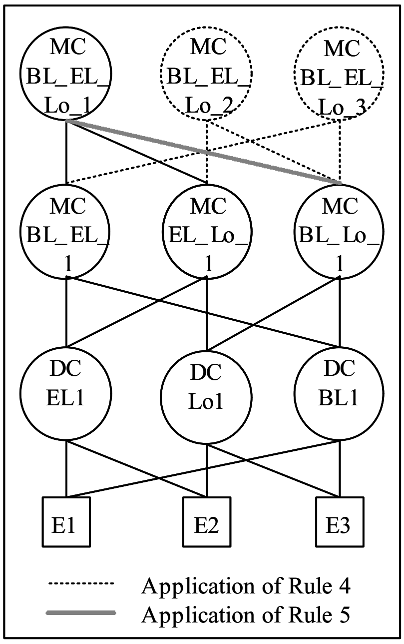

Figure 8: Application of Rules 4 and 5

Rule 1 does not apply in this case because there are three different device types. For the application of Rule 2, DC-EL1 and DC-BL1 both impact the state of E1 and thus MC-BL_EL_1 is needed to negotiate between them. Three MC nodes are created in this manner. When Rule 3 is applied, three MCs of second order are created. It is apparent that the model will not converge. In addition, the three nodes have the same name: MC-BL_EL_Lo_. This is an indication of duplicated functionality, specifically the coordination of devices BL, EL, and Lo. Thus, by Rule 4, nodes MC-BL_EL_Lo_2 and MC-BL_EL_Lo_3 are removed, and by Rule 5, node MC-BL_Lo_1, which is left without a parent node, is connected to the MC-BL_EL_Lo_1.

Table 2: Application of Rules 2 and 3

Multiple Controllers	Affected Sensor	Affected Device	Meta-Controller
Application of Rule 2			
EL1, BL1	E_1	N/A	MC-BL_EL_1
EL1, Lo1	E_2	N/A	MC-EL_Lo_1
BL1, Lo1	E_3	N/A	MC-BL_Lo_1
Application of Rule 3			
BL_EL_1, EL_Lo_1	E_1	DC-EL1	MC-BL_EL_Lo_1
EL_Lo_1, BL_Lo_1	E_2	DC-Lo1	MC-BL_EL_Lo_2
BL_EL_1, BL_Lo_1	E_3	DC-BL1	MC-BL_EL_Lo_3

CONTROL METHODS

Introduction

Both rule-based control (RBC) and simulation-based control (SBC) can be implemented in a building control system. Within this representational framework, each node can encapsulate a decision-making algorithm, and either RBC or SBC can be used within any of these nodes, at any level of the control system object model.

Rule-based control

In rule-based control, a simple statement describes the control function used to make decisions. As an example, the type of a rule used within the DC node defines the relationship between the state of a device and its corresponding impact on the state of the sensor.

Rules can be developed through a variety of techniques. For example, rules can rely on the knowledge and experience of the facilities manager, the measured data in the space to be controlled, or logical reasoning. The following example describes how rules were developed from measured data to capture the impact that the states of 4 electric lights had on the space interior illuminance.

The floor plan (see Figure 9) shows the location of four electric lights (EL) and two interior illuminance sensors (E).

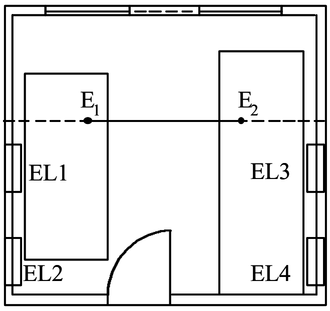

Figure 9: Floor Plan for Electric Light Rule Development

The luminaire rules were developed from measured data taken at night so that the influence of daylight on the results was excluded. The electric lights were individually dimmed from 100% to 0% at 10% intervals, and the desktop illuminance at two locations was measured. Figure 10 and Figure 11 show the impact each luminaire has on sensors E_1 and E_2.

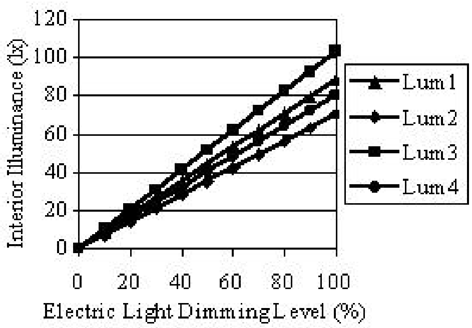

Figure 10: Interior Illuminance Rule, E_1

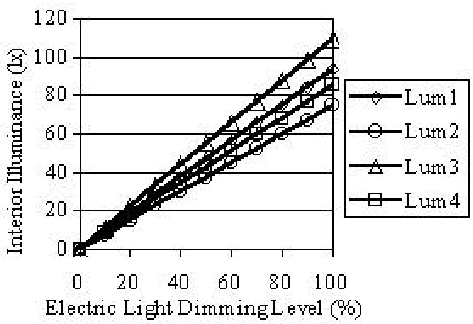

Figure 11: Interior Illuminance Rule, E_2

A second type of rule that could be implemented in this framework is the control strategy. At the level of the MC, coordination is accomplished by setting guidelines about what combination of device states are permitted to achieve the goal. For example, one may limit the control decision to that which fully utilizes daylighting before electric lighting.

Simulation-based control

Simulation-based control (Mahdavi 2001c, 1997, Mahdavi et al. 2000) can be used for building control within this representational framework through the following procedure: *i)* generation of multiple potential building states; *ii)* simulation of chosen states; *iii)* evaluation and ranking of simulation results based on an objective function

The state of a building is described by a combination of the states of its devices. For each node that utilizes SBC, multiple building states are generated from the state space of options, each of which is a potential control decision. When the state space is very large, a number of methods can be used to reduce it. In this research, the device states were discretized and state space reduction rules were applied. An example of such a rule is to choose only those building states where the individual device state moves in the appropriate direction, say that of increased service, based upon a reported sensor state below the setpoint. Once the final number of building states is chosen, each is simulated. However, this assumes that the simulation application has been calibrated to accurately reflect the behavior of the space. The calibration procedure is an iterative one, where the building is first simulated with initial assumptions based upon estimated values of space characteristics; if the simulation results do not quite match measured data, then the assumptions are corrected and the simulation repeated.

One of the simulation programs used in the present research was Lumina (Pal and Mahdavi 1999). Lumina predicts illuminance levels given space geometry, organization, surface characteristics, and devices, plus a luminance distribution model of the sky patches. Figure 12 and Figure 13 show pre- and post-calibration results from Lumina compared with measured data from the space it modeled. The calibration procedure improved Lumina's ability to accurately predict light levels in the space.

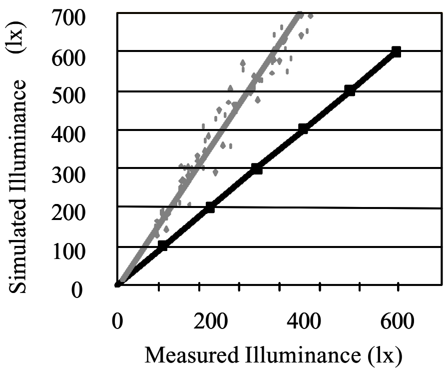

Figure 12: Pre-Calibration Results, E_1

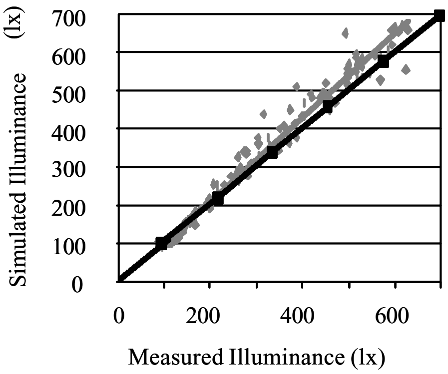

Figure 13: Post-Calibrated Results, E_1

The simulation results are evaluated and ranked according to their ability to satisfy the conditions of the objective function.

To implement SBC within the control system framework, an interface is necessary to map the control system data to the simulation and the results from the simulation to the controller, as shown in Figure 14. Each controller has a separate interface for each simulation.

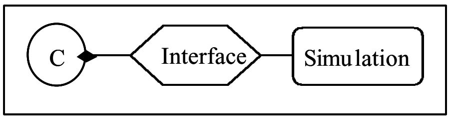

Figure 14: Controller-Simulation Interface

IMPLEMENTATION

Introduction

The goal of this demonstrative implementation is to investigate cooperation among devices within in the same building service domain (e.g., heating, lighting), the interaction between multiple domains, and the interaction between two spaces that share the same device. The objective function of the control system is to maintain all sensors within their setpoint range while obeying the rules at each node. Rule-based and simulation-based control functionalities are applied.

Implementation Parameters

The configuration of the spaces used in this implementation has been presented in Figure 5. Space1 is the test space in which measured data was collected for calibration of simulations and for sensory input during this implementation. Space2 is a virtual space, which is assumed to adjoin Space1, sharing geometric and organizational characteristics of Space1 plus the exterior movable louvers device. Measured data from the north side of the test space, including illuminance and temperature, represent the north side of the virtual test space.

Each space contains four electric lights and a local heating valve. An exterior light-redirection louver system is shared by both spaces. The device states have been discretized for state space reduction, and the performance indicators impacted by each device are listed in Table 3 as interior illuminance and temperature.

Table 3: Implementation Parameters

Type and Number of Devices		States	Performance Indicators
Light Redirection Louvers	1	0°, 70°, 90°, and 105° from vertical	illuminance, temperature
Electric Lights	8	0%, 33%, 67%, and 100%	illuminance
Heating Valve	2	0% - 100% in 5% increments	temperature

The control system was operated for 4 days (in the heating season during daylight hours) for which the following sensor data was available: interior illuminance and temperature for two spaces, outdoor dry bulb temperature, global and diffuse irradiation, and central system hot water supply temperature.

Algorithms

The object model generated for this implementation is presented in Figure 7. During the experiment, SBC was utilized for nodes DC-Va1, DC-Va2, and DC-Lo1. RBC was used for the remaining nodes. The rules used in the DC-EL nodes were derived from Figure 10 and Figure 11. The rules utilized at each MC node are summarized in Table 4.

Table 4: Rules Used for Implementation

Node	Rule
MC-EL_1 and MC-EL_2	Prohibit independent switching of lights
MC-EL_Lo_1 and MC-EL_Lo_2	Fully utilize daylighting before electric lighting
MC-Lo_Va_1 and MC-Lo_Va_2	Fully utilize solar heat before mechanical lighting
MC-II EL_Lo_Va_1	Choose option that meets setpoint need of all sensors

To implement SBC, the Nodem simulation tool was also used (Mahdavi and Mathew 1995, Mathew and Mahdavi 1998). Nodem predicts interior temperature by balancing heat gains and losses in the space.

The local heating valve was simulated as a heat gain to the space, which was added to other internal loads in Nodem. It was necessary to determine how much heat gain to each space is possible through the water local heating system at each valve state. The local supply temperature is dependent on the central supply temperature, which changes continually due to the changing needs of the building. Likewise, according to the following equation, heat supplied to the space is dependent on local supply temperature. Thus, the amount of heat provided by the local heating system changes with constant valve state.

$$Q = m \cdot c_P \cdot (T_1 - T_2) \quad (1)$$

m: mass flow rate of water in the pipe

c_P: specific heat at constant pressure

T_1: local pipe surface temperature at the entrance to the space

T_2: local pipe surface temperature at the exit from the space

Estimating the losses from the mullion pipes to the space was accomplished by estimating the local water flow rate and measuring the surface temperatures at both ends of pipe. Over the course of several days in the winter, the water mullion valve was moved to a new position every 20 minutes, and the resulting surface temperatures measured. The heat loss to the space was calculated for a valve position of 100% and binned according to the central system water supply temperature. The results are graphed in Figure. To utilize the chart, at each timestep the hot water supply temperature for the space from measured data is necessary. Thus, the DC-Va nodes use a rule to estimate values that are used in the simulation for control purposes.

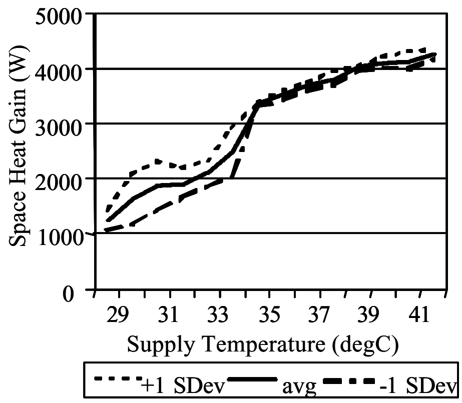

Figure 14: DC-Va Heat Supply, 100%

The louvers are simulated in both Lumina and Nodem. In Lumina, illuminance changes due to louver position were determined by modeling the louver as an exterior surface. Lumina calculates the inter-reflections of light between the louver surfaces as well as between the louvers and window surfaces. The result is an accurate estimation of the amount of light entering the space.

To calculate the amount of solar heat gain to the space at a given time, Lumina was used as well. For this purpose, it was necessary to make estimation about the relationship between the visible energy (light) and radiant energy (heat) that reaches the outside face of the window glass. The resulting solar heat gain was then input into Nodem as an additional heat gain.

Operation

The operation of the control system at each timestep begins with measured illuminance and temperature data that are mapped to the sensor representations in the object model. The device controllers read the new sensor values, determine whether they are out of range, decide on appropriate action based on their decision-making algorithm, and submit a request to their meta-controller parent(s). The format of the request is a table of optional device states and their corresponding predicted impact on each sensor with which they are associated. For example, in Figure 7, the DC-Lo1 reads the values of sensors E_1, E_2, t_1, and t_2. It then sends a request to each of four MCs: EL_Lo1, EL_Lo_2, Lo_Va_1, and Lo_Va_2.

The meta-controller is responsible for aggregating the requests of its child nodes, applying the decision-making algorithm, which is in this case a rule, ranking the options, and supplying its parent node(s) with a list of state options for each device state for which it is responsible.

At the highest level, the controller makes the final decision, sending the results back down through the hierarchy to the device controllers, which then set the new device states.

RESULTS

Figure 15 shows the simulated thermal performance of the Space1 when controlled by this control system. The interior temperature is maintained within its setpoint range. Appropriate action of the water mullion valve in response to decreasing temperature is seen as it opens the valve incrementally.

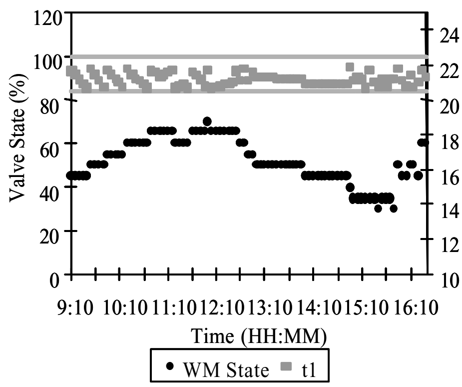

Figure 15: Thermal Results for Day1

Figure 16 shows the simulated interior illuminance in Space1 for Day1. The cooperation between electric lights (EL) and daylight is apparent as the electric light component of illuminance drops (EL state decreases) as daylight increases. The total interior illuminance is generally maintained within the setpoint range (400-600 lx).

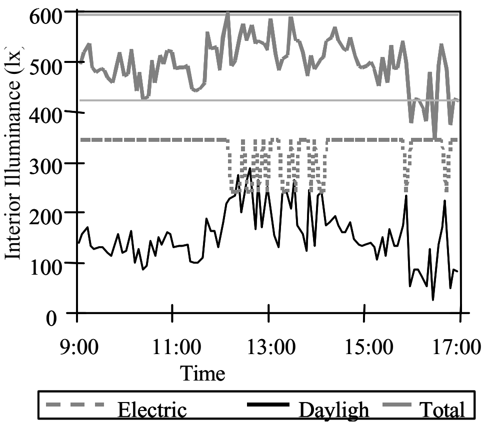

Figure 16: Illuminance Results for Day1

CONCLUSION

We presented a modular, flexible, and hierarchical representational framework for integrated building systems control. This representation can be automatically generated based on minimal spatial input requirements, thus enabling the control system to effectively respond to changes in spatial organization as well as control zone configuration and components. The implemented rule-based and simulation-based control algorithms and methods that animate the control system representation enable the consideration of the i) effects of control actions on multiple zonal performance indicators, ii) influences of devices in multiple domains, iii) conflicts amongst devices used to control multiple spaces, iv) cooperation between local and central systems, v) multiple control objective functions.

Future research must address in more detail implementation problems that may arise in view of scalabilty (e.g. addressing larger buildings with a larger number of spaces and systems, additional environmental systems such as security systems) and, event-handling (i.e., dealing with multiple temporal intervals and asynchronous processes in decision-making sequences). In addition, a more systematic exploration of the computational methods toward control state space reduction and option evaluation (ranking) is necessary, in order to arrive at efficient utilization of simulation-based control algorithms.

REFERENCES

Mahdavi, A. (2001a): Aspects of self-aware buildings. International Journal of Design Sciences and Technology. Europia: Paris, France.Volume 9, Number 1. ISSN 1630 – 7267. pp. 35 – 52.

Mahdavi, A. (2001b): Über "selbstbewusste" Gebäude. Österreichische Ingenieur- und Architekten-Zeitschrift (ÖIAZ), 146. Jg., Heft 5-6/2001. pp. 238 – 247.

Mahdavi, A. (2001c): Simulation-based control of building systems operation. Building and Environment. Volume 36, Issue 6, ISSN: 0360-1323. pp. 789-796.

Mahdavi, A. (1997): Toward a Simulation-assisted Dynamic Building Control Strategy. Proceedings of the Fifth International IBPSA (International Building Performance Simulation Association) Conference. Vol. I, pp. 291 - 294.

Mahdavi, A. and Mathew, P. (1995): Synchronous Generation of Homologous Representation in an Active, Multi-Aspect Design Environment. Proceedings of the Fourth International Conference of the International Building Performance Simulation Association (IBPSA) Madison, Wisconsin. pp. 522 - 528.

Mahdavi, A. – Chang, S. – Pal, V. (2000): Exploring Model-Based Reasoning in Lighting Systems Control. Journal of the Illuminating Engineering Society.. Volume 29. Number 1. Winter 2000. pp. 34 - 40.

Mathew, P. and Mahdavi, A. (1998): High-Resolution Thermal Modeling for Computational Building Design Assistance. Computing in Civil Engineering; Proceedings of the International Computing Congress, 1998 ASCE Annual Convention. pp. 522-533.

Pal, V. and Mahdavi, A. (1999): Integrated Lighting Simulation within a Multi-Domain Computational Design Support Environment. Proceedings of the 1999 IESNA Annual Conference. New Orleans. pp. 377 - 386.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of S. Lee, R. Ries, G. Zimmermann, and V. Hartkopf toward the completion of this study.