Optimised Embedded Distributed Controller for Automated Lighting Systems

Alie El-Din Mady, Menouer Boubekeur and Gregory Provan

Abstract—**The paper introduces a model-driven hybrid/multi-agent platform for the design and analysis of building automation systems. It describes an optimised parameterizable and predictive distributed control methodology for automated lighting systems. The modelling steps and the simulation results for a typical lighting system scenario are outlined throughout the paper. Moreover, the performance for the wireless network is evaluated. The contribution of the proposed lighting control strategy is highlighted by comparing it with several control techniques.**

Index Terms—**Lighting System Control, Hybrid System, PPD-Controller, Charon, Embedded Middleware, Distributed Control.**

I. Introduction

INTELLIGENT (or smart) buildings incorporate a Building Management System (BMS) to maintain a comfortable environment in an energy-efficient manner. A typical BMS would provide a core functionality that keeps the building's climate within a specified range, automates the lighting based on occupancy, and monitors system performance and device failures.

One major source of energy inefficiency in buildings is lighting, which can account for up to 30% of total energy waste in some retail and public offices [1]. The aim of our research is to define a methodology for efficiently model and integrate building management system services, with a focus on lighting and Heating, Ventilating, and Air Conditioning (HVAC) systems. In line with the recent focus on "energy management through active control" in the energy and control community, our work provides intelligent controllers for more energy-efficient buildings.

Given the rapidly growing complexity of modern building control systems, the centralized control approach faces numerous challenges in scaling, delays associated with collecting data, inefficient energy consumption, and unstable control tendencies [2], [3] (i.e., continuously oscillating around the set points). Further, the different requirements of different services place many challenges on centralized control solutions; for example, in lighting control, reaction times are anticipated within fractions of a second, whereas in HVAC control, the process dynamics is much slower and the sampling/actuation time is much larger. Rather than adopt worst-case timing solutions in a centralized controller when integrating several processes, a distributed approach may provide an better solution for time-scale challenges, by ensuring fast response and reducing the dependency on network communication.

Alie El-Din Mady, Menouer Boubekeur and Gregory Provan are with Cork Complex Systems Lab (CCSL), Computer Science Department, University College Cork (UCC), Cork, Ireland. website: http://www.cs.ucc.ie/ccsl/index.htm, e-mail: {mae1, m.boubekeur, g.provan}@cs.ucc.ie

Our ongoing research work consists of developing an integrated platform for intelligent control of building automation systems. This platform provides, among other features, predictability, reconfiguration, distribution and building energy optimisation. As shown in Fig. 1, the system design flow starts by defining relevant scenarios to be operated within the building. These scenarios are defined using the Unified Modelling Language (UML) [4]. The UML models are interpreted using specific models for simulations and analysis purposes. At this level we allow an optimisation loop to optimise the model at an early stage of the development. When the simulation gives satisfactory results, the models are auto-translated into embeddable code to be deployed over a distributed sensor/actuator network [5].

The integration process is performed using a model-/service-based middleware [6] platform, which connects components and facilitates data exchange. In this approach, all the different components of the architecture collaborate with the requirements module to ensure that the requirements are adhered to.

The main features of our platform will be illustrated through an example of a lighting system for an office area. This example illustrates the combination of discrete-event behaviour (presence detection, light actuation levels) and hybrid properties for the luminosity control, i.e., where both discrete and continuous aspects are considered. We describe a distributed lighting control system, which is embedded in a wireless network, that is both simple and effective. The lighting system has been modelled using our hybrid/multi-agent platform; the generated code has been emulated using the Java Sun-Spot platform [7]. We study several Quality of Service (QoS) metrics of the underlying Wireless Sensor/Actuator Network (WSAN) [8] using the VisualSense tool [9]. These metrics are essential to evaluate the safety, reliability and user comfort (i.e. the difference between the sensed value and the user preference) of the overall control application.

The remainder of the paper is organized as follows: Section II provides a survey covering the related work and discusses our contribution comparing to the state of the art. The proposed Parameterizable/Predictable Distributed Controller and its specification are discussed in Section III. Section IV introduces the Charon modelling of the lighting system and Section V describes the optimisation techniques we have used. In Section VI, we outline and discuss the simulation results. We end in Section VII by giving a discussion of our work and outlining future perspectives.

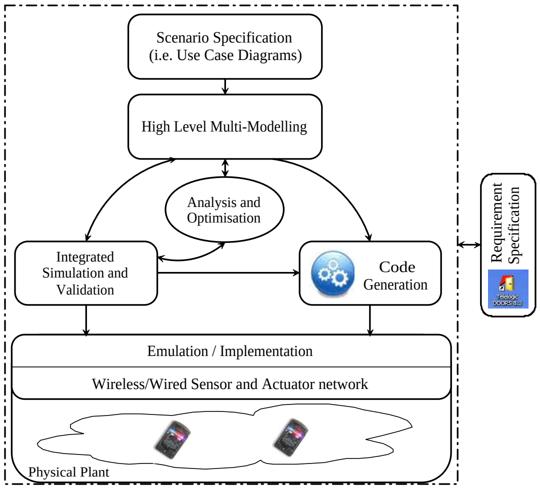


Fig. 1. System Architecture

II. Related Work

In the domain of control systems, there has been work on distributed control, especially of discrete-event systems, e.g., [10], [11]. However, this work is primarily of a theoretical nature, and has not been applied to the domain of automated energy systems. Recently, some work on distributed automation of energy systems has been appeared within the Multi-Agent Systems (MAS) community, e.g., [12]. This work focuses on the distribution of agents and on agent communication, rather than on the issues more pertinent to control theory, such as liveness, non-blocking, reachability, etc.

A wide range of research papers have considered the control of the lighting system using centralized controllers, where an optimisation engine is used to improve the energy consumption at the high level. In most cases, these approaches lack a clear modelling approach, don't consider issues such as daylight control, predictability, and reconfigurability, and use centralized controllers. However, [13] considers daylight control using an image processing technique, which is not suitable to be deployed on a limited-resource micro-controller.

Our contribution is to provide a parameterizable/predictive distributed control strategy that can improve the energy efficiency of lighting systems while guaranteeing particular levels of user comfort. We also aim to enhance the WSAN QoS through the implementation of a distributed system, thereby avoiding the previously-mentioned problems of a large-scale WSAN for the centralized control strategy, e.g., limitations in scaling and control instability. We use a simulation model to evaluate the system performance and improve the flexibility of the control strategy before deployment.

III. Parameterizable/Predictable Distributed Controller

In order to increase the control reliability, scalability, resource sharing and concurrency, a distributed control model [14] has been considered. In this context we have developed a Parameterizable and Predictable Distributed controller (called PPD-Controller) for automated lighting systems. The PPD-Controller is described throughout the following sections.

A. Lighting Model Specification

The most common lighting controller is the bounce controller, which switches the light on/off depending on the occupancy and the ambient light levels [15]. When a person is detected in the controlled area and the daylight luminance is below (above) a certain threshold, the controller turns the light on (off).

Another type of lighting control is the dimming control (manual or automatic), where the light luminance is controlled using DAC/PWM, which provides the control voltage/duty cycle as discrete values [16].

In our work we have considered an open office area with a typical architecture, as shown in Fig. 2. It contains 10 controlled zones; each zone contains one artificial light, one light sensor and one Radio-Frequency Identification (RFID) receiver. There are 4 windows/bindings on the right and left borders of the open area, and a fixed number of predefined occupant positions.

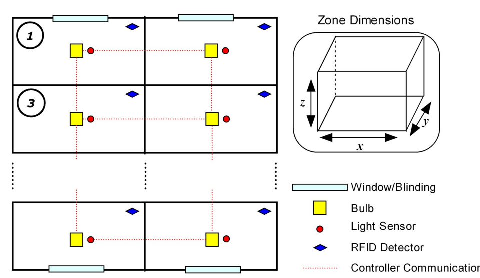


Fig. 2. Model Specification

Our lighting model includes integrated blind and lighting controls. In order to enhance the efficiency of the resulting control model, an optimisation technique has been implemented, as explained in Section V. The optimisation engine selects the light luminance and blind position depending on the user preferences and the energy consumed by the artificial light and the blinding actuators.

In brief, the lighting system scenario behaves as follows:

1. The user can switch on/off the automatic lighting system for several zones, or for the entire system (through a technician).
2. The users provide their preferences (light luminance and blinding position).
3. A person is tracked in each zone using RFID in order to service his preferences, which are ignored whenever he leaves his zone.

B. Control Strategy

In our PPD-Controller, the control functionality is distributed over 10 zones, where each zone contains one artificial light source and one light sensor. Depending on the sensor reading, the local controller modifies the artificial light source to achieve the "optimal" ambient light. It has been implemented as a closed loop controller, used to predict the next sample actuation value. The system is constructed in a modular way, for example the controllers of the zones that contain windows incorporate a separate module for blinding actuation.

In order to increase the flexibility of the control system, we have designed the controller so that a range of global parameters can be assigned and/or reassigned by users at any time. For example, the use can assign a priority parameter to specify that the occupants of a given zone have a high priority; therefore they can exert full manual control. The parameterization can also be used for setting different parameters for the distributed local controllers (i.e. considering the blinding, switching on/off the local controller, ...). This can also help with the distributed optimisation process, as will be explained later.

C. Control Model Description

In this section, we describe the overall control model. Fig. 3 shows the model of a local controller and its interactions with the environment models. The local controller modifies the light intensity inside its zone as follows:

1. The preference solver receives the user preferences for each zone, sends the optimal light luminance and blinding position back to the local optimisation engine.
2. The optimisation engine calculates the optimal actuation settings (artificial light level, blinding position) and sends them to the PI-Controller.
3. The controller actuates the artificial light and the blinding position accordingly, then go to 2 only if the preference has been changed otherwise the PI-Controller actuates only the artificial light relying on the external light and the light interference. The controller ignores blind updates triggered by minor changes in ambient light, since it leads to discomfort for the users.

The PI-Controller, as shown in Fig. 3, is used to predict the next actuation setting for the lighting level in a closed-loop fashion [17] using Eq. 1. The light-level refinement is one level, as the optimisation engine is used to recommend the initial setting for the controller. The PI-Controller has two main statuses: (a) the first is unstable when the difference between the sensed light intensity and the optimal one is greater than 70 Lux (one light actuation level), and (b) the second is stable, if the difference is less than or equal to 70 Lux.

A Light/Blinding Occlusion Preference Solver agent is used to provide the intermediate solution between several luminance/glare preferences in the same controlled zone.

$$A(t+1) = A(t) + \theta \qquad (1)$$
$$U(t) = A(t) + E(t) + I(t)$$

$$\theta = \begin{cases} \gamma - \dfrac{\beta}{\rho}, & \forall\ U(t) - S(t) > \epsilon \\ \dfrac{\beta}{\rho} - \gamma, & \forall\ S(t) - U(t) > \epsilon \\ 0, & \forall\ |S(t) - U(t)| \leq \epsilon \end{cases}$$

where: $A(t)$ is the actuation setting for light/blinding actuators, $E(t)$ is the daylight intensity (Lux), $I(t)$ is the interference light intensity (Lux), $U(t)$ is the sensed light intensity (Lux), $S(t)$ is the optimal preference settings, ϵ is the luminance level produced from a single dimming level (70 Lux), β is the maximum light intensity error (700 Lux), γ is the minimal light intensity error (0 Lux) and ρ is the total number of dimming levels (10 levels).

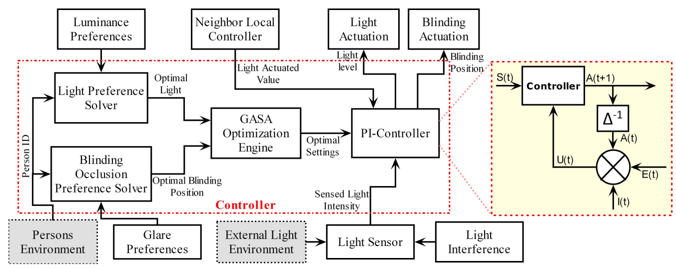


Fig. 3. Control Model

D. WSAN Deployment

We embed each PPD-Controller within a wireless node in a WSAN. Each local controller communicates with a light sensor, actuators (light, blinding), an RFID detector, and also with the neighbouring local-controllers as shown in Fig. 2. Each RFID device has been modelled as an event-driven agent fired whenever a person comes/leaves to/from the controlled zone, and then sends the occupancy status to the local controller as a binary-encoded variable. Therefore the wireless communication traffic is decreased and the RFID receiver is in a sleep mode unless an event occurred. This will induce savings in the consumed energy by RFID device which leads to increase the battery life time.

The neighbouring local-controllers communicate through message-passing their own actuation values in order for each controller to consider the expected light interference. To avoid heavy communication traffic, the communication is modelled as an event-driven that relies on the actuation update.

Among the wireless devices, the light sensor appears to be the most critical power-consumption device. This is due to the fact that it should send frequent updates to the local controller about the light luminance. Considering that the transmitting/receiving sampling rate is adaptable, depending on the local controller status, we modelled a mechanism to save power for light sensors. When the controller is in a stable state, it sends a request to the sensor for a decrease in its sampling rate, and when the controller goes to an unstable state, it requests increasing of the sampling rate in order to reach the stabilized phase more quickly. In case the sensor is using the stable sampling rate and the controller detects an unstable state, the controller will use the last received sample until the current sampling period is finished, and the sensor sampling rate is updated. Then, the controller can then receive the new sample.

IV. Charon Modelling of the Lighting System

In order to simulate the system and evaluate its performances, the lighting system and its environment have been modelled using the Charon toolset. In this section, the hybrid models for the PPD-Controller and the environments are explained.

A. Charon Modelling of the Controller

In the Charon modelling, one agent is used for the global controller, 2 other agents have been used to model the environments (external light and presence). For each zone, 4 agents are used: RFID, light sensor, blinding controller and light controller (local controller). As mentioned earlier, the global controller sets the configuration parameters for the local controllers, e.g. activate/deactivate some controllers (i.e. blinding controller) or some functions inside a controller (i.e. considering or not the blinding). The local controller contains 2 subagents, one is used to receive and calculate the light interferences coming from the adjacent zones, whereas the other one is used to send the actuation values and trigger the optimisation engine. Each agent contains a hierarchy of modes describing the corresponding behaviour, for example the local controller mode shown in Fig. 4, describes the behaviour of a local controller.

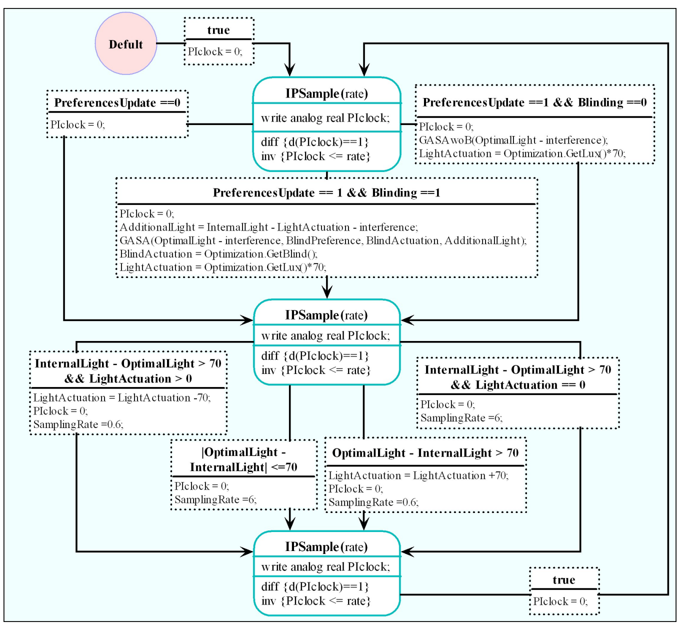


Fig. 4. Linear Hybrid Automata for the Local-Controller

B. Modelling of the Environment

There are two main environments for the lighting system, the daylight and the person movement environments. In order to verify the behaviour of the PPD-Controller, both environments have been modelled using hybrid systems, as the daylight model has continuous behaviour while the presence model has discrete behaviour.

In the daylight model shown in Fig. 5, five periods have been modeled as a first order differential equation with a constant slope (using linear hybrid automata [18]). During the first and last four hours of the day, the daylight slope and luminance are equal to zero, while during the second four hours the slope is equal to 100, which means that the maximum intensity in the day is 4000 Lux. In the next eight hours the slope is equal to zero and then goes to -100 in the following four hours, in order to reach zero luminance again at the end of the day. The light intensity that comes to the controlled zone is a percentage of the daylight intensity, this percentage relies on the dimensions of the window. In this model, 8% of the daylight is considered as the external light coming into the controlled zone [17].

The model for persons movement in the controlled zone follows a deterministic distribution with respect to the day time. In the first and last seven hours of the day, no one is in the zone, from 7:00 to 10:00 AM people arrive successively, then during the next seven hours enter or exit with a 50% probability, and finally, the next two hours people leave individually.

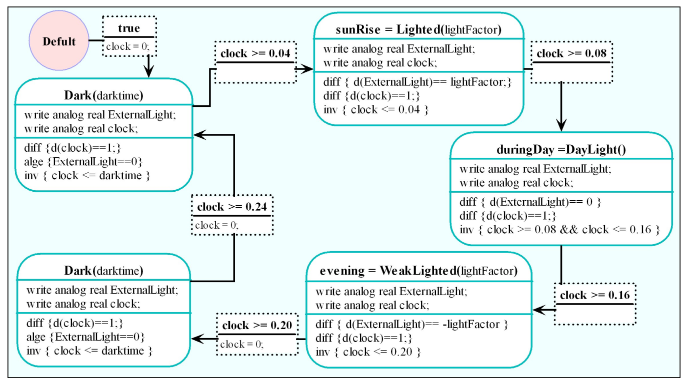


Fig. 5. Linear Hybrid Automata for the Daylight Environment

V. Distributed Optimisation Process

The control platform provides an optimisation mechanism that calculates the optimal artificial luminance level and blinding position. Each local controller includes an optimisation engine, which receives the Optimal Light (OL) and the Blinding Preference position (BF) provided by the preference solver, and sends back to the controller the optimal settings. We have chosen to use the Genetic Algorithm/Simulated Annealing (GASA) optimisation technique [19]. The stabilization of the overall optimisation process is guaranteed through a certain number of techniques including (1) linear interference predication, and (2) a scheduler for the global controller that synchronies local controllers.

A. GASA Optimisation Technique

Fig. 6 shows the GASA optimisation technique used to evaluate the artificial light luminance and blinding position in order to reach an intermediate optimal point between the user satisfaction and the energy consumption as follows:

1. Select randomly a predefined percentage (for example 10%) from the search population.
2. Calculate the cost function for each solution point.

3. Evaluate the Pareto points [20] and select the best two point which have the lowest cost function.
4. Apply the Genetic Algorithm (GA) on the two points selected previously to create a new solution point.
5. Apply the Simulated Annealing (SA) algorithm to the new population.
6. Evaluate the optimal point.
7. The algorithms stops when the calculated optimal point matches the stopping criteria, otherwise the calculated optimal point is attached to the new search population and the algorithm is repeated for another cycle.

The optimisation engine defines the User Discomfort (UD) as a function of the Blind Position Discomfort (BPD) and the Luminance Discomfort (LD). Whereas the Energy Consumption (EC) is defined as a function of the BLL and Energy Cost Factor (ECF) of the blind actuator. Blinding actuator's energy has been considered in the optimisation engine to avoid the frequent movement of the blinding which leads to user uncomforting. The corresponding metrics and equations are described below:

$$BPD = BF - BP$$
$$LD = OL - \text{ External Luminance } (EL) - BLL$$
$$UD = BPD + LD$$
$$EL = \text{Estimated Total } EL\ (ETEL) - (ETEL * BP)$$
$$ETEL = Current\,EL * (100\% - Current\,BP)$$
$$EC = BLL + ECF * (Current\,BP - BP)$$

The cost function is the sum of the optimisation metrics:

$$Cost\,Function(CF) = UD + EC \qquad (2)$$

The search space population contains all the possible values of the variables included in the system. Bulb Luminance Level (BLL) = {0%, 10%,:.,100%}, where 0% means that there is no light intensity and 100% is the maximum light intensity that comes from the bulb. Blinding Position (BP) = {0%, 10%,:.,100%}, where 0% means that the blinding is completely open and 100% is completely closed.

For the solution space, the optimisation engine considers all the visible solutions. Therefore, the Solution Space Population = 10 for the lighting × 10 for the blinding = 100 possible solutions.

Each solution point is a combination of two parts; each one is represented in binary format by 4 bits. The first part presents the 10 BLL possibilities and the second part is used for the 10 BP possibilities. In order to apply the GA to the best two points, the first point exchanges its BP (Blind1) part with the BLL (Lux2) part in the second point. Therefore, the improvement has been applied only on the BP in the first point and hence, the SA algorithm is applied on the BLL (Lux1) part of the best point (lowest cost function). In the SA algorithm, we consider 100% refuse for the generated point if it's cost higher than the ex-optimal one.

B. Distributed Optimisation Techniques

When considering distributed controllers with local optimisation engines, the problem of instability occurs since

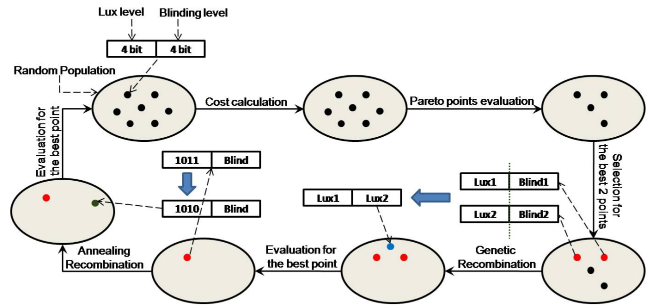


Fig. 6. GASA Optimisation Process

the decisions of the controllers are cyclical dependent. In order to avoid the control instability due to cyclic effect of interferences or at least reach a faster stable state, the following features have been used in our model:

Luminance Boundaries: In order to distribute the energy consumption over all the controlled zones, luminance boundaries have been set to limit the user's preferences of exceeding 700 Lux. This will also limit the interferences between zones.

Tuning Process: As explained earlier, the optimisation engine uses a random initial population to select the optimal setting. In order to improve the optimisation performance, the last optimal settings are added into the next search population. In this case, the optimal settings are tuned to obtain faster more accurate values. The controller is sensitive to 70 Lux margin corresponding to one dimming artificial light level. If the sensed value is more than 70 Lux different than the optimal one, the actuated light is decreased by one dimming level (70 Lux) instead of the exact Lux difference. This will diminish the interferences and then make faster the stabilization process.

Scheduling: The scheduling technique implemented in the global controller allows further improvement to overcome the instability due to the interferences. It follows the pseudo-code depicted in Fig. 7. It basically defines two sets of zones (S1, S2) where the zones of each set are interference-independent from the zones of the other set. S1 and S2 can then be executed concurrently. However this technique does not handle the potential initial instability cycle, and hence we have introduce the expected interference mechanism described next.

```
S = all zones (1 ... 10)
Z = Pick Randomly one zone form S;
Add (Z) to S1;
Add All zones dependant on Z to S2;

While ((S1 U S2) <= S)
{
    Si = All zones dependant on  S2;
    Add (Si) to S1;
    Sj = All zones dependant on  S1;
    Add (Sj) to S2;
}
Return (S1, S2);
```

Fig. 7. Pseudo-code for the Schedular

Expected Interference: In the first running cycle, a local controller does not have any information about the interferences that cause instability. To avoid this initial instability, an expected interference parameter is introduced using Linear Prediction Coding (LPC) algorithm. It is based on Weighted Least Square Error (WLSE) technique, the constant coefficients are calculated [21] using a specific equation. The 5^{th} order of the prediction filter polynomial has been considered in order to cover a week period (5 working days), moreover the first sample is considered as the average of the last week interferences. However in the initial running cycle for the overall system, the predictor does not have any value to start with, so it considers its own optimal value as it is the actuated value in the neighbour zone and then calculate the expected interference. Due to space limitations, the details of the equations are omitted in this paper, however are available in the internal report [22].

Fig. 8 shows an experimental test for the algorithm applied to a local controller for a month (20 working days). It is notable that the prediction error is always less than 70 Lux, which means that even considering the predicted values the controller will reach the correct actuation decision.

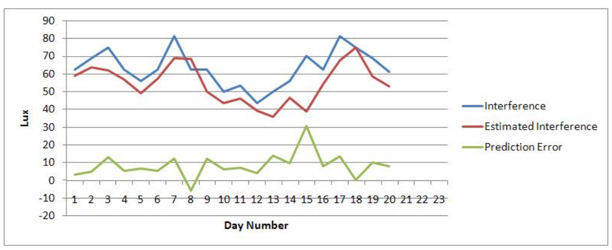
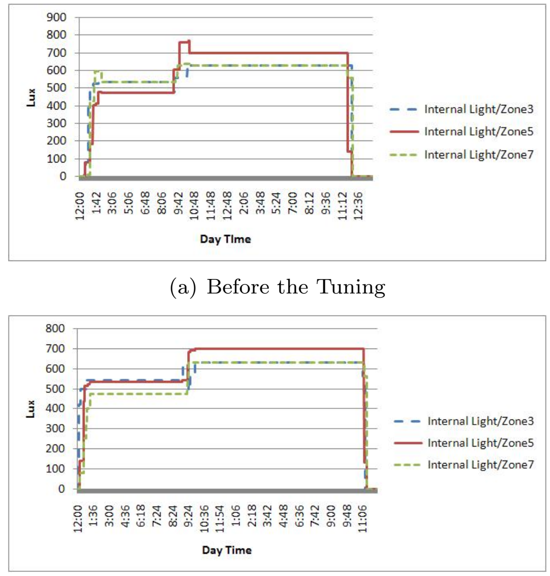


Fig. 8. Linear Prediction for the Interference During a Month

The simulation results show that due to the previous factors, the lighting controller gets stabilized after 2 cycles (maximum), however in [23] the system stability needs 100 cycles. Fig. 9(a) and Fig. 9(b) show the luminance changing in 3 neighbour zones before and after applying the aforementioned techniques, respectively.



(a) Before the Tuning

(b) After the Tuning

Fig. 9. Lighting Tuning Process

VI. Evaluation Study

In order to verify the modelling technique and show its similarity to the real environment, we compared the simulation of a 10-level dimming PI-Controller with a real scenario. The case study considers a single zone that contains an external light source (window), with 600 Lux as the maximum luminance that can be supplied and 350 Lux is the set point.

In order to evaluate the accuracy of the simulations models, we have compared our simulation results for the aforementioned model to a dataset from [17]. Fig. 10 shows light luminance variations for the experimental and the simulation model. Although we ignored several lighting factors, e.g. sky luminance distribution, window solar transmittance and visible reflectance of interior surfaces, the two curves reflect similar variations. This is mainly due to the control sensitivity, where 60 Lux sensitivity covers the influences of such factors.

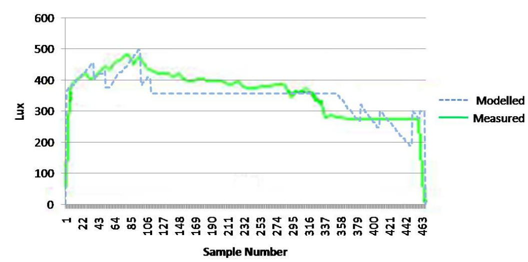


Fig. 10. Experimental Results Vs. Modelling Results

A. Lighting Baseline Models

In order to evaluate the potential improvement in power consumption, stability and response time, we have considered two control models as a baseline. These models are among the most popular control techniques, and have been applied to the same aforementioned scenario specification. In the first model scenario, called the Null model, the control strategy is based on user presence, where the controller switches the light on given a (fixed) predefined preference whenever a person is detected using Passive InfraRed (PIR) sensor. Moreover, the user can turn on/off the automatic lighting system for several zones, or for the entire system.

The second control model uses a standard PI-Controller [24] in order to predict the next actuation sample, depending on the accumulation of the pervious actuation errors, the daylight and the light interference.

B. Simulation Results

In this section we outline the simulation results for the PPD-Controller. Fig. 11 shows that the distributed control strategy has lower expected delay than the centralized one for the Null model.

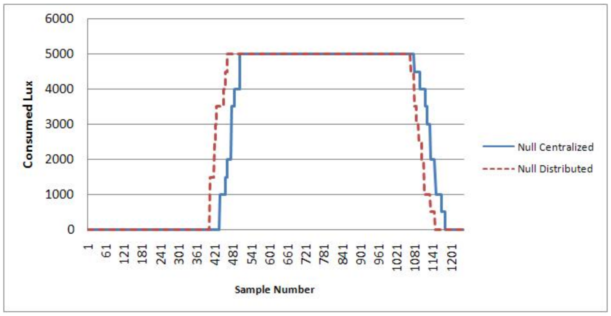
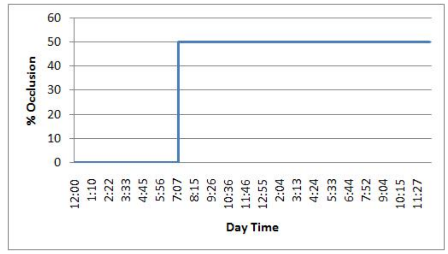


Fig. 11. Centralized Vs. Distributed Controller

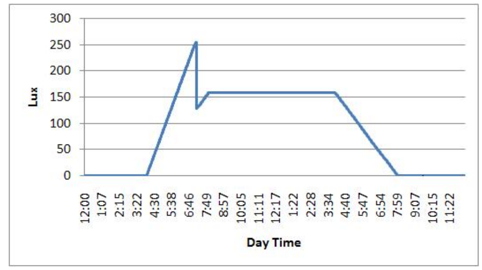
In the rest of the section we describe the simulation results for single and multiple zone(s) models, the energy consumption, and the WSAN performance evaluation.

Single Zone: In the single zone lighting control, we have considered a scenario of one controlled zone with one external light source (window) as a source of daylight luminance. To allow a clear comparison of the different results, we have fixed the preferences for all the persons inside the controlled area as 50% blinding occlusion and 560 Lux (500 Lux, European law UNI EN 12464). As shown in Fig. 12(a), the GASA optimisation engine selects the optimal blinding occlusion, which affects the external light coming into the controlled zone as depicted in Fig. 12(b). During the period from 12:00-7:00 AM, no person is in the controlled zone, and then the controller switches the artificial light off. In this case the light sensor detects only the external light intensity as internal light, the controller is then in a stable state and will request the WSAN to increase its sampling period to 12 min in order to save battery power, as shown in Fig. 12(d). When people start coming at 7:00 AM, the controller actuates the artificial light to 420 Lux and requests a faster sampling rate (1/6 min), which allows the controller to reach a stable setting faster. The controller considers 70 Lux as an acceptable difference margin between the sensed internal light and a given optimal light (calculated by the preference solver). If this margin exceeds 70 Lux, the controller updates the artificial light as illustrated in Fig. 12(c), where the artificial actuation is increased to 490 Lux when (at 7:00 PM) the external light decreased to make the margin exceed 70 Lux.

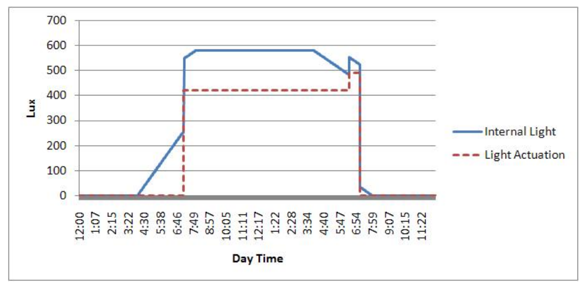
Multiple Zones: Unlike the single-zone model, the multi-zone controller considers the light interference between the different controlled zones. Fig. 13 shows the simulation results for zone 1 and zone 3. These two zones have been chosen for illustrative purposes; zone 1 has a window that provides external light, whereas zone 3 is an internal zone and it is affected by the light interference coming from zone 1 (Fig. 2). Based on the WSAN evaluation of the sampling interval for the light sensors, the minimum periods are: 36 sec in the unstable state and 6 min for the stable state. It is obvious that the internal light in zone 3 is more stable than in zone 1 which indicates an advantage for the sensor's battery power consumption by increasing its sleeping period. From the optimisation side,



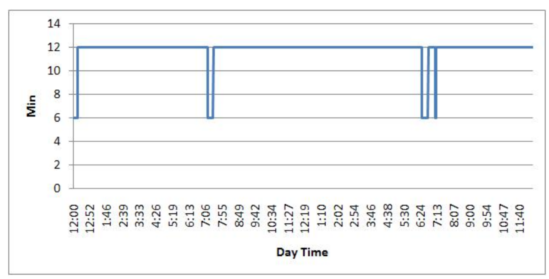
(a) Blinding Actuation



(b) External Light coming to Zone1



(c) Artificial Light Actuation & Internal Light

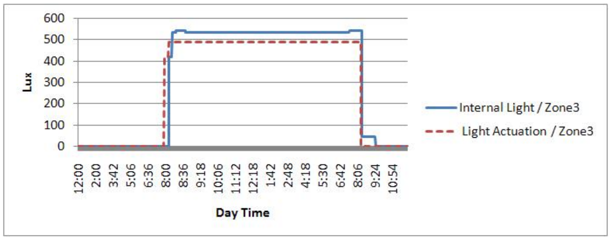


(d) Light Sensor Sampling Intervals

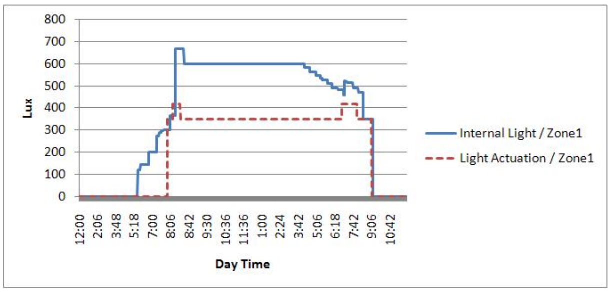
Fig. 12. Lighting Control for a Single Zone

the GASA optimisation engine is used to select a meta-optimal point between the energy consumption and the user comfort. As notable in Fig. 13(c), the GASA optimisation engine gives 40% blinding occlusion while the user requested 50%.

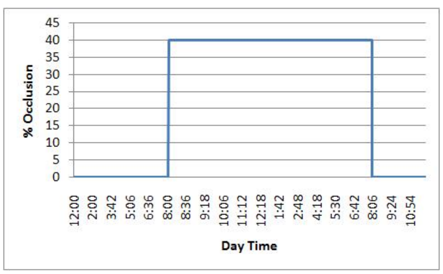
Energy Saving: In order to compare between different control strategies, we have used the luminance consumption in Lux as the energy consumption metric. Fig. 14 shows the summing of Lux consumed over time in all the zones for Null, PI-Controller and PPD-Controller strategies using a constant user preference (500 Lux). This primary test concludes that the PI-Controller improves energy consumption by 29% over the Null strategy; however PPD-Controller shows a 32% improvement, and hence the PPD-Controller improves 3.1% comparing to the PI-Controller for one time change in the user preference, as



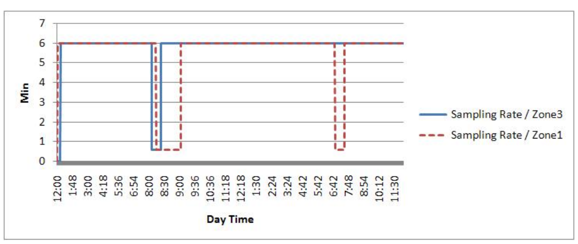
(a) Artificial Light Actuation & Internal Light/Zone3



(b) Artificial Light Actuation & Internal Light/Zone1



(c) Blinding Actuation/Zone1



(d) Light Sensor Sampling Intervals/Zone1&3

Fig. 13. Lighting Control for Multi Zones

shown in Fig. 16. In order to evaluate the optimisation engine, we consider that the minmal number of user preference changes per day is three (650, 500, 700 Lux), Fig. 15 shows the energy consumption for each strategy; the PI-Controller and PPD-Controller show 23% and 32% improvement, respectively, as shown in Fig. 16. We conclude that the optimisation engine saves nearly 3% of the energy consumption for each execution.

C. WSAN Performance Evaluation Results

In addition to the Hybrid/Multi-agent model explained earlier, the associated embedded Java code has been emulated using the Sun-Spot emulator [7]. The results produced by the emulator are matched to the simulation results obtained using the Charon simulator. We have also attempted to use the emulator to evaluate the network's performance; we found the tool to be inappropriate for such test, since it is dedicated to development, debugging and testing. For this reason we evaluate the WSAN QoS using a more appropriate tool, VisualSense.

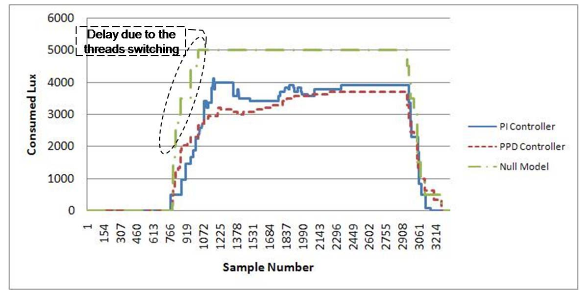


Fig. 14. Energy Consumption Using Fixed User Preference

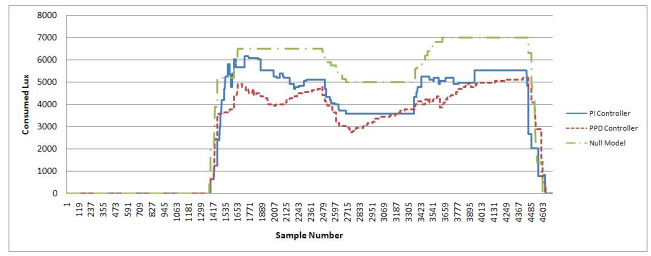


Fig. 15. Energy Consumption Using Variable User Preference

The QoS network performance metrics that we consider include buffer size, time response, packet loss (caused by packet collision), controller/receiver duty cycle, channel throughput and sensor's battery life time. Due to space limitations we show only the main results of the evaluation; a detailed description of the study is described in [22].

Table I provides the evaluation results for each model during 100 samples (100 minutes). It clearly shows that the PPD approach performs better than the centralized controller. The centralized controller has a higher collision probability in comparison to the PPD; moreover, it needs more memory to save the received requests, which leads to high controller duty-cycle and low time-response. This is due to the delay that can reach 287 minutes (479 samples and 0.6 min for service time) to serve the next request under a no drop-out strategy [25]. In relation to the battery lifetime for the sensors, all models have the same expected lifetime because of the fixed sampling rate for the sensors.

VII. Summary and Conclusion

This article described a model-based distributed controller for lighting systems, called a parameterizable and predictable distributed (PPD) controller. The parameterizable capability has been implemented through assigning global parameters, which alter the behaviours of the local controllers. The PPD-Controller incorporates an optimisation engine to compute the optimal settings for increased energy-efficient control. The local optimisations are coordinated to achieve a level of global optimlity, using some features and heuristics to guarantee better control stability. These features enable us to overcome the potential instability in our lighting model due to the limited interference of lighting levels across the zones. However, for

TABLE I
WSAN QoS

	Single Zone PPD	Multiple Zones PPD	Centralized Controller	Improvement
Packet Loss	0%	4%	8.6%	$\sim$ 53-100%
Buffer Size	5 packets	9 packets	479 packets	$\sim$ 98%
Controller Duty Cycle	35%	66%	100%	$\sim$ 34-65%
Response Time (after 100 samples)	1.8 minute	3 minute	287 minutes	$\sim$ 98-99%
Channel Throughput	0.58 packet/min	1.1 packet/min	6.46 packet/min	$\sim$ 82-91%
Battery Life Time	79.72 days	79.72 days	79.72 days	0%

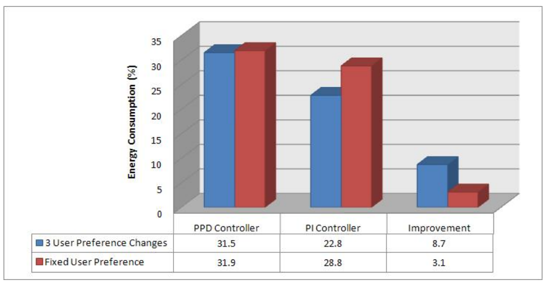


Fig. 16. Energy Saving

more interference-sensitive systems like HVAC, more sophisticated techniques are necessary .

As future work, we intend to implement a demonstration of the developed system in an actual building, the Environmental Research Institute (ERI) building, which is the ITOBO "Living Laboratory" [26]. We also intend to adapt this work to a more complex scenario including HVAC control.

Acknowledgment

This work was funded by Science Foundation Ireland (SFI) grant 06-SRC-I1091.

References

[1] "Scenarios for a clean energy future: Interlaboratory working group on energy-efficient and clean-energy technologies," The Interlaboratory Working Group on Energy-Efficient and Clean-Energy, 2000, NREL/TP-620-29379; ORNL/CON-476; LBNL-44029.

[2] S. Furubo H. Ban M. Nakamura, A. Sakurai, "Collaborative processing in mote-based sensor/actuator networks for environment control application," *Signal Processing, Elsevier*, 2008.

[3] S. Megerian Y. Lin, "Low cost distributed actuation in largescale ad hoc sensor-actuator networks," *International Conference on Wireless Networks, Communications and Mobile Computing*, pp. 975–980, 2005.

[4] I. Jacobson G. Booch, J. Rumbaugh, "The unified modeling language user guide," *Addison Wesley*, 1998.

[5] G. Provan A. Mady, M. Boubekeur, "Compositional model-driven design of embedded code for energy-efficient buildings," 2009, 7^{th} IEEE International Conference on Industrial Informatics (INDIN 2009), Cardiff, UK.

[6] F.Mattern K. Romer, O. Kasten, "Middleware challenges for wireless sensor networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, 2002.

[7] "Sunspot," Website, http://www.sunspotworld.com/SPOTManager/.

[8] IH. Kasimoglu IF. Akyildiz, "Wireless sensor and actor networks: research challenges," *Ad hoc networks, Elsevier*, 2004.

[9] "The ptolemy project," Website, http://ptolemy.eecs.berkeley.edu/.

[10] L. Soethout D. van Paassen A. Yahiaoui, J.L.M. Hensen, "Simulation based design environment for multi-agent systems in buildings," *7th International Conference on System Simulation in Buildings, Belgium*, 2006.

[11] L. Soethout D. van Paassen A. Yahiaoui, J.L.M. Hensen, "Developing web-services for distributed control and building performance simulation using run-time coupling," *Proceedings of the 10th IBPSA Building Simulation Conference*, 2007.

[12] S. Ploix M. Jacomino S. Abras, S. Pesty, "An anticipation mechanism for power management in a smart home using multi-agent systems," *Information and Communication Technologies: 3rd International Conference on From Theory to Applications (ICTTA)*, pp. 1–6, 2008.

[13] Ardeshir Mahdavi, "Predictive simulation-based lighting and shading systems control in buildings," *Tsinghua Press and Springer-Verlag*, 2008.

[14] M.van Steen A. S. Tanenbaum, "Distributed systems: Principles and paradigms," *Prentice Hall PTR*, 2001.

[15] D. Nelson, "Electric lighting controls," *Whole Building Design Guide (WBDG)*, 2008.

[16] ST Microelectronic, "Universal motor speed control and light dimmer with triac and st7lite microcontroller," *Application note*, 2007.

[17] G. Stavrakakis C. Lazos D. Kolokotsa, A. Pouliezos, "Predictive control techniques for energy and indoor environmental quality management in buildings," *Building and Environment*, 2008.

[18] TA. Henzinger, "The theory of hybrid automata," *Proc. 11th Annual IEEE Symposium on Logic in Computer Science (LICS 96)*, pp. 278–292, 1996.

[19] A. Abraham H. Al-Qaheri M.A. El-Hosseini, A.E. Hassanien, "Genetic annealing optimization: Design and real world applications," *Proc. of Eighth International Conference on Intelligent Systems Design and Applications (ISDA'08)*, 2008.

[20] L. Thiele E. Zitzler, "An evolutionary algorithm for multiobjective optimization: The strength pareto approach," *Swiss Federal Institute of Technology, TIK-Report*, 1998.

[21] M Garg, "Linear prediction algorithms," *Indian Institute of technology, Bombay, India*, 2003.

[22] G. Provan A. Mady, M. Boubekeur, "Optimised embedded distributed controller for automated lighting systems," *technical report, Cork Complex Systems Lab (CCSL),University College Cork, Ireland*, 2010.

[23] K. Imazato M. Miki, T. Hiroyasu, "Proposal for an intelligent lighting system, and verification of control method effectiveness," *IEEE Conference on Cybernetics and Intelligent Systems*, 2004.

[24] Douglas J. Cooper, "Practical process control, proven methods and best practices for automatic pid control," *e-textbook*, 2008.

[25] O. Schelen U. Bodin, "Drop strategies and loss-rate differentiation," *Ninth International Conference on Network Protocols*, 2001.

[26] "Environmental research institute," Website, http://www.ucc.ie/en/ERI/.

Alie El-Din Mady was born in Egypt, in 1984. He received the B.Sc. degree in electronics and communication engineering with honor degree from Alexandria University, Alexandria, Egypt, in 2006, and the M.Sc. degree in embedded system design from Università dellà Svizzera italiana, Advanced Learning and Research Institute (ALaRI), Lugano, Switzerland, in 2008. He is currently a Ph.D. researcher at Cork Complex Systems Laboratory (CCSL), University College Cork (UCC), Cork, Ireland, where he is researching aspects of embedded middleware for optimised building operations under the supervision of Prof. Gregory Provan and Dr. Menouer Boubekeur.

Menouer Boubekeur received his Ph.D. degree in computer science from the University of Joseph Fourier, Grenoble, France in 2004. He obtained his engineering degree in computer science in 1997 from the University of Es-Senia (Oran, Algeria). From 1997 to 1999 he worked as a computer science engineer at ADEM Info (Private company in Oran). He Joined IMAG institute in Grenoble where he obtained his Master degree in computer sciences in 2000. Later, he joined the VDS group of TIMA Laboratory in the University of Joseph Fourier where he obtained his PhD Degree. After being a post-doc in CEOL centre working in real-time systems analysis, he joined CCSL as a researcher in 2008. His main research interests are real time and embedded systems, modeling and formal verification methods, smart buildings and integrated software environment.

Gregory Provan is a Professor at the Computer Science Department at University College Cork (UCC), in Cork, Ireland. He is the Director of the Complex Systems Lab (http://www.cs.ucc.ie/ccsl/) at UCC. He received a BSE from Princeton University, an MSc from Stanford University, and a DPhil from Oxford University. Prior to his appointment at UCC, he spent 10 years as technical manager at Rockwell Scientific, leading the Autonomous Systems group. His research interests include model-based diagnostics, control, large-scale systems modeling, model transformation, component-based model development, embedded systems and embedded code generation.