Thermal Inertia: Towards An Energy Conservation Room Management System (Technical report)

Yi Yuan*, Dawei Pan†, Dan Wang*, Xiaohua Xu‡,Yu Peng†,Xiyuan Peng†, Peng-Jun Wan‡
*The Hong Kong Polytechnic University,†Harbin Institute of Technology, ‡Illinois Institute of Technology

***Abstract*—We are in an age where people are paying increasing attention to energy conservation around the world. The heating and air-conditioning systems of buildings introduce one of the largest chunk of energy expenses. In this paper, we make a key observation that after a meeting or a class ends in a room, the indoor temperature will not immediately increase to the outdoor temperature. We call this phenomenon *Thermal Inertia*. Thus, if we arrange subsequent meetings in the same room; than a room that has not been used for some time, we can take advantage of such un-dissipated cool or heated air and conserve energy.**

We develop a green room management system with three main components. First, it has a wireless sensor network to collect indoor, outdoor temperature and electricity expenses of the heating or air-conditioning devices. Second, we build an energy-temperature correlation model for the energy expenses and the corresponding room temperature. Third, we develop room scheduling algorithms. In detail, we first extend the current sensor hardware so that it can record the electricity expenses in re-heating or re-cooling a room. As the sensor network needs to work unattendedly, we develop a hardware board for long range communications so that the Imote2 can send data to a remote server without a computer-relay close-by. An efficient two tiered sensor network is developed with our extended Imote2 and TelosB sensors. We apply laws of thermodynamics and build a correlation model of the energy needed to re-cooling a room to a target temperature. Such model requires parameter calibration, and uses the data collected from the sensor network for model refinement. Armed with the energy-temperature correlation model, we formally describe the problem for finding the minimum energy schedule. We prove our problem is NP-complete in general case. We develop an optimal algorithm for special case and two fast heuristics for general cases.

Our system is validated with real deployment of a sensor network for data collection and thermodynamics model calibration. We conduct a comprehensive evaluation with synthetic room and meeting configurations; as well as a real class schedules and classroom topologies of The Hong Kong Polytechnic University, academic calendar year of Spring 2011. We observe a 20% energy saving as compared with the current schedules.

I. Introduction

There is a huge interest in building a green world recently. The key focus is energy conservation and energy efficiency. Computer scientists are actively contributing our effort in two directions, 1) improve energy efficiency of computing systems, and 2) apply computing systems (e.g., sensor networks) for energy conservation in broader disciplines.

For the first category, many studies are working on energy efficiency for data centers [16][17][19], a top energy consumer among all computing devices. While the energy expenses of computing industry is increasing fast in recent years, the largest portion of energy consumption is still dominated by such areas as commercial buildings, residential usage, transportation, manufactory industry [22]. Especially, for regions where the Industrial sector is small, the electricity consumption by commercial buildings can be more dominating; for example in Hong Kong, 65% of electricity in 2008 goes to the commercial sector [5].

The heating and air conditioning of commercial buildings has the largest chunk in energy expenses. In 2008 the Office Segment of Hong Kong, 54% electricity goes to space conditioning (i.e., air-conditioning), 14% goes to lighting, 13% goes to office equipments such as computers [5]. Monitoring the conditions of the buildings and efficient utilization of heating, ventilation, and air conditioning (HVAC) has been a long time topic; and advanced commercial buildings can automatically turn off lights and HVAC systems of rooms when humans are not in presence. Nevertheless, we notice that even if the heating or air-conditioning of a room is turned off, the heat or the cool air will not immediately dissipate. We call this phenomenon *Thermal Inertia*.[1] We consider the un-dissipated cool or heated air a valuable resource that can be utilized, so that future usage of this room can take the advantage without re-heating or re-cooling the room.

Based on this observation, we develop an energy conservation room management system, such that the allocation of the rooms of a building (or classrooms in campus) is based not only on a schedule (e.g., meeting time, room capacity), but also on the existing heating or air-conditioning conditions of the rooms. In the rest of the paper, we will only use air-conditioning as an example to ease our presentation.

Clearly, our room management system falls into an optimization problem. It is not straightforward, however to know how much energy will be saved if a room is scheduled. As an example, the recommended office temperature in Hong Kong is 26°C (79°F). Assume a room was used 20 minutes ago, and its current temperature is 29°C (84°F). The outdoor temperature is 37°C (99°F). If we schedule a meeting 5 minutes later in this room, how much electricity is needed to re-cool it to the targeted temperature 26°C (79°F)?

This is affected by such factors as the room specifics (size, wall materials, etc), indoor and outdoor temperature, the targeted temperature etc. A key difficulty is to build a correlation among these factors. The more accurate this correlation model is, the better the scheduling algorithm we can run on top of it. Building this model does not solely fall into the computer science domain. Advanced thermodynamics theories may be needed. We believe that in the sensor network research today,

[1]This name follows a recommendation from a senior practitioner and researcher from Building and Service Engineering.

it is very common that cross-discipline understandings are required; for example, it is shown that knowledge on sensor placement quality in the sense of civil engineering can make the structural health monitoring system built by computer scientists more plausible [8]. A careful management on the degree of understanding on different disciplines is very important. In our work, we choose to apply rudimental thermodynamics theory to build a simple initial energy-temperature model. We then use sensor data to calibrate this model. We validate the effectiveness of such design by a real experiment.

Another difficulty is that we do not have off-the-shelf components for our sensor network. We thus extend Imote2 to an electricity-meter so that it can record electricity usage of air-conditioners. As we expect that our system should work in public unattendedly for a period of time, we develop a board for long range communication so that Imote2 can send data directly to a remote server. As such, no laptop computer, which may be easily stolen, needs to be placed as a relay close-by. We develop a two tier sensor network with TelosB (as temperature sensor) and Imote2-based electricity-meter. Such system is flexible and efficient (the Imote2-based electricity-meter does not have energy constraint).

On top of these, we develop room scheduling algorithms. We first formally describe the problem for finding a minimum-energy schedule. We prove that our problem is NP-complete in general cases. Then we develop an optimal algorithm for a special case where all rooms are equal. For the general case, we develop two efficient heuristics.

Besides a real world system deployment for model validation and data collection, we evaluate our system with comprehensive simulations with synthetic room configurations and meeting schedules. We also evaluate our algorithms with real class schedules and classroom topologies of The Hong Kong Polytechnic University, academic calendar year of Spring 2011. We observe that we can save 20% of electricity as compared to the current room schedules of The Hong Kong Polytechnic University, and even higher for synthetic data.

II. Room Management System: An Overview

We discuss some high level system choices. As a first work, we confine our study that given the schedules, how the classes/meetings should be arranged. We leave a detailed investigation of online room management as future work.

To accurately schedule rooms and maximally conserve energy, an important part of our system is that we need to build an *energy-temperature correlation model* so that the room scheduling algorithm can run on top of it. More specifically, we need a function such that given the current temperature and room environment configurations, the energy to be consumed to achieve the target temperature. There are two extreme ways for building such model. First we can apply advanced thermodynamics theories and material sciences to explicitly compute such function. Second, we can build a database with entries of the environment parameters (e.g., indoor temperature, outdoor temperature, and targeted temperature) and the corresponding energy consumptions. In the room scheduling algorithm, whenever an estimation on the energy expenses is needed, an entry in this database that has the most similar environmental configuration can be extracted.

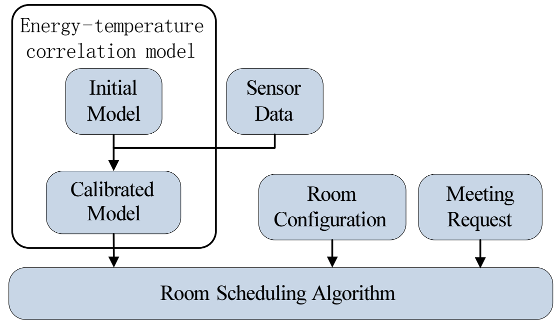

Fig. 1: The framework of the room management system.

The first choice falls into the expertise of Building and Service Engineering. We have consulted experts of BSE from both academia and industry. While there are sophisticated tools such as EnergyPlus [2], they admit that it is difficult to build a model purely from theory. For the second choice, to build the correlation database, a sensor network can be deployed to collect such data as temperature and energy expenses. The accuracy depends on the granularity of the data collection. The more samples the database has, the more accurate to find the energy expenses with a similar environmental configuration. After some studies on physical laws on heat conduction and some field experimental validation, our choice finally falls into a mixture of the two extremes. We use an initial model following rudimental Fourier's law of heat conduction. In this model, some parameters are difficult to compute from theory. These parameters are invariants, however, e.g., only affected by the materials of the room. Thus we inversely calibrate the parameters of this model using the data collected by a sensor network. The high-level framework of our system is in Fig. 1.

We also want to clarify that in this paper, we use electricity expenses as our optimization objective. For end-users, having their electricity bills cut directly means money saving.

The remaining part of the paper proceeds as follows. In section III, we present our design of the sensor network. Section IV is devoted to our energy-temperature correlation model and a real world experiment validation. We detail our room scheduling algorithms in section V. In section VI, we evaluate our algorithm comprehensively. We present related work in section VII and section VIII concludes the paper.

III. Sensor Network Design

For a building, or a campus, there are multiple rooms. For each room, we need to build an energy-temperature correlation model (detail in Section IV) to be used for the scheduling algorithm (details in Section V). As such, a sensor network should be deployed in each room. In this sensor network, there should be a sensor to record electricity usage to air-conditioning the room. We also need to record the temperature. As the temperature in different locations of the room may not be uniform, a set of temperature sensors is suggested. We would like to comment that the sensor network is only used for

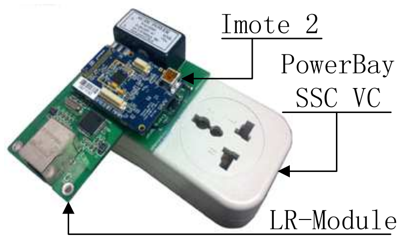

Fig. 2: An Electricity-Meter

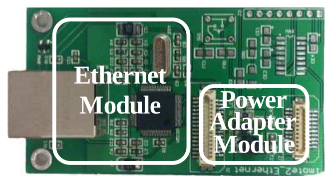

Fig. 3: A Long-range Data Communication Module for Imote2

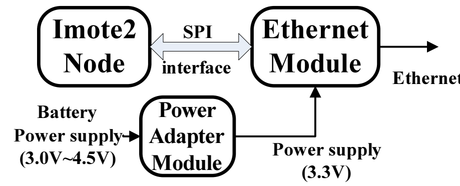

Fig. 4: Flow Chart of LR-module

the construction of the energy-temperature correlation model for each room. After the model is built, we can predict the energy consumption using the model.

Since the sensor network needed in each room is the same, in practice, we can deploy a sensor network and build the energy-temperature correlation model room-by-room.

Our system needs to work unattendedly in a building for a period of time. The sensors can usually be protected by a cover and placed on walls, roofs etc. However, it is impossible to place a laptop computer (as a base station) unattendedly. The rooms are public and the laptop computer can be stolen. This is in contrast to some smart home systems, where we can assume that the laptop/desktop computer will work in a private apartment. As the sensor network is deployed in buildings, power is not as critical as those applications in the wild.

For some functions we need, there is no off-the-shelf components. Before discussing the implementation of our sensor network, we extend the hardware and build an electricity-meter and a long-range data communication module as follows.

A. Design of an Electricity-meter

Our system needs to estimate the energy consumption for air-conditioning the room to a targeted temperature. We extend Imote2 with a PowerBay SSC VC to record electricity current (see Fig. 2). PowerBay SSC VC also becomes a power supply to Imote2. In operation, PowerBay SSC VC will record the power (in Watt) and such data will be digitized and output to Imote2. The data can then be transmitted out by Imote2.

B. A Long-range Data Communication Module for Imote2

We develop a long-range high-rate data communication module (LR-module) for Imote2 (See Fig. 3). This LR-module performs a series of transformation to convert Imote2 output data to the Ethernet port. The flow chart of our design is shown in Fig.4.

LR-module is a separate board which can be connected with the Imote2 node through the basic-connector, and it is directly controlled by the Imote2 node. LR-module chooses an SPI interface as the communication channel with Imote2 node. It has a bit-rate ranging from 6.3Kbps to 13Mbps. This makes our design bypass the throughput bottleneck. LR-module has the following two hardware components:

1) An ethernet module: LR-module applies W5100 as the network chip. W5100 integrates a hardware network stack, offering 4 separate socket interfaces. After simple initialization, developers can use TCP connection or send UDP packet

Fig. 5: The Sensor System. Here we present our enhanced Imote2 node and 3 TelosB temperature sensors.

the same as other applications. This choice avoid the high complexity of the network stack and network card driver for the operation system designers, especially for a simple OS like TinyOS. This makes the system simple and stable.

2) A power adapter module: The power adapter module change the battery power supply (3.0V - 4.5V) to a stable 3.3V power supply which is required by our hardware network stack module. We use LTC3429 Micropower Synchronous Boost Converter as the kernel chip for this module. LTC3429 has an up to 96 power conversion efficiency, leading to less power loss during conversion.

Equipped with the LR-module, the data can be sent to a remote server, e.g., in practice, we use 3G. Note that the choice of 3G is not special. It is possible to develop a module that use GPRS or WiFi for data transmission. We use 3G as it is more universally applicable than WiFi and has a greater transmission rate than GPRS. In our experiment, the effective data stream throughput of our module can reach 520K bps.

C. Development of Sensor Network

We show the design of our sensor network by integrating these components. We develop a two tiered sensor network. The first tier is a set of enhanced Imote2-based electricity-meters. The second tier is a set of TelosB-based temperature sensors (see Fig. 5). For the first tier, an electricity-meter monitors the electricity usage of the air-conditioner. It is also equipped with the LR-module and can communicate with a remote server. The Imote2-based electricity-meter is powered by alternating current and is thus not energy constrained.

For the second tier, we deploy a few indoor and outdoor temperature sensors. We use TelosB, as it is cheaper. To have better flexibility, in practice these temperature sensors can use batteries. TelosB is more energy efficient than Imote2.

The routing architecture of our sensor network is from the temperature sensors to the electricity-meter (one hop). We implement our sensor system in TinyOS, and use Collect Tree Protocol (CTP) [3] for data routing among sensor nodes. The electricity-meter then send these temperature data and its electricity readings to a remote server directly (one hop but long range data communication).

The lifetime of our sensor system is determined by TelosB nodes if they use battery power. In practice, every node gets the temperature and transmits 32 bytes every 10 seconds; The projected lifetime of our sensor network can thus reach 2000 hours. We find that this is far enough for us to collect data and calibrate the energy-temperature correlation model.

IV. Design of Energy-Temperature Correlation Model and Experimental Validation

In this section, we develop a model where the electricity is a function of (a room, current indoor/outdoor temperature, targeted temperature). Our idea is as follows. With our sensor network, we can measure the electricity usage, the indoor and outdoor temperatures and we know the targeted temperature in advance (in Hong Kong, it is 26°C as recommended by the administration). If there is an artificial perfect room with six identical walls with the same conductivity, we can easily build the energy-temperature correlation model for this room from Fourier's law of heat conduction [11]. We do not have a perfect room, however. Materials, shape and conductivities of the six walls (i.e., four side walls, a ceiling and a floor) are all different. Our key observation is that these factors are invariants. They are determined by their physical materials and do not change (or change ignorably) with outside factors.

Therefore, for each real-world room, we can build a virtual perfect room to mimic it. For this virtual perfect room, we build an energy-temperature correlation model using Fourier's law of heat conduction with the set of invariants undetermined. To compute these invariants, we collect a set of electricity and temperature data by our sensor network. We then inversely derive these invariants. After fitting these invariants back to the model, we can compute (or predict) electricity usage under any indoor/outdoor temperature and targeted temperature for this room. That is, we have our model.

In what follows, we first show the details of the development of our model. Then we will conduct a real-world experiment to validate our method.

A. Energy-Temperature Correlation Model

As explained, we use a virtual perfect room where 1) The room space is enclosed, i.e., no air exchange with other spaces; 2) All walls, ceiling and floor are made of materials with the same thermal conductivity and have identical thickness; 3) All outside temperature of the room is same and is constant. We show that for any real-world room with different shape and different materials, we can build a virtual perfect room with uniformed parameters to emulate it.

We also assume 1) the electrical power P of the air conditioner is constant when it is in operation, and is zero if it stops; and 2) the electricity-energy transformation rate r is a constant; this indicates the energy injected into a room per unit time when an air-conditioner is in operation is constant. We will also calibrate the condition of the air-conditioner.

Notation	Definition	Unit
T	Indoor temperature	K or $^\circ C$
P	Electrical power of the air conditioner	J/s
r	Energy transformation ratio of the air-conditioner	–
P_e	Effective energy injected to the air of the room per second, $P_e = r \times P$	J/s
T_o	Temperature outside the room	K or $^\circ C$
k	Thermal conductivity of a material	$W/(K \cdot m)$
L	Thickness of a material	m
A	Total area of six walls	m^2
m	Mass of the air in the room	kg
C	Specific heat capacity of air	$J/(kg \cdot K)$
Q	Heat transfer rate from outdoor to the room	J/s
λ	Conductivity of the room	$J/(s \cdot K)$

Let T be the indoor temperature. Let T_o be the temperature outside the room. Let Q be the heat transfer rate from outdoor to the room. Let k be the thermal conductivity of the material. Let A be the total area of the six walls. Let L be the thickness of a material. According to Fourier's law [11], we have

$$Q = \frac{kA}{L}(T_o - T) \tag{1}$$

Eq. 1 basically says that the heat transfer rate is proportional to thermal conductivity of the material, the size of the walls, the temperature difference and is inversely proportional to the thickness of a material.

Let P_e be the effective energy injected to the air of the room in every second. Let m be the mass of the air of the room. Let C be the heat capacity of the air of the room. In other words, C is the energy needed for one kilogram of a specific material (in our context, the air) to increase one degree of Celsius. The temperature changing rate $\frac{dT}{dt}$ of the room is [18]:

$$\frac{dT}{dt} = \frac{Q + P_e}{mC} \tag{2}$$

Let $\lambda = \frac{kA}{L}$. We say λ as the conductivity of this specific room. Combining Eq. 1 and Eq. 2, we obtain the following function for indoor temperature change:

$$T(t) = T_o + P_e \times \frac{1}{\lambda} + C_0 e^{-\frac{\lambda}{mC}t} \tag{3}$$

Here C_0 is an initialization parameter determined by $T(0)$, the temperature at time 0:

$$C_0 = \begin{cases} T(0) - T_o; & \text{air-conditioner not in operation} \\ T(0) - T_o - \frac{P_e}{\lambda}; & \text{air-conditioner in operation} \end{cases} \tag{4}$$

The energy-temperature correlation model is Eq. 3 and Eq. 4. As said, we do not compute λ from theory since this is difficult. We consider λ as an invariant, because it is related to the physical properties of the materials. Therefore, we calibrate this parameter by sensor data. We also calibrate T_o. We emphasize that T_o is artificial that approximates the overall outdoor situation of all walls. Though one wall may have a bigger change in outdoor temperature, T_o does not change abruptly. We will show that this is true in our experiments.

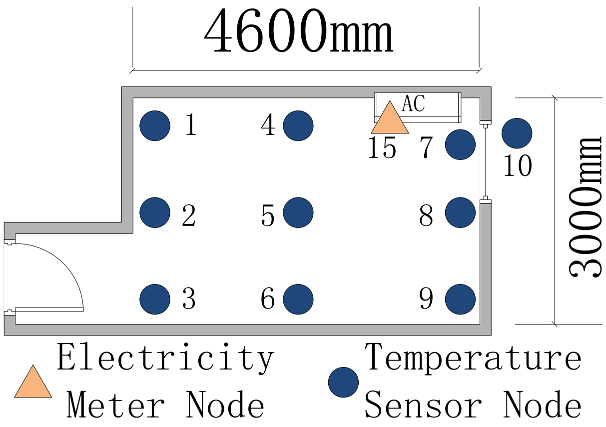

Fig. 6: Experiment Environment

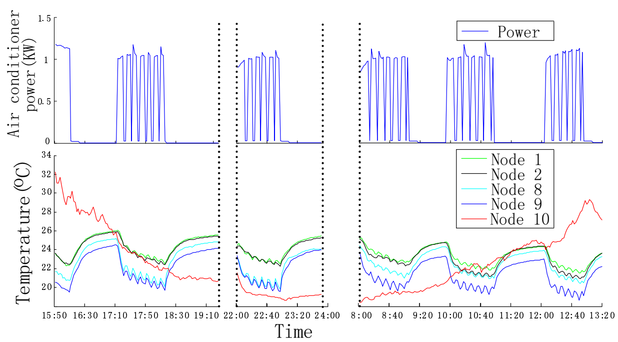

Fig. 7: Experiment Results

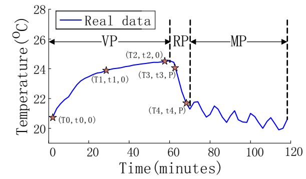

Fig. 8: Data selection for IndividualCal()

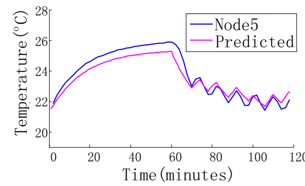

Fig. 9: Predicted T vs. temperature data of Node 5.

Besides the room parameters, we also need to know P_e, or r. It is easy to get P, i.e., the electricity consumption of the air-conditioner by our electricity-meter. P_e, the effective energy injected into the room every second, however, depends on the quality of individual air-conditioner. The effectiveness of a single air-conditioner will not change abruptly. Therefore, we calibrate r using the data collected by our sensor network.

In summary, we will use the sensor data to inversely compute the invariants λ, r and semi-invariant T_o. We use $\hat{\lambda}$, $\hat{T_o}$ and $\hat{r}$ to denote them. Then we fit $\hat{\lambda}$, $\hat{T_o}$ and $\hat{r}$ back into Eq. 3 and Eq. 4 (our energy-temperature correlation model). When future prediction is needed, we use Eq. 3 and Eq. 4 with $\hat{\lambda}$, $\hat{T_o}$ and $\hat{r}$. The details of the calibration is as follows.

Algorithm 1 IndividualCal()

Input: 1) 3 (T, t, P) from VP, 2) 2 (T, t, P) from RP, 3) mC
Output: $\hat{\lambda}_i$, $\hat{T}_{oi}$ and $\hat{r}_i$ for Node i

1: Set $P_e = 0$;
2: Use 3 data points of VP to construct 3 equations according to Eq.3; // We need 3 equations to solve 3 variables $\hat{\lambda}_i$, $\hat{T}_{oi}$ and C_0 of VP;
3: Compute $\hat{\lambda}_i$, $\hat{T}_{oi}$ and C_0 of VP;
4: Set $P_e = r \times P$;
5: Use 2 data points of RP and $\hat{\lambda}_i$, $\hat{T}_{oi}$ to construct 2 equations according to Eq.3 // We need 2 equations to solve 2 variables $\hat{r}$ and C_0 of RP;
6: Compute $\hat{r}$ and C_0 of RP;

The operation of a room can be cut into three periods: 1) the vacancy period (VP); 2) the re-cooling period (RP); and 3) the maintaining period (MP). The energy-temperature function of VP and RP are different (see the two phases of Eq. 3 and Eq. 4). MP is a combination of short periods of VP and RP.

Through the sensor network, we will collect temperature sequence $T_{i1}, T_{i2}, \ldots$, for each sensor node i and electricity sequence $P_1, P_2, \ldots,$. For each of these sequences, we identify periods of VPs, RPs and MPs. For each node i, we then apply Algorithm IndividualCal() to calibrate $\hat{\lambda}_i$, $\hat{T}_{oi}$ and $\hat{r}_i$. We select three points in VP and two points in RP. For example in Fig. 8, we can select T_0, T_1 and T_2, the start, middle, and end points of VP and T_3 and T_4, the start and end points of RP. We comment that it is not necessary to find the exact start, middle or end points as this does not greatly affect the accuracy of model calibration. We need 5 points since we have $\hat{\lambda}_i$, $\hat{T}_{oi}$ and $\hat{r}_i$ and two C_0s, a total of 5 variables.

$\hat{\lambda}_i$, $\hat{T}_{oi}$ and $\hat{r}_i$ computed from each sensor i are not fully equal (see Table I). We apply Algorithm ModelCal(see alg.2) ()to get the final $\hat{\lambda}$, $\hat{T}_o$ and $\hat{r}$. The basic idea is to remove outliers by setting an upper error bound ε and compute a weighted average of all $\hat{\lambda}_i$, $\hat{T}_{oi}$ and $\hat{r}_i$.

We have wrapped up the Energy-Temperature Correlation Model construction into an open source ModelCalibration using MatLab. The package is available at [23].

Algorithm 2 ModelCal()

Input: 1) Temperature sequence $T_1, T_2, \ldots$ for all nodes, 2) Power sequence $P_1, P_2, \ldots$, 3) ε
Output: $\hat{\lambda}, \hat{T}_o$ and $\hat{r}$

1: Determine VP and RP using sequence $P_0, P_1, \ldots$
2: **for all** node i **do**
3: Get λ_i, T_{oi} and r_i using IndividualCal();
4: Predict temperature sequence $T_0', T_1' \ldots, T_N'$;
5: $W_i = \sqrt[2]{\sum_{j=1}^{N} |T_j - T_j'|^2}$ // compute weight of node i, according to the accuracy of prediction
6: $\hat{\lambda} = \sum_{W_i<\varepsilon} (\frac{1}{W_i} \times \lambda_i) / \sum_{W_i<\varepsilon} (\frac{1}{W_i})$
7: $\hat{T_o} = \sum_{W_i<\varepsilon} (\frac{1}{W_i} \times T_{oi}) / \sum_{W_i<\varepsilon} (\frac{1}{W_i})$
8: $\hat{r} = \sum_{W_i<\varepsilon} (\frac{1}{W_i} \times r_i) / \sum_{W_i<\varepsilon} (\frac{1}{W_i})$

B. Experiment Validation

We conduct a real experiment to validate our model. It also serves as a test for our sensor network. Our experiment was conducted in a hotel room in Shenzhen, China. The configuration of the room and sensor network is shown in Fig. 6. There were nine indoor sensors (No. 1 to No. 9), one outdoor sensor (No. 10) to collect temperature and an electricity-meter (No. 15) connected to the air-conditioner. Our experiment lasted one day from March 2nd to 3rd 2011. We periodically turned on and off the air-conditioner(AC). The result is shown in Fig. 7. The bottom part of Fig. 7 shows the temperature of four indoor sensors and the outdoor sensor. The upper part of Fig. 7 shows the corresponding output power level of the air-conditioner (in terms of Watt).

Fig. 7 indicates the weak connection between the outdoor temperature (Node 10) and the indoor temperature. Actually Node 10 readings were the temperatures of the day. We can also see that the air conditioner turned on and off automatically in the Maintaining Period.

Using Algorithm IndividualCal() we get the $\hat{\lambda}_i$, $\hat{T}_{oi}$, $\hat{r}_i$ for each indoor sensor. Using Algorithm ModelCal() we get $(\hat{\lambda}, \hat{T}_o, \hat{r})$. The results are shown in Table I.

TABLE I: Calibrated (λ, T_o, r)

Node ID	$\hat{\lambda}_i$	$\hat{T}_{oi}$	$\hat{r}_i$
1	63.46	26.08	-0.22
2	63.98	25.97	-0.20
3	7.53	24.63	-0.08
4	53.86	25.12	-0.23
5	57.93	26.00	-0.38
6	48.78	25.22	-0.29
7	56.82	25.38	-0.27
8	71.46	25.23	-0.48
9	60.89	24.64	-0.44
$\hat{\lambda}, \hat{T}_o, \hat{r}$	58.72	25.38	-0.32

Fitting $(\hat{\lambda}, \hat{T}_o, \hat{r})$ back to our energy-temperature correlation model, we draw a predicted temperature curve in Fig. 9 by applying the same initial temperature, the same time to turn off the air-conditioner from 16:11pm and 18:10pm. We also draw the real temperature sequence of Node 5. We see that the predicted temperature is fairly close to the real temperature sequence. Thus we conclude that our model can be used to estimate future room electricity usage.

We want to further emphasize that after we have the energy-temperature correlation model, we do not need the sensor network in the room. Our experience shows that to build the model, it is enough to use the sensor network for a day or few. In our sensor network, TelosB sensors used batteries and the electricity-meter uses alternating current. The energy is not a problem. In addition, though the electricity-meter carries its own data and the temperature data of TelosB sensors, the traffic throughput is also not a problem. It is easy to see that our sensor network can be directly used in other rooms too.

V. Room Scheduling Algorithm

With the energy-temperature correlation model, we are prepared to develop the room scheduling algorithm. We have searched existing room scheduling algorithm in literature. To the best of our knowledge, we did not find any standard algorithm. We believe ad-hoc scheduling is used because of two reasons: 1) the number of rooms is not always tight, 2) there is no optimization objective, only to fit the meetings in. As such, advanced algorithms might not be necessary.

We would like to comment that by no means our intention is to conserve energy needed within meetings. Conservation of such energy is beyond the scope of this paper; but we would like to admit that if the meeting time is long (e.g., three hours), the proportion of the energy that we conserve as compared to the total energy of the meetings can be small. Nevertheless, we are working on one of the most energy consuming sectors of our society. The sheer amount of energy we conserve, as compared to not using our system, is significant.

We formally state the problem. Given a set $\mathcal{R}$ of n rooms and a set $\mathcal{M}$ of m meetings to be scheduled. A meeting $M_i \in \mathcal{M}$ is associated with a time interval (b_i, e_i) and a target temperature T_{ti}, where b_i, e_i represent the start time and the end time of the meeting respectively. Each meeting M_i has a capacity requirement c_i. Each room $R_j \in \mathcal{R}$ has a capacity C_j. For R_j that can hold M_i, we must have $c_i < C_j$. Every room R_j is associated with a function $E_j(T_t, t)$ showing the energy needed to maintain the target temperature T_t for t and a function $RE_j(T_t, t)$ showing the energy needed to re-cool the room to T_t where last meeting has ended for t. $E_j(T_t, t)$ and $RE_j(T_t, t)$ can be computed by our energy-temperature correlation model (See appendix for details). We assume the target temperature T_t is a constant. As such, we use $E(t)$ and $RE(t)$ for short. We want to find a schedule $\mathcal{S}$ consisting a set of time intervals, one for each meeting. The objective is to reduce the total energy of $\mathcal{S}$.

In this section, we first develop an optimal algorithm when the rooms are uniform. For the general problem with non-uniform rooms, we develop two fast heuristics for different scenarios.

We first define a concept of skyline. It indicates the last time each room is used. Our algorithms will iteratively move the skyline to the end times of the schedule.

Definition For n room, *skyline* is a set of numbers $(k_1, k_2, \ldots, k_n)$, where k_j is the last time of room R_j usage.

A. Complexity analysis

Theorem 1: MESP is NP-complete when there are more than two types of rooms with different energy consumption ratio.

Proof: It is easy to verify that calculating re-cooling energy consumption of a schedule is NP. Therefore, MESP is in NP class. To shown MESP is NP-complete, we reduce a job schedule problem to it. The former is proven NP-complete in [4]. The proven theorem is stated as follow: *Given a set $J = \{J_1, J_2, \ldots, J_n\}$ of n jobs, job J_i has fixed start time and end time (s_i, t_i). There are k kinds of machines where $k > 2$. For each class $j = 1 \ldots k$, the unit time processing cost is P_j where $P_j \neq P_l$ if $j \neq l$. All jobs can be processed on any kind of machines. It is NP-complete to schedule all jobs with minimum cost.*

Given an instance (J, U, JU): $J = \{J_1, J_2, \ldots, J_n\}$ is the set of n jobs, $U = \{U_1, U_2, \ldots, U_l\}$ is the set of l processors and these processors are grouped into k classes $JU = \{JU_1, JU_2, \ldots, JU_k\}$. The unit time processing cost of class JU_j is P_j. We construct a set of meetings $\mathcal{M} = \{M_1, M_2, \ldots, M_n\}$ and a set of rooms $\mathcal{R} = \{R_1, R_2, \ldots, R_l\}$. b_i and e_i for M_i are equal to s_i and t_i of J_i respectively. All meetings have same capacity requirement $\bar{c}$, all rooms have same capacity $\bar{C}$ and $\bar{c} < \bar{C}$. Let all meetings have same T_t and let the outdoor temperature be a constant. All rooms are grouped into $\mathcal{RG} = \{RG_1, RG_2, \ldots, RG_k\}$. R_i is in RG_j if and only if U_i is in JU_j. For each class of rooms, we build a simplified energy-temperature correlation model: All rooms in RG_j have same function $E_j(T_t, t) = P_j * t$ to calculate in-meeting energy consumption and $RE_j(T_t, t') = 0$.

Next, we show that by finding the minimum energy schedule $\mathcal{S}$, we can find the minimum cost schedule to process all jobs in polynomial time. Replacing (M_i, R_j) in $\mathcal{S}$ with (J_i, U_j), we have a job schedule $\mathcal{S}'$ which is a valid schedule for all jobs. For a M_i scheduled to a room in RG_j, energy consumption of M_i is $P_j * (e_i - b_i)$. And b_i and e_i for M_i are equal to s_i and t_i of J_i respectively. Thus, $\mathcal{S}'$ uses minimum cost if $\mathcal{S}$ consumes minimum energy. ■

B. Rooms with Uniform Capacity

Our algorithm Energy-Aware Room Scheduling (Uniform), Energy-RS(Uniform) for short, is a greedy-based algorithm. We sort the meetings in ascending order based on their starting times. We then group the meetings with the same starting time. Our algorithm performs in iterations and in each iteration, we handle a group of meetings with the current earliest starting time. We allocate these meetings to the rooms that have ending times that are closest these meetings.

Lemma 2: Let $\mathcal{K}$ be the set of permutations of numbers $k_1, k_2, \dots, k_n$. For all $K_i \in \mathcal{K}$, different skyline represented by K_i does not affect later scheduling.

Proof: The rooms are uniform, so exchanging the order of rooms does not affect later scheduling. ■

Lemma 3: Let two uniform rooms R_1 and R_2 have skyline (k_1, k_2). For two unscheduled meetings $M_1(b_1, e_1)$ and $M_2(b_2, e_2)$, if $k_1 < k_2 \le b_1 < b_2$, the optimal schedule should put M_1 in R_2, and M_2 in R_1.

Proof: The total interval between the scheduled meeting and unscheduled meeting $(b_1 + b_2 - k_1 - k_2)$ is constant. For two uniform rooms, they have same function $RE(t')$. Because $RE(t')$ is a concave function of t, we have the following inequality: $RE(b_1 - k_1) + RE(b_2 - k_2) > RE(b_2 - k_1) + RE(b_1 - k_2)$. As the total meeting length of M_1 and M_2 is constant, re-cooling energy consumptions determine the difference of total energy consumptions. We conclude it is more energy efficient to put M_1 in R_2 while M_2 in R_1. ■

Algorithm 3 Energy-Aware Room Scheduling (Uniform)

Sort meetings in start time ascending order;
$k_1, k_2, .., k_m = 0$;
$i = 1$;
while $i! = m$ **do**
  Find R_j and M_i where $b_i - k_j = \min_{\forall R_j, b_i > k_j}\{b_i - k_j\}$;
  Schedule M_i in R_j;
  $k_j = e_i$;
  $i = i + 1$;

Theorem 4: The total energy consumption by Algorithm Energy-RS(Uniform) is minimum.

Proof: We prove by contradiction. For any skyline $(k_1, k_2, ..., k_m)$ and two unscheduled meeting M_i,$M_h(b_i < b_h)$, M_i is scheduled to R_j where $b_i - k_j = \min_{\forall R_j, k_j \le b_i}\{b_i - k_j\}$. Assume the contrary holds, it is energy efficient to put M_i in R_l and put M_h to R_j. We have $k_l < k_j \le b_i < b_h$. This violates lemma 3 and 2, where it is more energy efficient to put M_i in R_j. ■

Theorem 5: The total number of rooms scheduled by Algorithm Energy-RS(Uniform) is minimum.

Proof: For any skyline$(k_1, k_2, ..., k_m)$, M_i is scheduled to R_j in Algorithm Energy-RS(Uniform). Assume there is a minimum room algorithm who puts M_h to R_j and put M_i in R_l. We have $k_l \le k_j \le b_i \le b_h$. Thus, the position of M_h and M_i are interchangeable. And according to claim 2, this interchange does not affect the later scheduling. So the schedule result of Algorithm Energy-RS(Uniform) uses as many rooms as the minimum room algorithm. ■

Existing ad-hoc meeting scheduling algorithms usually do not bother if using more rooms. This theorem indicates that Algorithm Energy-RS (Uniform) will select the smallest number of rooms. This is useful for the general algorithm with non-uniform rooms; since we try not to schedule meetings with small capacity requirements into oversized rooms.

C. Rooms with Non-Uniform Capacity

1) Energy-RS(): We use Algorithm Eenergy-RS (Uniform) as a building block to develop algorithm Energy-Aware Room Schedule (Energy-RS()). We outline our basic idea. Assume the number of different capacities of all rooms is g. We classify the rooms into different groups $\mathcal{RG}_1, \mathcal{RG}_2, \dots, \mathcal{RG}_g$ according to their capacity. Let GC_k be the room capacity of $\mathcal{RG}_k$. We have $\forall R_j \in \mathcal{RG}_k$, $C_j = GC_k$. Assume $\mathcal{RG}_1, \mathcal{RG}_2, \dots, \mathcal{RG}_g$ is sorted in ascending order according to their capacity GC_k. We classify the meeting into different groups $\mathcal{MG}_1, \mathcal{MG}_2, \dots, \mathcal{MG}_g$ according to the capacity requirements of the meetings. For a meeting M_i with a capacity requirement c_i, it is grouped into $\mathcal{MG}_k$ where $GC_{k-1} < c_i \le GC_k$. As an example, assume the room capacities of all rooms are 20, 40, 60. The meeting requirements are 17, 18, 34. We thus classify the meetings with capacity requirements of 17 and 18 into the group of 20 and the meeting with capacity requirement of 34 into the group of 40.

We schedule meetings of $\mathcal{MG}_k$ into room group $\mathcal{RG}_k$ in ascending order. For each group pair $(\mathcal{MG}_k, \mathcal{RG}_k)$, we apply Algorithm Energy-RS (Uniform). If there is some meetings cannot be scheduled, we move these unscheduled meetings into $\mathcal{MG}_{k+1}$. From Theorem 5, we know that Algorithm Energy-RS (Uniform) uses the smallest number of rooms. Thus, the chance that a meeting with small capacity requirement is pushed into an oversized room is minimized.

Claim 6: The complexity of Algorithm Energy-RS() is $O(nm)$.

2) TimeUr-RS(): In our framework, each meeting has a capacity requirement and a meeting time requirement. This is the case for many scenarios. For some cases, however, the meeting time can be determined by the room scheduling system. For example, when we query the class schedule of The Hong Kong Polytechnic University, the administrative personnel could not give out specific reasons why one specific class must be at a specific time, as long as there is no conflict. We conjecture this is a general case since there is no optimization goal for many meeting schedules; only to fit all the meetings in without meeting-meeting, room-room conflict.

We propose a simple greedy-based algorithm which allows reassignment of meeting times, we call Time Unrestricted Energy Aware Room Scheduling (TimeUr-RS()). TimeUr-RS() is greedy by sorting meeting capacities in descending order and then fitting into the rooms. This algorithm can be used to provide suggestions for the decision makers, in case there is no compulsory reason to have strict meeting times. In our simulation, TimeUr-RS() is used as a performance comparison.

VI. Performance Evaluation

A. Simulation setup

We evaluate our system in two settings. The first one is real class schedules and classroom topologies of The Hong Kong Polytechnic University (denoted PolyU thereafter), academic calendar year of Spring 2011. The second is a set of synthetic room arrangements we generate semi-randomly.

PolyU has 155 classrooms (see Table II for the full configurations). λ, mC and P are calculated based on the room size and the equations in section IV.A. In academic year Spring 2011, there are around 600-700 classes every weekday.

TABLE II: Room configuration of PolyU

Cap (Seats)	Num	Size ($L \times W \times H$, m)	λ ($J/s \cdot K$)	mC (J/K)	P (kW)
20	8	$4 \times 5 \times 3$	70.5	1200	1.5
40	42	$8 \times 5 \times 3$	118.5	2400	2.4
60	67	$6 \times 10 \times 3$	162	3600	4.7
80	10	$8 \times 10 \times 3$	201	4800	6.2
100	4	$10 \times 10 \times 3.3$	249	6600	9.4
150	17	$10 \times 15 \times 4$	375	12000	15.6
200	5	$15 \times 14 \times 5$	533	21000	21.9
300	2	$15 \times 20 \times 6$	765	36000	31.3

The second setting is synthetic data. We consider rooms with uniform capacity and non-uniform capacity separately. For the uniform case, the default room capacity is 100 seats and the total number of rooms is 150. The meeting times are randomly generated in range [8:00, 22:00]. The lengths of the meetings are randomly chosen from a few fixed options, as we believe most meetings have semi-fixed length. We have three options, $\mathcal{O}_1 = [1, 1.5, 2, 2.5, 3]$, $\mathcal{O}_2 = [1, 2, 3]$, $\mathcal{O}_3 = [1, 2]$. That is, for $\mathcal{O}_1$, the meeting lengths are randomly chosen from one of the five choices, 1, 1.5, 2, 2.5, or 3 hours.

For the non-uniform capacity case, we have eight different types of rooms with capacities of 20, 40, 60, 80, 100, 150, 200, 300 (similar to PolyU). The numbers of different types of rooms follow a poisson distribution with a mean of 3. This indicates that the majority of our rooms are those with capacity of 60 seats. The total number of rooms is also 150. The meeting times are randomly generated in range [8:00, 22:00]. The length of the meetings are also from the three options, $\mathcal{O}_1$, $\mathcal{O}_2$, and $\mathcal{O}_3$. The capacity requirement for the meetings follows a poisson distribution with a mean of 3.

The default values of our simulation are $\hat{T}_o = 25°C$, $\hat{r} = -0.32$ for all rooms (same to the value in Table I). We set the target temperature $T_t = 20°C$ for all meetings.

For the PolyU setting, we directly compare the schedule computed by our algorithm with the existing schedule. For synthetic data, we choose to compare our algorithm with an ad-hoc room scheduling algorithms (denoted as RS) that can can satisfy the meeting time and room capacity requirements.

We choose our primary performance metric as the total energy needed to re-cool the rooms to the target temperature for all rooms and all meetings. Note that we exclude the energy needed during the classes; which we cannot conserve. This metric is stable for all room scheduling algorithms.

B. Simulation results

1) Results on Synthetic Data: In Fig. 10, we show the total energy for re-cooling the rooms for different algorithms. In Fig. 10 (a), we see that the re-cooling energy needed for ad-hoc room scheduling RS is always greater than our algorithm Energy-RS and TimeUr-RS. This is not surprising as the RS only satisfies the meeting requirements. When the number of meetings increases, we can see that all three algorithms need more energy in re-cooling the rooms. This is because there are more meetings and more rooms to be used. RS increases much faster than our algorithms, however; as both of our algorithms have taken the energy conservation into consideration. More specifically, we can see that if there are 800 meetings to schedule, the total electricity needed by RS, Energy-RS and TimeUr-RS is 503 kWh, 214 kWh, and 141 kWh respective. We can see that we have reduced the electricity consumption for more than half. If the meeting time is not restricted, we can make a suggestion on meeting times so as to reduce the electricity consumption to less than one third.

Consider a regular desktop computer. The electricity consumption per hour is usually less than 0.1 kWh when it is in full CPU operation. If the computer is in use for 12 hours per day, and in hibernation for the rest time. This means that the electricity we saved can support more than 300 desktop computers. This reflects that air-conditioning (and also heating) is indeed a major sector for energy consumption.

We then see Fig. 10 (b) and (c) where the meeting time is randomly chosen from $\mathcal{O}_2$ and $\mathcal{O}_3$. We see similar trend as that in Fig. 10. We also see that the less number of choice that we have in meeting time, the greater the benefit of our algorithms. This is because if there is a smaller number of meeting length options, there is also a smaller number of small time segments that we cannot fit the meetings in due to more irregular meeting time length. On the contrary, we do not see improve for RS as its schedule is ad-hoc.

This can be more clearly seen from Fig. 11. We call the re-cooling energy ratio as the re-cooling energy needed by Energy-RS (or TimeUr-RS) as against to the re-cooling energy needed by RS. In Fig. 11, we plot the re-cooling energy ratio for the case where the number of meetings is 800. We can see when the meeting length become more uniformed, the re-cooling energy ratio of Energy-RS and TimeUr-RS becomes smaller. This suggests that to save more energy, it is better to have the meeting length more uniform.

Fig. 12 shows re-cooling energy needed when we use different room capacity (our default is 100 seats). The total number of meetings is 800 and we choose $\mathcal{O}_3$ as our meeting length. Clearly, the larger the room capacity, the more re-cooling energy is needed for all algorithms. Our algorithms greatly outperforms RS for more than 50%.

The energy consumption is closely related with the target temperature. We adjust the target temperature T_t from 20°C to 24°C. From Fig. 14, we see that every degree counts! For example, the re-cooling energy is around half if we increase our target temperature from 20 to 23. This suggests that the best way to save energy is to set the temperature bar higher.

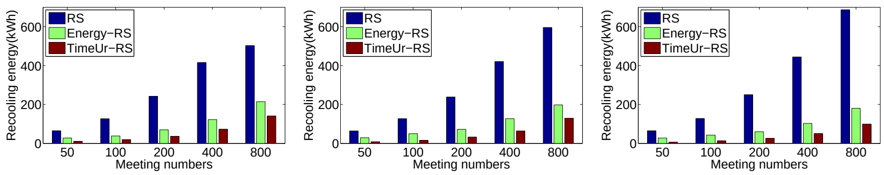

Fig. 10: Total energy expense for re-cooling the rooms as against to the number of meetings; rooms with uniform capacity. (a) meeting length: option $\mathcal{O}_1$, (b) meeting length: option $\mathcal{O}_2$, (c) meeting length: option $\mathcal{O}_3$.

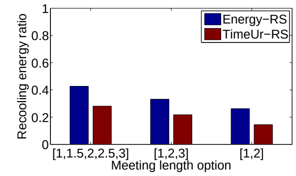

Fig. 11: Re-cooling energy ratio.

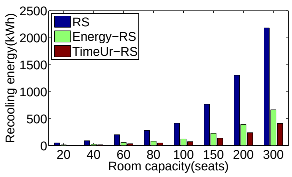

Fig. 12: Re-cooling energy as a-
gainst room capacity

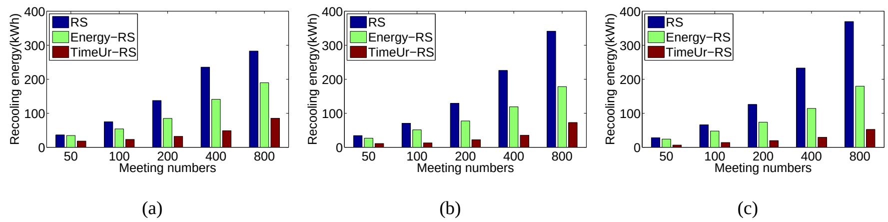

Fig. 13: Total energy expense for re-cooling the rooms as against to the number of meetings; rooms with non-uniform capacity. (a) meeting length: option $\mathcal{O}_1$, (b) meeting length: option $\mathcal{O}_2$, (c) meeting length:

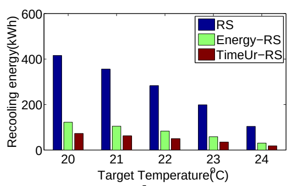

Fig. 14: Re-cooling energy as against target temperature.

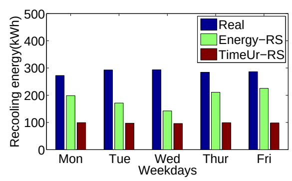

Fig. 15: Re-cooling energy in weekdays

Our algorithm again significantly outperforms RS.

We then study the general case where rooms are of non-uniform capacity. We show the results in Fig. 13. We see that the gain of Energy-RS is smaller. This is because, in each type of room capacity, we have a much smaller number of meeting choices. If one takes a closer look at Fig. 10 (a), we can see that the best performance arrives when the number of meetings is 800. When the number of meetings is 100, or 50, the gain is smaller. In our general case, we have 8 different types of rooms resulting in a smaller number of meetings in each type. Thus, the gain is smaller. We can summarize that the more meetings, the more choices; leading to more re-cooling energy needed; and a better performance of Energy-RS as compared to RS.

2) Results on PolyU Data: We next study a real data set of PolyU. Note that though the academic calendar year of Spring 2011 spans for an entire semester, the class schedule for each week is the same. For example, the class schedule of PolyU of every Monday (or any other weekday) is the same in the entire semester. As such, we will only schedule for five weekdays and our schedule can be used in every week of the semester.

Fig. 15 summarizes the results. We can see that every day the re-cooling energy needed is approximately the same. This is because the total classes in different weekdays are more or less the same, which is the usual case of a university. We also see a general 20% conservation in electricity for each weekday. If there is less restriction in class time, then we will achieve higher energy conservation.

VII. Related Work

We are in an age where people are paying increasing attention to energy conservation around the world. Computer scientists study energy conservation of data centers [16][17][19], and backbone routers [24]. The general principle of these works is to turn off unnecessary usage of machines and reschedule their load. To assist data center monitoring, sensor network is used for energy sensing [9].

There are many efforts in developing smart homes and buildings. In [6][7], an energy auditing network is built. One main objective is to have a fine-grained granularity on electricity readings for all equipments. As a continuation, in [1], sMAP is developed, which can record different physical readings and provide general interface for different applications. Motion sensors and door sensors are developed [12] to model the occupancy pattern of people at home. It turns off the light, air-conditioning etc when people are not at home. A similar system [14] analyzes the occupancy against pre-booked conference rooms, so as to turn off unnecessary energy usage. A few studies use sensors and actuators to work collaboratively to monitor buildings [10][13][20].

In building and service engineering, there are mature Central Control and Monitoring Systems (CCMS) designed for high-end commercial buildings. Building Management System (BMS) is part of CCMS and is responsible for monitor and control HVAC equipments. There is a standard protocol BACNet for BMS. The main objective is to have a uniformed way to connect different direct digital controllers (DDC). BACNet rides on Ethernet. Recently, there are some studies try to use wireless networks to carry BACNet [15][21].

Our work is a specific application for sensor networks to monitor buildings. Our work targets on the heating and air-conditioning sector for commercial buildings, which is the largest factor for energy consumption. Our work does not try to monitor the presence of people in a room and turn on and off room functions; a common topic in many studies. We see many rooms to date have already equipped with this function. Instead, we reduce unnecessary heating and air-conditioning by taking advantage of scheduling subsequent meetings or classes to rooms that have just been used (heated, or air-conditioned). Our work can become an integrated part for future smart BMS system for meeting arrangement. Or it can work individually for class schedules of universities. To the best of our knowledge, we are the first to study such problem.

VIII. Conclusion and Future Work

In this paper, we took the advantage of *Thermal Inertia*; that is, after a meeting ends in a room, the cool air will not immediate dissipate. We designed a new room management system for energy conservation. We extended sensor hardware (some of which can be used beyond this work) and designed a two tier sensor network. We develop an energy-temperature correlation model and validate the model with our sensor network in real-world experiment. We further developed efficient room scheduling algorithms. Comprehensive simulations on synthetic data and a real class and room configuration of The Hong Kong Polytechnic University were conducted.

As a first work in Thermal Inertia, our work has many limitations. First, in our current paper, we assume the meetings are determined in advance. This is true for university schedules. However, many companies/hotels/resturants face online room booking. Second, in the current schedule, we assume the room capacity is the only constrain to meetings. We admit that there are other constraints, such as facilities (e.g., projectors) in the room, the distance between rooms so that people have enough time to go from one room to another. To facilitate future studies, we release an open source for our energy-temperature correlation model in MatLab. One can use our work to generate realistic input on energy consumptions for different room scheduling problems. Third, our electricity-meter can only measure energy usage of general air-conditioners. We plan to develop advanced meters for central controlled air-conditioners. Fourth, we are in collaboration with people from Building and Service Engineering, to see how our system can be fitted into general building management systems.

References

[1] S. Dawson-Haggerty, X. Jiang, G. Tolle, J. Ortiz, and D. Culler, “sMAP - a Simple Measurement and Actuation Profile for Physical Information”, in Proc. *ACM SenSys’10*, Zurich, Switzerland, Nov. 2010.

[2] Basic Concepts Manual - Essential Information, the US Department of Energy, USA, http://apps1.eere.energy.gov/buildings/energyplus/pdfs/gettingstarted.pdf.

[3] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss and Philip Levis, “Collection Tree Protocol”, in Proc. *ACM SenSys’09*, Berkeley, CA, Nov. 2009.

[4] Q. Huang and E. Lloyd, “Cost Constrained Fixed Job Scheduling”, Lecture Notes in Computer Science, vol. 2841, 2003

[5] Hong Kong Energy End-Use Data, 2010, The Energy Efficiency Office, Electrical and Mechanical Service Department (EMSD), Hong Kong, http://www.emsd.gov.hk/emsd/e_download/pee/HKEEUD2010.pdf

[6] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler, “Design and Implementation of a High-Fidelity AC Metering Network”, in Proc. *ACM IPSN’09*, San Francisco, CA, Apr. 2009.

[7] X. Jiang, M. Ly, J. Taneja, P. Dutta, and D. Culer, “Experiences with a High-Fidelity Wireless Building Energy Auditing Network”, in Proc. *ACM SenSys’09*, Berkeley, CA, Nov. 2009.

[8] B. Li, D. Wang, F. Wang, and Y. Q. Ni, “High Quality Sensor Placement for Structural Health Monitoring Systems: Refocusing on Application Demands”, in Proc. *IEEE INFOCOM’10*, San Diego, CA, Mar. 2010.

[9] C. Liang, J. Liu, L. Luo, A. Terzis, and Feng Zhao, “RACNet: A High-Fidelity Data Center Sensing Network”, in Proc. *ACM SenSys’09*, Berkeley, CA. Nov., 2009.

[10] K. Lin and R. Gupta, “Towards Automated Building Management through Cooperative Sensor-actuator Networks”, in Proc. *HotPower’09*, Big Sky, MT, Oct. 2009.

[11] J. Lienhard IV and J. Lienhard V, “A heat transfer textbook,3rd edition”, pp. 12-18, 2003.

[12] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, E. Field, K. Whitehouse, “The smart thermostat: Using Occupancy sensors to save energy in homes”, in Proc. *ACM SenSys’10*, Zurich, Switzerland, Nov. 2010.

[13] J. Lu, D. Birru, K. Whitehouse , “Using Simple Light Sensors to Achieve Smart Daylight Harvesting”, in Proc. *ACM BuildSys’10*, Zurich, Switzerland, Nov. 2010.

[14] K. Padmanabh, A. Malikarjuna, S. Sen, S. Katru, A. Kumar, S. Pawankumar, S. Vuppala, and S. Paul, “iSense: A wireless sensor network based conference room management system”, in Proc. *ACM BuildSys’09*, Berkeley, CA, Nov. 2009.

[15] S. Park, W. Lee, S. Kim, S. Hong, and P. Palensky, “Implementation of a BACnet-ZigBee gateway”, in Proc. *INDIN’10*, Osaka, Japan, July, 2010.

[16] L. Parolini, B. Sinopoli and B. Krogh, “Reducing data center energy consumption via coordinated cooling and load management”, in Proc. *HotPower’08*, San Diego, CA, Dec, 2008.

[17] R. Raghavendra, P. Ranganathan, V. Talwar, Z.Wang, X. Zhu, “No Power Struggles: Coordinated Multi-level Power Management for the Data Center”, in Proc. *ACM ASPLOS’08*, Seattle, USA, Mar, 2008.

[18] H. Sauer, R. Howell, W. Coad, “Principles of heating, ventilating, and air conditioning”, Section 2.4, 2001.

[19] Y. Shang, D. Li and M. Xu, “Energy-aware Routing in Data Center Network”, in Proc. *ACM SIGCOMM Workshop on Green Networking’10*, New Delhi, India, Aug. 2010.

[20] L. Schor, P. Sommer, and R. Wattenhofer, “Towards a Zero-Configuration Wireless Sensor Network Architecture for Smart Buildings”, in Proc. *ACM BuildSys’09*, Berkeley, CA, Nov. 2009.

[21] K. Tsang, W. Lee, K. Lam, H. Tung, and X. Kai, “An integrated ZigBee automation system: An energy saving solution”, in Proc. *the 14th International Conference on Mechatronics and Machine Vision in Practice*, Xiamen, China, 2007.

[22] Energy in the United States, http://en.wikipedia.org/wiki/Energy_in_the_United_States

[23] Y. Yuan, D. Pan, D. Wang, X. Xu, Y. Peng, X. Peng, P. Wan “Thermal Inertia: Towards An Energy Conservation Room Management System.” Technical report and Matlab package, Jun. 2011. Available at http://www4.comp.polyu.edu.hk/~csyiyuan/projects/ECRMS.html

[24] M. Zhang, C. Yi, B. Liu, and B. Zhang, “GreenTE: Power-Aware Traffic Engineering”, in Proc. *IEEE ICNP’10*, Kyoto, Japan, Oct. 2010.

Appendix

Let t be the interval time that the room has not been used. Let t_i be the length of M_i. From our energy-temperature correlation model, we have $\hat{\lambda}$, $\hat{T}_o$, $\hat{r}$. We compute $E(T_t, t_i)$ and $RE(T_t, t)$ as follows:

$$E(T_t, t_i) = \left(\frac{(T_t - \hat{T}_o)}{\hat{r}} \times \hat{\lambda} \right) \times t_i; \tag{5}$$

$$RE(T_t, t) = (P - \frac{(T_t - \hat{T}_o)}{\hat{r}} \times \hat{\lambda}) \times - \frac{mC}{\hat{\lambda}} ln \left(\frac{T_t - \hat{T}_o - \frac{\hat{r} \times P}{\hat{\lambda}}}{(T_t - \hat{T}_o) e^{-\frac{\hat{\lambda}}{mC} t} - \frac{\hat{r} \times P}{\hat{\lambda}}} \right) \tag{6}$$