Experiences with Occupancy Based Building Management Systems

Nipun Batra, Pandarasamy Arjunan, Amarjeet Singh, Pushpendra Singh
Indraprastha Institute of Information Technology, Delhi, India

Abstract—**Buildings are one of the largest consumers of electricity. Dominant electricity consumption within the buildings, contributed by plug loads, lighting and air conditioning, can be significantly improved using *Occupancy-based* Building Management Systems (Ob-BMS). In this paper, we address three critical aspects of Ob-BMS i.e. 1) Modular sensor node design to support diverse deployment scenarios; 2) Building architecture to support and scale fine resolution monitoring; and 3) Detailed analysis of the collected data for smarter actuation. We present key learning across these three aspects evolved over more than one year of design and deployment experiences.**

The sensor node design evolved over a period of time to address specific deployment requirements. With an opportunity at the host institute where two dorm buildings were getting constructed, we planned for the support infrastructure required for fine resolution monitoring embedded in the design phase and share our preliminary experiences and key learning thereof. Prototype deployment of the sensing system as per the planned support infrastructure was performed at two faculty offices with effective data collection worth 45 days. Collected data is analyzed accounting for efficient switching of appliances, in addition to energy conservation and user comfort as performed in the earlier occupancy based frameworks. Our analysis shows that occupancy prediction using simple heuristic based modeling can achieve similar performance as more complex Hidden Markov Models, thus simplifying the analytic framework.

I. INTRODUCTION

Across the globe, buildings, both commercial and residential, are the largest consumers of energy, accounting for 47% of national energy use in India [7], 45% in UK [10], and 41% in the USA [1]. The energy performance of buildings directive by European Union has asserted that the way in which buildings are used, besides their design and built, affects their energy consumption. On similar lines, prior research work has shown the influence of occupant behavior in the overall energy consumption of the buildings [9]. Recent work has also shown that for similar appliances, family size and Heating, Ventilation and Air Conditioning (HVAC) systems, differences in usage behavior result in large variations in energy consumption [14]. As a result, it is imperative to monitor how the building is being used (by monitoring physical parameters in the building space) to correspondingly understand the effect of occupant behavior on building energy consumption.

Several commercial Building Management Systems (BMS) currently exist that monitor multiple physical and electrical parameters and correspondingly perform control for efficient building management. However these systems mostly provide centralized control through a facilities department, are very expensive to install and operate and are difficult to expand for monitoring additional parameters and performing additional actuation as may be required by building occupants. Notably, most of these BMS involve sensing and control over wired interfaces.

Motivated by the commercial BMS and their limitations, the overall objective of our research is to develop affordable pervasive sensing and computing technologies for sustainable buildings. Correspondingly, we develop occupancy based BMS (Ob-BMS) using a hybrid of wired and wireless sensing and actuation, involving open source hardware and software systems. Proposed Ob-BMS can be easily expanded to perform necessary actions as desired by occupants and building managers. In this work, we focus on hardware and associated support infrastructure required for low-cost, large scale deployment of Ob-BMS. We have also recently developed SensorAct system [4] that can be easily customized to provide necessary software support for data aggregation, monitoring, control and analytics required for Ob-BMS.

We did a pilot deployment of the proposed hardware system as per the planned support infrastructure across two faculty offices for more than 50 days. As an illustration of possible control strategies that can be adopted in Ob-BMS, we model the occupancy based energy management across four different dimensions namely prediction accuracy, user comfort level, energy saving and switching efficiency. We use hybrid sensory information involving Passive Infra Red (PIR) sensor giving motion information and magnetic reed sensor providing door open/close status and evaluate multiple heuristic approaches for occupancy prediction. Our analysis show that simple heuristic based approaches provide equivalent accuracy as more complex Hidden Markov Model (HMM), thus simplifying the analytic framework for Ob-BMS.

II. RELATED WORK

Typical occupancy based systems involve diverse sensing elements providing multitudes of information, data aggregation and analytical systems and finally smart actuation. In this section we first discuss some of the hardware platforms that have been commonly used for various pervasive sensing applications, particularly for building related monitoring. We then discuss some of the middleware platforms that enable data aggregation and analytics both within the system and through support of third party applications. Finally we discuss existing work on occupancy based control, specifically for the case of HVAC control.

Sensing and communication platforms: Several hardware platforms, such as Arudino[1] and TelosB[2], have been designed

---

[1]http://www.arduino.cc/
[2]http://openwsn.berkeley.edu/wiki/TelosB

(a) Initial SenseKit Design   (b) Evolved SenseKit Design   (c) OpenPicus based (Bottom view)   (d) OpenPicus based (Top view)
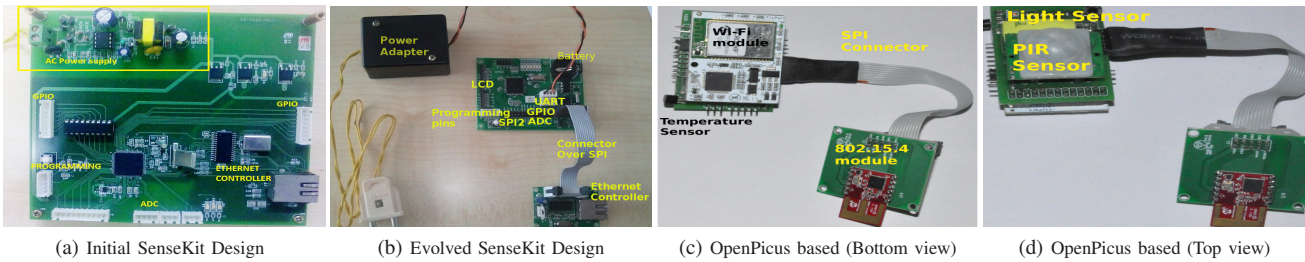
Fig. 1: Evolution of SenseKit

and used for building related sensing applications in the recent past [3]. These sensing platforms generally involve common hardware functionalities such as Analog to Digital Converters (ADCs) for Analog sensing, Digital I/Os for interfacing with digital sensors or using 1-wire interface[3] and several other communication interfaces such as I2C, SPI and UART. Preliminary analysis of the collected data is performed on-board and then the processed data is sent over multitude of networking interfaces ranging from 802.15.4 and Bluetooth for Personal Area Network to WiFi and Ethernet for Local Area Network. Custom sensing applications can then be built using available hardware resources and networking interfaces. Our proposed SenseKit platform was designed in a modular manner and iteratively improved upon keeping in mind specific application requirements.

Middleware for Building Management: Several middleware architectures, such as SensorAndrew [13], sMAP [5] and SensorAct [4] have been designed for overall building management. Their primary functionality includes 1) data aggregation from heterogeneous sensing hardware platforms 2) providing easy ways to manage several resources and subsystems in the building and 3) provide easy data access for the authorized persons to build custom applications and infer interesting patterns by analyzing the data. We use one such system [4] for data aggregation and analytics in our proposed Ob-BMS.

Occupancy based HVAC: Recent systems [2], [11] show that occupancy is an important parameter in energy efficient HVAC control and show the reduction in energy usage by optimizing the same. Our work builds on the prior work in this domain and aims to establish the importance of appliance life in addition to occupant comfort and energy efficiency. We further show that simple heuristic based approaches can provide similar level of accuracy as compared to more complex HMM based approach for occupancy based control.

Commercial BMS: For end-to-end management of buildings several commercial BMS[4] have emerged. However, these systems are typically controlled by a facilities department and are largely operated on static schedules. For instance in one of our buildings the air is chilled to 25 degree celsius from 7 AM to 7 PM on weekdays. This approach does not take occupancy into consideration and hence is extremely energy inefficient and may cause discomfort to occupants staying

beyond 7 PM. We plan to interact with existing commercial BMS over protocols such as BACnet and incorporate our occupancy analytics and finer grained sensor information for improved Ob-BMS.

III. SENSEKIT - A MODULAR SENSING PLATFORM

We developed SenseKit, a hardware system that went through multiple iterations of design with primary objective of being flexible to support diverse sensors and easy to customize as per application requirements. While it was desired that SenseKit consume low energy during operational mode, but ultra low-energy is not critical as the system will be deployed inside the building where power is readily available. SenseKit contains PIC24FJ128GA010 microcontroller from Microchip, selected for the following reasons:

- Embedded hardware modules (e.g. processor, real time clock, flash) and interfaces (e.g. SPI, I2C, UART), are enough to support diverse application requirements while keeping the cost low.
- Microchip offers a free licensed TCP/IP stack[5] useful for collecting data from diverse sensors and communicating it over LAN to a central server. Additionally, Microchip provides a TCP/IP graphical configuration utility within their MPLAB IDE for easy configuration of the stack as per application requirements.
- Microchip also provides a broad range of low power networking interfaces that can communicate with its microcontrollers over standard SPI interface, e.g. Ethernet[6], WiFi[7] and 802.15.4[8], thus providing the flexibility to add diverse networking interfaces as per the requirement.

SenseKit exposes all the necessary interfaces such as ADC, I2C, SPI, UART and GPIOs for interfacing with diverse sensors and actuators as well as for network communication. Hardware design of SenseKit system evolved over multiple iterations, as shown in Figure 1. In our first version of SenseKit , as shown in Figure 1a, power supply and Ethernet interface were integrated in a single board. From our first deployment experience, we realized that it is challenging to connect high voltage AC supply to where the node was deployed. Additionally, limiting to the hard wired Ethernet interface was limiting the deployment scenarios, particularly when we wanted the place the node in the ceiling of the rooms.

---

[3]http://en.wikipedia.org/wiki/1-Wire

[4]http://www.trane.com, http://www.johnsoncontrols.com

[5]http://goo.gl/T6XWk

[6]http://goo.gl/4IlUU

[7]http://goo.gl/paz4k

[8]http://goo.gl/0IFfR

Based on the initial deployment experiences, the SenseKit system was re-designed with separate daughter boards with different network interfaces (supporting different boards for Ethernet, Wi-Fi and 802.15.4) and separate power supply board thus only requiring to run low voltage DC up to the node. Both the power supply and network daughter boards are connected to the main microcontroller board using SPI interface pins. The microcontroller board also provides the flexibility to attach a 16X2 LCD display for remote debugging without the computer. The new SenseKit system design is shown in Figure 1b.

Embedded program, supporting sensor and actuator interfaces together with network interface for data transmission and receiving control commands, is designed in a modular manner. As an example, a programmer needs to add/modify only 15-20 lines of code across 2 different source files to add a new sensor or actuator and set its configurations. We also extended Microchip's opensource TCP/IP library to additionally provide support for 802.15.4 interface thus allowing the SenseKit to interact seamlessly over diverse network interfaces. As a proof of concept of modularity and ease of programming, several student groups (new to microcontroller programming) were able to easily adapt the embedded software for diverse course projects in Embedded Systems course in Winter semester[9].

Recently, we have moved towards OpenPicus[10], a COTS based microcontroller platform which is very similar to our new SenseKit design. Several reasons that motivated moving to a commercially available platform were:

- It also uses a 16-bit PIC microcontroller thus providing all the necessary features we wanted in our hardware system. Further the Flyport module uses PIC24FJ256 that inherently allows for pin remapping thus providing additional flexibility for external interfaces using smaller number of I/O pins
- TCP/IP stack and the application are kept separate using FreeRTOS[11] supported on OpenPicus. Additionally, they provide a free openPicus IDE that further abstracts out the stack interface and provides easy configuration and programming options.
- OpenPicus is now developing a worldwide community, provides a well tested and mature design and is easily available in local markets. As a result of volume manufacturing, it is also of lower cost than developing similar design ourselves in small production quantities.

We designed a simple two layer board to extend the OpenPicus header to support for diverse sensors (currently supporting temperature sensor over 1-wire interface, light sensor over I2C interface and PIR sensor over digital interface), connect external 802.15.4 radio module over SPI interface, voltage regulation and ease of programming. Extended OpenPicus modules with WiFi interface for network communication, as shown in Figure 1c and Figure 1d are currently under
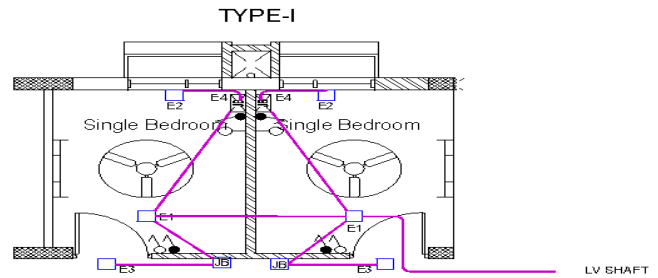
[9]https://sites.google.com/a/iiitd.ac.in/emsys2012/

[10]http://www.openpicus.com/

[11]http://www.freertos.org/

Fig. 2: Dormitory room layout

deployment in one of the research wing at our campus.

IV. PLANNING A LARGE SCALE BUILDING SENSOR NETWORK

With an opportunity of participating in the design phase of two dorm buildings being constructed on campus, we planned infrastructure to support large scale Ob-BMS at a low cost. Support infrastructure included conduits and junction boxes (that can house sensors or sensing platform) planned as per the diagram shown in Figure 2. Junction Box E1, located at the center of the room in the ceiling, was planned to house sensors such as motion, light and temperature. Junction Box E2 was placed right above the window to detect window status as well as to connect to sensors that may be placed external to the room (e.g. light and temperature). Junction Box E3 was placed right above the door to detect for door status as well as other sensors that may be placed in the hallway. Window in each room opens externally while the door opens in the hallway. Each of E1, E2 and E3 are internally connected to Junction Box E4 which is planned to house the sensing node (e.g. our SenseKit System). Each of the electrical points (for lighting, fan, air conditioning and plug loads) are also internally connected with E4 so as to allow for actuation in the future. Additionally an Ethernet cable is provisioned for E4 to transmit the collected data to a central server.

We planned for wired interface to ensure robustness and scalability at a low cost (not requiring wireless radio for each of the sensor deployed at a physically separate location). This infrastructure design was replicated across more than 400 rooms across the two dorm buildings. Prototype deployment, similar to this deployment was performed across two faculty offices for several days. Analysis from the collected data is presented in Section V-A. Following are the learning from being involved in large scale infrastructure design and prototype deployment in two faculty offices:

1) Some of the conduits get blocked by concrete at the time of construction. It is not possible to remove this blockage at a later stage. As a result an alternate conduiting path should be planned to work around the blockage. In our case, we connected the two E1 Junction Boxes and E4 Junction Boxes across two adjacent rooms to provide for alternate path.
2) Good quality shielded cable should be chosen for wiring the sensors. We tested multiple cables before deciding on using the internal wires of a Cat6 cable for sensing.
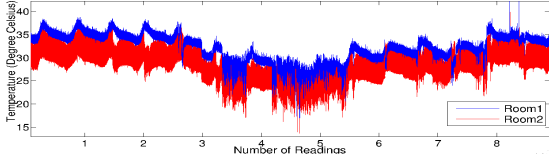
Fig. 3: Temperature difference across the two rooms due to varying wire lengths

3) Conduiting is typically done by electrical department. Since we chose Cat6 for sensing, the networking department did the wiring. There was a lot of confusion interacting with multiple departments over several reasons (e.g. conduit blocking with concrete). Based on our experience, we would recommend using the same department for both conduiting and wiring.

4) Sensor interfaces should be carefully chosen so as to admit long transmission lengths. Interfaces such as I2C are short distanced and should be avoided. Further analog signals will experience voltage drop dependent on the length of the cable. This was particularly observed when we transmitted analog output from temperature sensor in our prototype deployment. As shown in Figure 3, wherein Room-1 cable length was smaller than Room-2, a constant temperature difference (as the difference in cable lengths was not accounted for) was observed even when the two rooms were adjacent to each other. We have finally decided on 1-wire interface for temperature, analog interface for light (since we only need relative light intensity to outside and absolute value is not desired) and digital interfaces for motion and door/window status. Initial deployment with these sensory interfaces in 1 room of the new dorm building is working well.

We collected data using second version of SenseKit on the hardware side and SensorAct [4] on the software side. We believe that our infrastructure design and corresponding learning will act as blueprint for anyone planning on sensing infrastructure for smart buildings at the design phase.

V. OCCUPANCY PREDICTION ANALYTICS

Preliminary Ob-BMS data, collected using prototype deployment involving SenseKit and the planned support infrastructure, consisted of temperature information (using an analog temperature sensor), door status (using digital magnetic Reed sensor) and motion information (using digital PIR sensor). SenseKit was deployed across two faculty offices with sensors sampling at 1 Hz for 45 days. The Reed sensor gives a high (1) immediately when the door is closed and a low (0) otherwise. The PIR sensor gives an immediate high (1) when motion is detected and a low (0) after a small timeout since the last detected motion. The timeout is built into the PIR sensor hardware itself and is typically configurable. Since our data was primarily collected in winters, when air conditioning was not used, temperature information did not provide any information gain in our prediction schemes. Thus it was omitted from the prediction schemes described in Section V-A.

While using the occupancy information for sustainable buildings, prior related work has established metrics corresponding to user comfort (such as Miss Time), and energy efficiency (such as Energy Saving) [11]. We extend existing work and propose switching efficiency as another metric for evaluation of Ob-BMS. Switching efficiency was included since rapid switching of appliance state reduces its life and leads to additional electricity consumption due to typically high starting load. All these metrics are determined by the accuracy of occupancy prediction system. Correspondingly, we add prediction accuracy as the fourth evaluation metric.

Each of these four metrics depend on the occupancy prediction algorithm used (A), correspondingly predicted occupancy time series (P), and the actual ground truth information, represented as time series G. To evaluate different occupancy prediction algorithms, we define four factors (normalized between 0 and 1), corresponding to each of the four defined metrics as:

- **Prediction Accuracy Factor** (PAF, also termed as Accuracy) is the ratio of correctly predicted time samples to the total number of samples, i.e., $PAF(A,G) = \dfrac{\sum_{i=1}^{size(G)} P(i)=G(i)}{size(G)}$

- **User Comfort Factor** (UCF, also termed as Sensitivity) is the ratio of correctly predicted time samples when the occupant is present to the total number of samples when occupant is present, i.e., $UCF(A,G) = \dfrac{\sum_{i=1}^{size(G)} P(i)=G(i)=1}{\sum_{i=1}^{size(G)} G(i)=1}$

- **Energy Conservation Factor** (ECF, also termed as Specificity) is the ratio of correctly predicted time samples when the occupant is absent to the total number of samples when the occupant is absent, i.e., $ECF(A,G) = \dfrac{\sum_{i=1}^{size(G)} P(i)=G(i)=0}{\sum_{i=1}^{size(G)} G(i)=0}$

ECF does not account for energy wastage due to higher input load because of multiple appliance switching.

- **Switching Efficiency Factor** (SEF) is the ratio of efficient appliance state switches to the total number of switches. We define an *efficient switch*, given a time threshold *Threshold*, as one where the time difference between an appliance being switched off and subsequently being turned on is greater than *Threshold*. We use a *Threshold* of 60 seconds throughout our analysis. $SEF(A,G,Threshold) =$
$$\dfrac{\sum_{i=1}^{size(G)-Threshold} P(i)=1 \& P(i+1)=0 \& (\sum_{j=1}^{Threshold} P(i+j))=0}{\sum_{i=1}^{size(G)-Threshold} P(i)=1 \& P(i+1)=0}$$

Ground truth was collected by manually labeling the time series data from Reed and PIR sensor.

Accuracy of occupancy prediction can potentially improve if historical occupancy information from the near past is also used. For this purpose, we define Retention Factor (K), corresponding to previous K samples of a given time series. This factor is similar to *threshold* [11], *TD* [8] and *Threshold* [6], used in prior work in a similar context. With sampling
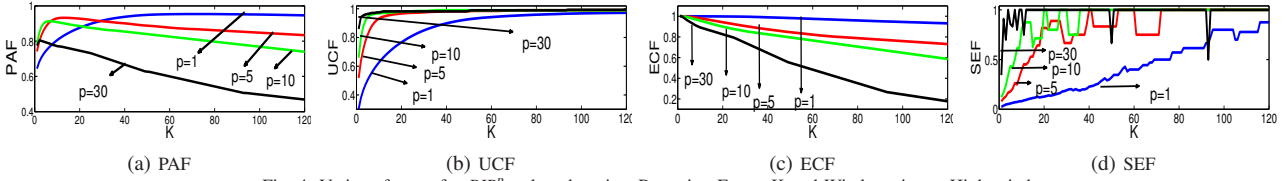
Fig. 4: Various factors for $PIR^p{}_K$ plotted against Retention Factor K and Window size p. Higher is better

frequency of 1 Hz, K corresponds to previous K seconds worth of data. We introduce a notion of downsampled (p-windowed) time series, wherein p denotes the duration of a single sample in the new downsampled series. Given an *input* timeseries (I), corresponding *p-windowed* timeseries will have a high for a p duration sample, if corresponding p samples in I have non-zero high, and low otherwise. Downsampled data allows understanding the influence of using aggregated information, as compared to high resolution input data.

A. Occupancy Prediction Algorithms

We now present analysis for different occupancy prediction algorithms across the 4 metrics defined above. Different metrics calculated using all these algorithms are shown in Table I.

1) Naive PIR (PIR$_N$): This algorithm does not take any history into account and predicts an occupant to be present if motion is detected by PIR sensor and absent otherwise. Due to missed occupancy by PIR sensor, when the occupant was relatively still (some contiguous durations being as high as 7 minutes), PAF and UCF are low. On the contrary, when there is no occupant in the room, PIR sensor will never detect motion, thus resulting in no energy wastage and correspondingly ECF being 1. Since minimal motion and absence of motion leads to appliance toggling, SEF is low. Thus no commercially available motion based switching product uses instantaneous motion information.

2) PIR with Retention Factor (PIR$_K$): predicts occupancy to be true in the current time instance if there was any motion detected by PIR sensor in the last K samples and false otherwise. The intuition behind this algorithm is that the motion may not be detected for K time samples, while the occupant may still being present in her room. Different values of K can account for varying intervals when the occupant's motion is not detected. Figure 4a shows the dependence of PAF on K. As we take more history into account (higher K) our accuracy increases upto a threshold after which it starts decreasing. This threshold is the optimal relevant history (K_{opt}) or the optimum timeout interval for the PIR sensor as per the current occupant. When K is increased beyond (K_{opt}), PAF reduces since it predicts occupant's presence using the motion information from a long time ago while the occupant may have recently left the room.

From Table I and Figure 4b we also observe that UCF is monotonically increasing with K. This monotonic increase is attributed to the fact that higher the history that is taken into account, more likely it is for PIR$_K$ to predict the occupant to be present (even when she may not be present), thus reducing the chances of predicting absence when the occupant may still be present and increasing UCF. For the same reason, ECF

monotonically reduces with increasing K. Figure 4d shows the dependence of SEF on K. As we increase K, the number of switches reduces, thereby also reducing the number of efficient switches. However as we take more history into account, more conservative appliance switching results in higher percentage of efficient switching and correspondingly increasing SEF.

3) p-Window PIR with Retention Factor (PIR$^p{}_K$): This algorithm downsamples the time series to correspond to p seconds for each time instance (PIR$_K$=PIR$^1{}_K$). Window-size, p, is varied as 5, 10, 30 and corresponding results are summarized in Table I and Figure 4. Increasing p corresponds to accounting for higher history and therefore the results are similar to when Retention Factor is used. It is interesting to observe that as we increase p, corresponding peak in PAF is achieved at a lower value of k while the factor p*K_{opt} remains approx. the same. We hypothesize that this factor p*K_{opt} corresponds to specific user characteristics and propose to analyze it with data from ongoing bigger deployment.

4) PIR and Reed based (PIR$_R$): This algorithm is inspired by prior work on occupancy detection using a combination of PIR and Reed sensor [2]. If the door status is open, we consider the room to be occupied. A door close event may correspond to three situations: 1) The main occupant closing the door and leaving the room unoccupied; 2) Someone entered and closed the door while the main occupant was already in her room; or 3) Main occupant closed the room upon entry. We wait for M seconds and observe motion in the same. If no motion is detected we consider it to be the first scenario occuring in case of a door-close event and label the room as unoccupied until next door event. If motion is detected we label the room to be occupied and do not differentiate between second and third scenario. Analysis with varying M is reported in Table I. This approach is biased towards user comfort as it waits for M seconds after a door-close event to label the room as unoccupied resulting in higher UCF and lower ECF, as shown in Figure 5. As we increase M, PAF reduces as there is higher wait involved in predicting user absence but SEF increases since number of inefficient switching events is reduced.

5) HMM based on PIR (HMM): Hidden Markov Models are well suited to applications pertaining to time series (or sequential) data [12]. We train our HMM model based on PIR series and Ground truth series using Baum Welchback algorithm to learn the model parameters such as Emission Probabilities and Transition Probabilities. We chose PIR time-series as the observed sequence and Occupancy time series as the latent sequence to be predicted by the Viterbi algorithm. From Table I we can see that this approach is an improvement
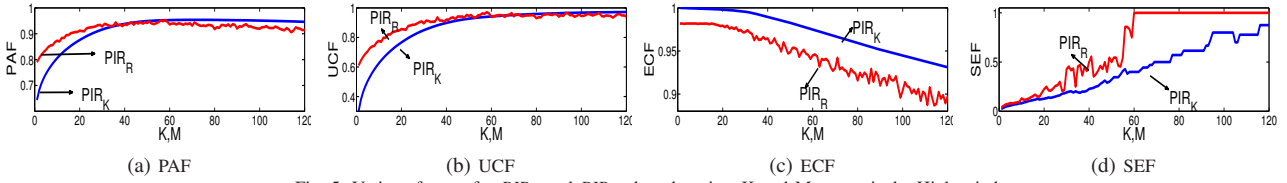
(a) PAF   (b) UCF   (c) ECF   (d) SEF

Fig. 5: Various factors for PIR_K and PIR_R plotted against K and M respectively. Higher is better

TABLE I

Comparison of PAF, UCF, ECF and SEF Across Different Algorithms and Corresponding Optimum K and M Values.

Algorithm	PAF,K,M	UCF,K,M	ECF,K,M	SEF,K,M
PIR_N	.64,NA,NA	.3,NA,NA	1,NA,NA	.02,NA,NA
PIR_K	.95,60,NA	.96,120,NA	1,1,NA	1,120,NA
$PIR_K{}^5$	.95,13,NA	.98,120,NA	1,1,NA	1,66,NA
$PIR_K{}^{10}$	.96,7,NA	.98,120,NA	1,1,NA	1,35,NA
$PIR_K{}^{30}$	.96,2,NA	.99,120,NA	1,1,NA	1,16,NA
HMM	.82,NA,NA	.64,NA,NA	.99,NA,NA	.22,NA,NA
HMM^5	.94,NA,NA	.88,NA,NA	.99,NA,NA	1,NA,NA
HMM^{10}	.97,NA,NA	.99,NA,NA	.95,NA,NA	1,NA,NA
HMM^{30}	.97,NA,NA	.97,NA,NA	.98,NA,NA	1,NA,NA
PIR_R	.95,NA,58	.98,NA,120	.98,NA,1	1,NA,62

over PIR_N. UCF improves significantly since the learned model encodes the fact that a person may be present even though the PIR sensor may not detect motion. This increase in UCF also increases PAF and SEF to a certain extent. However ECF remains close to 1, as was the case with PIR_N.

6) HMM based on p-Window PIR (HMM^p): In this approach, we modified the training timeseries to be *p-windowed* PIR, with different values of p as per Table I (HMM=HMM^1). From Table I we observe that UCF is most affected by this approach and as a result PAF increases and SEF becomes 1. The number of switches are drastically reduced in comparison to earlier approaches and most of the switching events that occur are efficient, since adequate history and transitions are taken into account by HMM model.

Results presented in Table I clearly indicate that simple heuristic based approaches (e.g. PIR_K^p with K_{opt}) provide similar performance across the four defined metrics as complex approaches such as HMM. The distinct advantage of simple approaches lies in their easy interpretability, implementation and reduced training overheads. Additionally, such simple approaches can be easily implemented in low end microcontrollers.

VI. CONCLUSIONS AND FUTURE WORK

In this work we presented some key learning and outcomes from our experiences in developing and deploying sensing infrastructure for Occupancy based Building Management Systems (Ob-BMS). Experiences from building infrastructure deployed across two newly constructed buildings will provide an initial blueprint for planning sustainable buildings with high resolution sensing. We present extensions to existing work in Ob-BMS analytics by adding switching efficiency metric. Analysis of the data collected from pilot deployment shows that simple heuristic based approaches provide similar performance for occupancy based analysis, across the four metrics considered, as complex HMM based approaches.

We propose to extend the work in Ob-BMS analytics using large scale data from ongoing deployment of sensing infrastructure across more than 400 dorm rooms in the new campus of IIIT Delhi. We have also built a custom tag (transmitting information over 802.15.4) that we intend to use to collect ground truth data in the upcoming deployment in one of the research wings at IIIT Delhi. Ongoing efforts towards integrating our sensor deployments with commercial BMS (communicating over BACnet) controlling HVAC systems will help us in performing richer Occupancy based analytics and control.

REFERENCES

[1] U. E. I. Administration. Aeo2011: Annual energy outlook. 2011.

[2] Y. Agarwal, B. Balaji, S. Dutta, R. Gupta, and T. Weng. Duty-cycling buildings aggressively: The next frontier in hvac control. In *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, pages 246 –257, april 2011.

[3] G. Anastasi, F. Corucci, and F. Marcelloni. An intelligent system for electrical energy management in buildings. Intelligent Systems Design and Applications, pages 702–707, 2011.

[4] P. Arjunan, N. Batra, H. Choi, A. Singh, P. Singh, and M. B. Srivastava. SensorAct: A Privacy and Security Aware Federated Middleware for Building Management. In *Fourth ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings*, BuildSys 2012.

[5] S. Dawson-Haggerty, X. Jiang, G. Tolle, J. Ortiz, and D. Culler. sMAP: a Simple Measurement and Actuation Profile for Physical Information. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys, 2010.

[6] D. T. Delaney, G. M. P. O'Hare, and A. G. Ruzzelli. Evaluation of energy-efficiency in lighting systems using sensor networks. In *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, BuildSys '09, pages 61–66, 2009.

[7] M. Evans, B. Shui, and S. Somasundaram. Country report on building energy codes in india. *PNNL*, 177925, 2009.

[8] V. Garg and N. Bansal. Smart occupancy sensors to reduce energy consumption. *Energy and Buildings*, 32(1):81 – 87, 2000.

[9] K. B. Janda. Buildings don't use energy: people do. *Architectural Science Review*, 54(1):15–22, 2011.

[10] K. Lomas, D. Shipworth, E. Shove, M. Kelly, and D. Uzzell. How people use and misuse buildings, 2009.

[11] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, E. Field, and K. Whitehouse. The smart thermostat: using occupancy sensors to save energy in homes. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, pages 211–224, 2010.

[12] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[13] A. Rowe, M. E. Bergeés, G. Bhatia, E. Goldman, R. Rajkumar, J. H. Garrett, J. M. F. Moura, and L. Soibelman. Sensor andrew: large-scale campus-wide sensing and actuation. *IBM J. Res. Dev.*, 55(1&2):66–79, Jan. 2011.

[14] R. Socolow. Saving energy in the home:princeton's experiments at twin rivers. *Ballinger*, 1978.