A

Unsupervised Residential Power Usage Monitoring using a Wireless Sensor Network

Rui Tan, Nanyang Technological University
Dennis E Phillips, Michigan State University
Mohammad-Mahdi Moazzami, Michigan State University
Guoliang Xing, Michigan State University
Jinzhu Chen, Michigan State University

Appliance-level power usage monitoring may help conserve electricity in homes. Several existing systems achieve this goal by exploiting appliances' power usage signatures identified in labor-intensive *in situ* training processes. Recent work shows that autonomous power usage monitoring can be achieved by supplementing a smart meter with distributed sensors that detect the working states of appliances. However, sensors must be carefully installed for each appliance, resulting in high installation cost. This paper presents *Supero* – the first *ad hoc* sensor system that can monitor appliance power usage without supervised training. By exploiting multi-sensor fusion and unsupervised machine learning algorithms, Supero can classify the appliance events of interest and autonomously associate measured power usage with the respective appliances. Our extensive evaluation in five real homes shows that Supero can estimate the energy consumption with errors less than 7.5%. Moreover, non-professional users can quickly deploy Supero with considerable flexibility.

CCS Concepts:•**Networks** → **Sensor networks;**

General Terms: Design, Experimentation, Measurement

Additional Key Words and Phrases: Nonintrusive load monitoring, wireless sensor networks, sensor fusion, unsupervised learning.

1. INTRODUCTION

Appliance-level power usage monitoring can improve the efficiency of electricity use in homes. Research [McMakin et al. 2002] has shown that giving users detailed information about their energy usage fosters conservation. Moreover, the information enables utility companies to assess the electrical efficiency of homes by data mining. For instance, by comparing the power usage of appliances across different homes, we can rank the efficiency of the appliances and inform their owners to guide the replacement or repairs of dated and inefficient appliances.

Previous systems for appliance-level power usage monitoring can be broadly classified into two categories. The first category, *direct sensing*, measures per-appliance power usage by smart plugs [Jiang et al. 2009a] and smart switches [Insteon 2015]. As smart plugs are placed between the appliances and power outlets, they cannot be

The first two authors contribute equally.

Part of this work was published in The 11th IEEE International Conference on Pervasive Computing and Communications (PerCom 2013).
This work was supported in part by U.S. National Science Foundation under grants CNS-1218475 and CNS-0954039 (CAREER), and in part by a Start-up Grant at Nanyang Technological University.

Author's addresses: R. Tan, N4-02C-85, 50 Nanyang Avenue, Singapore 639798; D. Phillips, G. Xing, M. Moazzami, and J. Chen, 428 South Shaw Lane, East Lansing, MI 48824, USA.

DOI: http://dx.doi.org/10.1145/0000000.0000000

used for appliances hardwired to power lines, such as ceiling lights. Replacing normal wall switches with smart switches needs cumbersome hardwiring and possibly expensive modifications to walls. In light of the installation overhead, direct sensing is suitable only when permanent monitoring is desired. The permanent appliance-level monitoring capability may be widely available in future smart homes due to adoption of smart switches, plugs, and appliances. However, the complete realization of this vision may take a considerable time period (up to decades). At present and in the near future, a system for swift one-off deployments for identifying power wastage and diagnosing inefficient appliances is still desirable. The second category, *indirect sensing*, is less intrusive as it infers the working states and energy consumption of individual appliances by detecting their power usage patterns [Hart 1992; Patel et al. 2007] or ambient signals they emit during operation [Gupta et al. 2010; Kim et al. 2009]. However, these techniques require either labor-intensive *in situ* supervised training, due to their dependency on the appliance characteristics [Hart 1992] and electrical wiring [Patel et al. 2007; Gupta et al. 2010], or careful sensor installation for each appliance [Kim et al. 2009], leading to high installation cost and reduced usability.

In this work, we aim to design a residential power usage monitoring system that (i) uses only inexpensive and easy-to-install sensing devices, (ii) can be deployed by non-professional users with straightforward instructions, and yet (iii) can work effectively based on a small amount of easily obtained prior information without resorting to supervised *in situ* training. Such a system must automatically detect the events of interest, autonomously associate the events with the correct appliances, and finally infer the power usage of each appliance. This brings three key challenges. First, inexpensive sensors typically have limited sensing capabilities; hence, they can produce false alarms or miss important events of monitored appliances. Second, when sensors are installed in an *ad hoc* manner, multiple sensors may detect the same event, and it becomes difficult to associate the event with the appliance that is the source of the event. Lastly, to make the system practical, we must minimize the amount of prior information that users will need to collect.

This paper presents the design and implementation of *Supero* – a system for unsupervised power monitoring. Supero utilizes a smart meter to measure real-time total household power consumption and inexpensive light and acoustic sensors that are deployed in an *ad hoc* manner to detect interesting events of appliances. It uses multi-sensor fusion to correlate data collected by power, light, and acoustic sensors and reduce possible sensing errors. By using advanced unsupervised clustering algorithms, Supero analyzes the signal signatures of different appliances and identifies the events generated by the same appliance. Moreover, Supero autonomously associates the classified events with the appliances through an optimization framework that accounts for environmental factors such as light signal propagation. Given a small amount of easily obtained prior information such as sensor-appliance distances and rated powers of a small subset of the appliances, our unsupervised algorithms work together to disaggregate the total household energy consumption into usage by the individual appliances. To the best of our knowledge, Supero is the first practical *ad hoc* sensor system that can accurately monitor appliance power usage without supervised training. Supero aims at swift one-off deployments for power usage diagnosis over short time periods (e.g., a few days to weeks). As such, there should be little concern about user privacy or any negative visual impact of the sensor installation.

We prototyped Supero using a network of TelosB/Iris motes [Memsic Corp. 2011] and a smart meter, and evaluated it in five real homes of different sizes and with different characteristics of electricity consumption. A 10-day evaluation in an apartment shows that Supero can estimate the energy consumption with errors less than 7.5%. Our results also demonstrate that Supero can be quickly deployed by non-professionals with considerable flexibility.

The major contribution of this paper is the application of advanced signal processing and unsupervised learning algorithms to address various practical challenges and realize a novel *ad hoc* sensor system for appliance-level power usage monitoring. As a scientific value of this paper, the evaluation results extend our understanding on the strategy and flexibility in sensor installation beyond the state-of-the-art strategy of installing sensors dedicated to specific appliances.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the overview of Supero. Section 4 presents the event detection and multi-sensor fusion algorithms. Section 5 and Section 6 present the unsupervised event clustering and autonomous appliance association algorithms, respectively. Section 7 discusses estimating the power consumption of a class of high-power heating appliances. Section 8 and Section 9 present our system implementation and evaluation results, respectively. Section 10 concludes this paper.

2. RELATED WORK

This section discusses representative indirect sensing approaches for appliance power usage monitoring, and identifies their differences from Supero. Early work in this area [Hart 1992; Drenker and Kader 1999; Farinaccio and Zmeureanu 1999] utilizes per-appliance power operating characteristics, measured at power panels, to disaggregate the total energy consumption. These approaches need either *in situ* training [Hart 1992; Farinaccio and Zmeureanu 1999] or a comprehensive database of *a priori* power characteristics of appliances [Hart 1992; Drenker and Kader 1999]. Jiang et al. [Jiang et al. 2009b] present the experience of monitoring the power usage of a laboratory using smart plugs [Jiang et al. 2009a] and light sensors. In [Jung and Savvides 2010], binary sensors are used to help deploy power meters to estimate energy breakdowns in a building. Both of the studies [Jiang et al. 2009b; Jung and Savvides 2010] exploit the tree topology of the subject power supply system. Patel et al. [Patel et al. 2007] detect and classify electrical events based on transient noises generated by the appliances. Their transient signatures are heavily influenced by the electrical wiring, which results in the need for *in situ* training. In [Gupta et al. 2010; Taysi et al. 2010], appliances are recognized based on their electromagnetic interference and acoustic signals. Similarly, their work requires labor-intensive *in situ* training. A typical training process involves switching on/off appliances, and collecting and labeling signals. Recently, Ho et al. [Ho et al. 2011] use a thermal camera to detect the on/off states of appliances and infer the per-appliance energy consumption. The thermal camera can be hard to install and can raise privacy concerns in residential environments.

ViridiScope [Kim et al. 2009] is an appliance-level power usage monitoring system closest in design to Supero. It features an autonomous regression framework that can calculate per-appliance energy consumption based on the appliances' working states and the total household power trace. ViridiScope detects the working states by carefully installed sensors. For instance, light and magnetic sensors are placed in close proximity to or attached to each appliance, and must not be triggered by other appliances to ensure correct power estimation. Such an installation of the sensors is hard for difficult-to-access appliances such as ceiling lights. In Supero, due to the use of unsupervised learning and novel sensor fusion/association techniques, sensors are not dedicated to specific appliances, and so can be deployed in an *ad hoc* manner, leading to significantly lower installation costs. ViridiScope uses two acoustic sensors to monitor a refrigerator compressor and reject ambient noises. In this paper, we propose a systematic approach for monitoring a range of acoustic appliances, which jointly processes the data from all acoustic sensors to detect the appliances' working states.

3. OVERVIEW OF SUPERO

3.1. Design Objectives and Challenges

The goal of Supero is to produce appliance-level electricity usage reports over specific time durations (e.g., a specified week) in a household. A report includes the energy consumption of particular appliances, as well as when they were turned on/off. Supero aims at achieving the following design objectives:

— It should be possible to deploy the sensors in an *ad hoc* and *non-intrusive* manner. A non-professional should be able to deploy battery-powered wireless sensors with intuitive instructions such as "place a light sensor with unobstructed view of the light" and "place an acoustic sensor on top of the microwave."
— We aim to reduce needed efforts for system configuration by avoiding the use of labor-intensive training and extensive user inputs.
— Supero should be able to operate for a long enough time period (e.g., a few weeks) without changing the sensors' batteries, such that the generated report is meaningful and informative enough for identifying wasteful energy usage and diagnosing efficiency problems in appliances.

Four major challenges are brought by the above design objectives:

— In an *ad hoc* deployment, a sensor may pick up signals emitted by multiple appliances, which can make it difficult to pinpoint the appliance that is consuming power. For instance, a light sensor can sense light emitted by various sources, and an acoustic sensor in the kitchen can hear sounds from the exhaust fan, waste disposer, microwave, etc.
— Without careful installation, sensors typically suffer from sensing errors caused by ambient noises and human activities. For instance, light sensors can report false alarms when nearby window blinds are opened, and acoustic sensors may pick up sounds such as human conversations that are unrelated to power consumption.
— Without *in situ* system training, unsupervised learning often requires more prior information than supervised learning. In Supero, we strive to reduce the burden on users to obtain the prior information required, while maintaining good monitoring accuracy.
— To extend the system lifetime, wireless sensors should adopt lightweight sensing algorithms and minimize the data transmissions, which, however, raises challenges for accurate monitoring of appliance working states.

3.2. Motivating Observations

To meet the aforementioned objectives, Supero utilizes a household power meter and a small number of inexpensive light and acoustic sensors that are deployed in the home in an *ad hoc* manner. Based on an unsupervised approach, it does not require any *in situ* system training. Rather, it leverages a small amount of prior information that can be easily obtained by non-professional users. We now discuss several important observations that motivate our approach.

Real-time total household power metering. Nowadays, the real-time total household power consumption can be easily measured by installing a commercial off-the-shelf smart meter (e.g., TED [Detective 2015] and AlertMe [Alertme 2015]) on the main circuit panel. These meters are inexpensive and most of them can be easily installed without hardwiring with the power lines. Moreover, as the coverage of smart grid increases, the real-time total household power readings are increasingly available to the homeowners, without resorting to a personal smart meter.

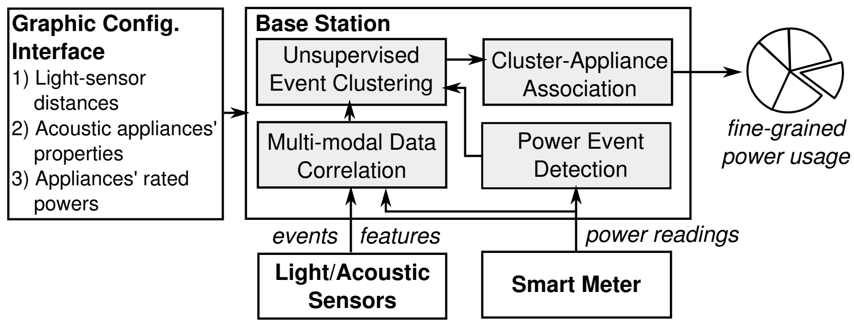


Fig. 1. The Supero architecture.

Sensing modalities. According to a survey of U.S. Department of Energy [U.S. DoE 2006], the average distribution of electricity consumption in household is: heating 24%, lights 24%, air conditioners 20%, refrigerators 15%, dryers 9%, and electronics 9%. As most heating appliances consume substantially more power than other appliances, their consumption trace often can be identified from the real-time total household power readings. Most lights, air conditioners, refrigerators and dryers emit light and acoustic signals. As a result, on average, more than 90% power consumption of a typical household can be captured by a combination of a smart meter and a set of light and acoustic sensors.

Useful prior information. To avoid expensive *in situ* system training, Supero leverages unsupervised learning techniques and a small amount of prior knowledge including rough sensor-appliance distances and the rated powers of a small subset of appliances. As the light/acoustic signal decays with the distance from the source appliance, the distances between sensors and appliances provide important hints for associating the detected events to the right appliances. Moreover, although the rated power of an appliance often has small discrepancy with the actual power consumption, it helps identifying the consumption trace of a small number of difficult-to-detect appliances from the household power readings. Rated powers are often available from the labels on the appliances or the user manuals. Moreover, there exist a few publicly available databases (e.g., [TPCDB 2015]), which provide rated power based on the appliance brand and model.

3.3. System Architecture

Supero consists of a number of wireless sensors distributed in the home being monitored, a smart meter, and a base station for receiving information from the sensors and the smart meter. In this work, we only consider light and acoustic sensors while other sensing modalities such as infrared can be easily incorporated by Supero. Fig. 1 illustrates the *two-tiered* architecture of Supero. In the first tier, sensors sample signals and detect events that are possibly caused by switching appliances on/off. On the detection of an event, a sensor extracts various features of the event and sends an event message to the base station. Further details of the first tier will be presented in Section 4. The base station provides a graphic configuration interface that allows user to input prior information such as sensor-appliance distances and appliances' rated powers. Based on the collected data and the prior information input by the user, the base station executes the following second-tier algorithms periodically (e.g., every week) or upon an electricity usage report request:

Multi-modal data correlation. The base station correlates sensor events and power readings to differentiate between true appliance events and false alarms unrelated to power consumption. (Section 4.4)

Unsupervised event clustering. Leveraging unsupervised clustering algorithms, we can classify the events generated by an appliance into the same cluster, and estimate the power consumption of the appliance by correlating the events with measurements by the smart meter. (Section 5)

Autonomous event-appliance association. Supero associates the classified events with their appliances based on features of the events and the prior information. It then calculates the energy consumption of each appliance. (Section 6)

We note that sufficient appliance events are needed for accurate event clustering and event-appliance association. Thus, Supero needs to collect data for a sufficient time period (e.g., one or more days) before it can generate a latest electricity usage report. Therefore, Supero is not designed to achieve real-time load disaggregation that estimates the instantaneous powers of appliances in real time (e.g., every several minutes). However, Supero can provide an appliance-level electricity usage report at a fine time granularity. For instance, it can report the energy consumption of an appliance within a short time duration (e.g., several minutes) based on the event clustering and event-appliance association results over the day that includes the short time duration.

4. EVENT DETECTION AND DATA CORRELATION

4.1. Light Event Detection

Light sensors sample light intensity periodically ($4\,\text{Hz}$ in our implementation) and detect light events by an *exponential difference filter* (EDF), which is a lightweight and yet effective detection algorithm. By denoting $x[n]$ as the sensor reading at time step n, the exponential moving average, denoted by $\bar{x}[n]$, is computed by $\bar{x}[n] = \alpha \cdot x[n] + (1 - \alpha) \cdot \bar{x}[n-1]$, where $\alpha \in (0, 1)$. By setting $\alpha = \alpha_s$ or $\alpha = \alpha_l$ where $\alpha_s > \alpha_l$, we have the *short-term* and *long-term* moving averages denoted by $\bar{x}_s[n]$ and $\bar{x}_l[n]$. The changes of $\bar{x}_s[n]$ and $\bar{x}_l[n]$ capture the transient light changes and natural ambient light dynamics, respectively. Given two positive thresholds η_L and τ, the sensor counts the number of continuous samples satisfying $|\bar{x}_s[n] - \bar{x}_l[n]| \geq \eta_L$ and raises a detection once the count exceeds τ. The sign of $(\bar{x}_s[n] - \bar{x}_l[n])$ indicates whether the appliance is turned on or off. Whenever the sensor raises a detection, it reports a *light event* message including current reading and the two averages. Moreover, it sets $\bar{x}_l[n] = \bar{x}_s[n]$ to quickly adapt the long-term average to the most recent light intensities. The sensor maintains a Gaussian noise model based on the recent measurements when $|\bar{x}_s[n] - \bar{x}_l[n]| < \eta_L$. The threshold η_L is continuously updated according to the noise model to achieve a low false alarm rate, e.g., 5%. The settings of α_s, α_l and τ will be discussed in Section 8. Fig. 2 shows the operation of the EDF on the readings of a photodiode when two lights are turned on/off and a person moves around. It can be seen that the light events can be accurately detected and the human movements do not trigger false alarms. Light sensors may still pick up events unrelated to power consumption (which we refer to as *non-power event*s), such as those caused by human movements and the opening/closing of window blinds, which will be identified by a multi-modal data correlation technique given in Section 4.4 and then discarded.

4.2. Acoustic Event Detection

A challenge in acoustic sensing is that a high sampling rate is often required to extract event features. Supero adopts a duty-cycled and adaptive sampling scheme to reduce the energy consumed in the sampling and computation. For each second, an acoustic sensor is active for 0.08 seconds only. Initially, it samples the signal at $1\,\text{kHz}$ when it

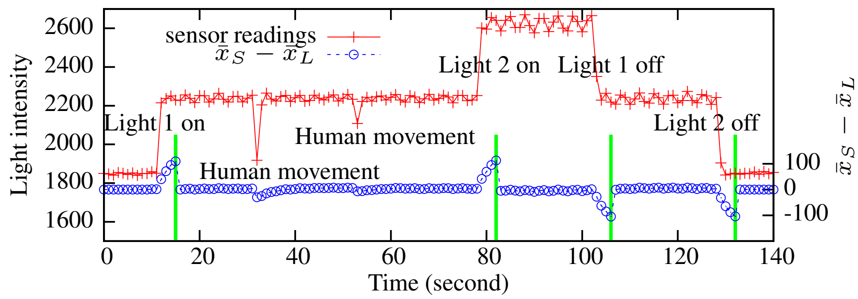


Fig. 2. EDF result on light readings sampled at 4 Hz. Vertical lines represent detections. A person passes by Light 1 at the 31st and 53rd seconds.

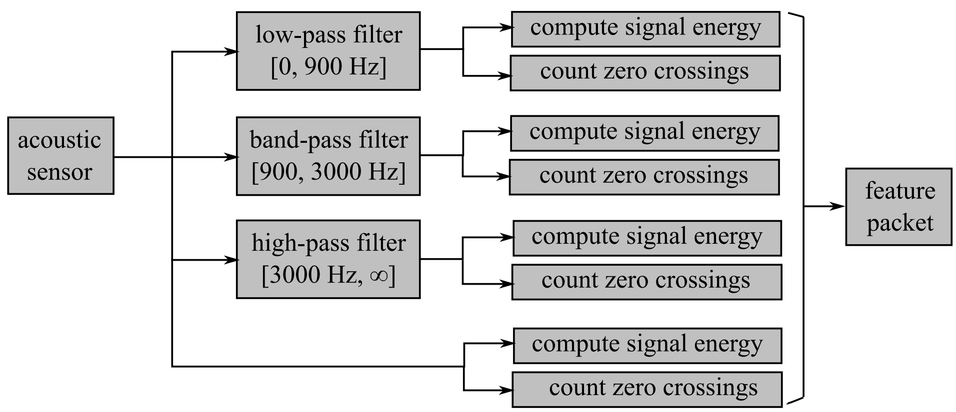


Fig. 3. Acoustic signal is separated into three bands using lattice wave digital filters for feature extraction.

is active. If the signal energy exceeds a threshold η_A, the sensor switches to a high sampling rate of 12.5 kHz to capture more details of the potential event. As shown in Fig. 3, we use three lattice wave digital software filters to decompose the signal into low-pass, band-pass, and high-pass components. The passbands of the three filters are $[0, 900\,\text{Hz}]$, $[900\,\text{Hz}, 3000\,\text{Hz}]$, and $[3000\,\text{Hz}, \infty)$, respectively. The signal energy and zero-crossing counts of the signals in the whole band and the three subbands are computed as acoustic features and transmitted to the base station. The sensor remains in the fast sampling mode as long as the signal energy is above η_A. We set a low threshold η_A conservatively such that the acoustic sensors will not miss any sounds generated by an appliance. Note that different from a *light event*, that refers to the switching on/off of a light, an *acoustic event* refers to the sound heard by a sensor. Therefore, the sensor will continuously report acoustic events while the sound persists. We refer to the switching or phase change of an acoustic appliance as an *acoustic transition*. Owing to intrinsic complexity of the acoustic modality, acoustic transitions are detected by advanced learning algorithms running on the base station, as we will discuss in Section 5.2.

4.3. Power Event Detection

As the total power consumption is critical for identifying appliance events and estimating per-appliance consumption, real-time power readings by the smart meter are transmitted to the base station for storage. Moreover, the base station applies EDF to detect rapid increases and drops in the power measured. The thresholds in the EDF are tuned in offline experiments such that power changes as small as $50\,\mathrm{W}$ can be always detected. In this paper, we assume that the appliances are not duty-cycled at high rates, except those explicitly specified. In Section 7, we develop an approach for monitoring high-power duty-cycled appliances (e.g., stove burner) and discuss how to integrate the approach with Supero.

4.4. Multi-modal Data Correlation

Because of their limited sensing capability and the complexity of home environments, the sensors can easily raise false alarms or miss important on/off events of appliances. For instance, opening/closing a window blind can trigger the nearby light sensors, and human conversations may trigger the acoustic sensors. To deal with these sensing errors, we present a two-tiered fusion approach to correlate the light/power events and acoustic transitions reported by different sensors. The first tier uses a short moving window to correlate the events/transitions reported by multiple sensors of the same modality. The events/transitions falling into the same window are regarded as generated by the same source. This is equivalent to an OR-rule for decision fusion and can greatly reduce the overall miss rate. The second tier correlates the results of the first tier with readings by the smart meter to remove false alarms. Specifically, if the change in power on an event/transition is smaller than a conservatively low threshold (e.g., $5\,\mathrm{W}$), the event/transition will be discarded. The evaluation in Section 9 shows that this approach is effective in removing sensor false alarms.

5. UNSUPERVISED EVENT CLUSTERING

A novel feature of Supero is that it automatically classifies the detected events and associates them with the right appliances, without any *in situ* system training. This section presents our unsupervised event clustering algorithms. We first define the following notation:

— The appliances that cannot be easily or reliably detected by light and acoustic sensors are referred to as *unattended appliance*s (e.g., rice cookers). A power event detected by EDF is considered caused by an unattended appliance if there is no simultaneous light event or acoustic transition. Such power events are referred to as *unattended event*s. N_L and N_A are the total numbers of light and acoustic sensors. M_L, M_A, and M_U are the total numbers of light, acoustic, and unattended appliances, respectively. Δ_k denotes the absolute power change on the k^{th} light/power event or acoustic transition.

— x_i is the *feature* of sensor i in an event. For the light modality, x_i is the absolute change of light intensity, which can be calculated from the current reading and the long-term average; for the acoustic modality, x_i includes signal energies and zero-crossing counts in the subbands; for unattended power events, by letting the index of the smart meter be 0, we have $x_0 = \Delta_k$. For the light and acoustic modalities, the *feature vector* is $X = [x_1, x_2, \ldots, x_N]^{\mathrm{T}}$, where $N = N_L$ or N_A.

5.1. Light Event Clustering

Because of the *ad hoc* deployment approach, the signal emitted by an appliance can be sensed by multiple sensors. Moreover, according to the spatial distribution of the sensors/appliances, the set of sensors that can detect an appliance is generally different for each appliance. However, the feature vectors of the events caused by the same appliance are clustered in the feature space. Fig. 4 shows the feature vectors measured by two light sensors when three standing lights nearby the sensors were turned on and off. We can clearly see that the feature vectors are clustered together.

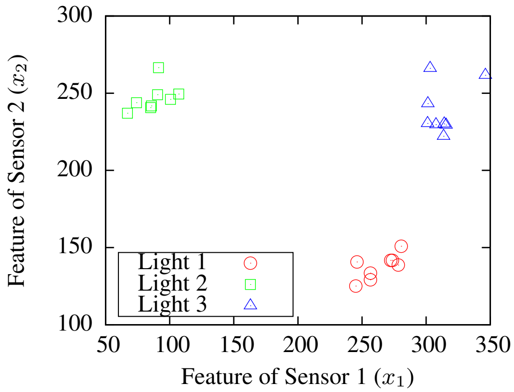


Fig. 4. Light feature vectors of two sensors.

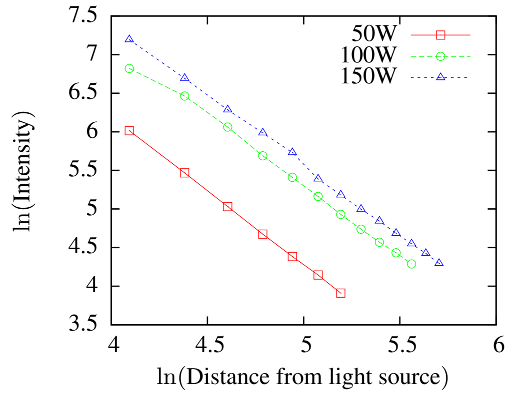


Fig. 5. Light intensity vs. distance (cm) in log-scale.

The light event features will be clustered into M_L clusters. The Euclidean distance between two feature vectors can be small when non-zero vector entries are measured by completely different light sensors, leading to potentially false clustering results. To solve the problem, Supero adopts a novel dissimilarity metric that incorporates identity information of the sensors. Let $b_{k,i} \in \{0, 1\}$ denote the detection decision made by light sensor i regarding event k, where $b_{k,i} = 1$ means that sensor i detects an on/off event of some appliance. The *decision vector*, denoted by B_k, is given by $B_k = [b_{k,1}, b_{k,2}, \ldots, b_{k,N_L}]^{\mathrm{T}}$. The dissimilarity between two decision vectors B_k and B_j is defined as $d(B_k, B_j) = \sum_{i=1}^{N_L} b_{k,i} \oplus b_{j,i} - \sum_{i=1}^{N_L} b_{k,i} \cdot b_{j,i}$, where $\oplus$ represents exclusive OR, $\sum_{i=1}^{N_L} b_{k,i} \oplus b_{j,i}$ is the number of sensors that can only detect either event k or j but not both, and $\sum_{i=1}^{N_L} b_{k,i} \cdot b_{j,i}$ is the number of sensors that can detect both events k and j. Hence, $d(B_k, B_j)$ quantifies the net difference between the sets of sensors observing the two events. By denoting $\|X_k - X_j\|$ as the Euclidean distance between the feature vectors X_k and X_j for the events k and j, the new dissimilarity metric is defined as

$$d(X_k, X_j) = \begin{cases} \|X_k - X_j\|, & d(B_k, B_j) < d_0, \\ \|X_k - X_j\| + \delta, & d(B_k, B_j) \geq d_0, \end{cases} \tag{1}$$

where d_0 is a threshold and δ is a large constant that can separate the feature vectors observed by very different subsets of sensors into different clusters. In our implementation, we set $d_0 = 2$, i.e., two feature vectors should be classified into two distinct clusters if the number of sensors that can only detect the first event is two more than that for the second event. Supero adopts a merging-based clustering algorithm [Duda et al. 2012], which is applicable to nonlinear dissimilarity measures, to group the feature vectors into M_L clusters. Because of space limitation, here we omit the details of the algorithm, which can be found in [Duda et al. 2012, p. 552].

Our experience shows that, the clusters with a small number of feature vectors often affect the accuracy of clustering results. To improve the robustness of clustering, we detect outliers as follows. If the size of a cluster is smaller than a small threshold, its member feature vectors are regarded to be outliers, which are discarded and then the clustering algorithm is re-executed. Outliers are produced by unidentified false alarms and rarely used appliances and hence removing them has little impact on the accuracy of overall energy consumption estimation. In our implementation, we set the outlier cluster size to be 2, i.e., we ignore the appliances that generate two or fewer events in a long period.

5.2. Acoustic Event Clustering and Transition Detection

A challenge of acoustic event clustering is that many appliances, such as multi-speed fans, have multiple phases of operation. Unfortunately, for many appliances, their number of phases cannot be easily determined by the user. For instance, refrigerators have different phases depending on the brand/model and when they were made. Moreover, the number of actually used phases of an appliance, such as multi-speed fans, strongly depends on the habit of the user and is, therefore, unpredictable. The overlaps between sounds from different appliances and noises (e.g., shower and water flush) further result in an unpredictable number of acoustic patterns. Consequently, it is infeasible to assume a known and fixed number of clusters for the collected acoustic events. We propose the following approach based on advanced pattern recognition algorithms to address the above challenges.

To reduce the computational overhead in clustering, Supero first applies principal component analysis (PCA) to reduce the dimensionality of the feature vectors. For instance, in one of our experiments, to keep a 99% sample variance, the dimensionality can be reduced from 40 to 8 when 5 acoustic sensors are deployed. Supero then estimates the number of clusters as $k_{\text{opt}} = \arg\max_k \frac{\det(S_b(k))}{\det(S_w(k))}$ [Strehl and Ghosh 2003], where $S_b(k)$ and $S_w(k)$ are the between-cluster and within-cluster variance matrices when the specified cluster number is k. For each given k, the k-means algorithm is executed to cluster the events and calculate $S_b(k)$ and $S_w(k)$. Based on the clustering results with $k = k_{\text{opt}}$, Supero detects acoustic transitions as the transitions between clusters over time. Specifically, by dividing time into small windows, we count the number of feature vectors belonging to each cluster and identify the cluster with the largest number of feature vectors (referred to as major cluster) in each small time window. Then, the edges between two consecutive windows with different major clusters are detected as the acoustic transitions.

We have developed a method to choose the window size to reduce the average misclassification rates in the windows. Specifically, the window size is selected to minimize the product of the number of acoustic events and the sum of misclassification rates in all windows. The misclassification rate in a window is the ratio of the number of features that do not belong to the major cluster in the window to the total number of features in the window. The rationale of jointly considering the number of acoustic events in the minimization objective is as follows. The misclassification rate typically decreases with the window size. Therefore, only minimizing the sum of misclassification rates will mostly result in an unreasonable small window size.

As a simple example, Fig. 6 shows the results of using an acoustic sensor to detect the phase changes of a 3-speed fan. Specifically, we place the acoustic sensor close to the fan to record the 8-dimensional acoustic features as described in Section 4.2. After applying PCA to the collected features, we have the major PC as shown by the points in Fig. 6(b). Fig. 6(a) shows the $\frac{\det(S_b(k))}{\det(S_w(k))}$ versus the number of clusters k. From the figure, we can see that, with $k = 3$, the ratio $\frac{\det(S_b(k))}{\det(S_w(k))}$ is maximized. Thus, the number of clusters is estimated as 3. This indicates that the fan has three phases, which is consistent with the number of speed levels used in the experiment. The k-means algorithm with $k = 3$ classifies the event features into three clusters, which are represented by different colors in Fig. 6(b). Finally, the vertical dashed lines in Fig. 6(b) represent the transitions between phases as detected by our system.

5.3. Unattended Power Event Clustering

For the unattended power events, Supero adopts the Euclidean distance between the power changes as the dissimilarity metric, and applies the k-means algorithm to clus-

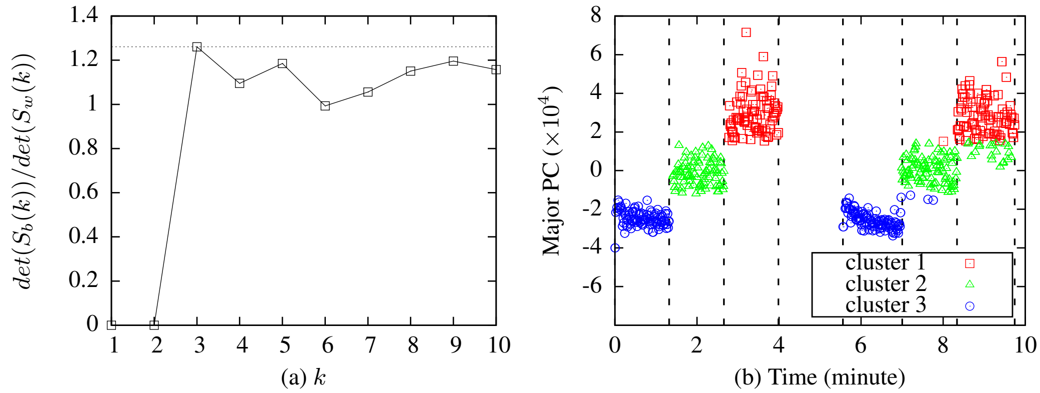


Fig. 6. Acoustic event clustering and transition detection for a 3-speed fan. (a) The number of phases is identified as three; (b) Clustering and transition detection results, where Y-axis is the major principle component (PC) and vertical lines represent the detected acoustic transitions.

ter the events into M_U clusters. To simplify the discussion, in this paper, we assume that the unattended appliances are not multi-phase. However, by extending the approach developed for the acoustic modality, Supero can be readily extended to address multi-phase unattended appliances.

6. AUTONOMOUS APPLIANCE ASSOCIATION

Event clustering does not tell us which appliance triggers the events in a cluster. This section associates the right appliances with the clusters by exploiting the correlations between event features, sensing models, and other prior information. Based on the association results, each appliance's energy consumption can be calculated either by integrating power over time or by a regression approach [Kim et al. 2009] for improved robustness.

6.1. Light Cluster-Appliance Association

The decay of light intensity follows the power law, which can be exploited to associate light appliances with clusters. We conducted extensive measurements to verify the decay model in various household environments. Fig. 5 reports one set of results, which plots the light intensity readings of a sensor versus the line-of-sight distance from a light bulb in a $5 \times 3.2\,\mathrm{m}^2$ living room. Both axes of Fig. 5 are in log-scale. The linear relationship in the figure conforms to the power law. Moreover, at a certain distance, the sensor reading is proportional to the power of the light bulb. Therefore, we assume that the intensity measured by sensor i, denoted by y_i, is given by $y_i = \beta \cdot P_j \cdot d_{ij}^{-\alpha}$, where P_j is the power of light j, d_{ij} is the line-of-sight distance between sensor i and light j, α is the path loss exponent of the power law, and β is a scaling factor. α and β can vary with the deployment environment, but have bounded ranges. For instance, α typically ranges from 2.0 to 5.0.

The association between clusters and lights is represented by a matrix $A = [a_{m,j}]_{M_L \times M_L}$. If cluster m is associated with light j, $a_{m,j} = 1$; otherwise, $a_{m,j} = 0$. Let μ_m denote the average of the feature vectors in cluster m. Hence, the i^{th} component of μ_m, denoted by $\mu_{m,i}$, is the average change of light intensity measured by sensor i when the corresponding light is turned on and off. By denoting R_m as the set of sensors that make positive decisions in cluster m, we define the error caused by associating cluster m with light j as $e_{m,j} = \sum_{i \in R_m} \left|\beta \cdot P_m \cdot d_{i,j}^{-\alpha} - \mu_{m,i}\right|$, where P_m is the power of the light that generates the events in cluster m. We estimate P_m as the median value of the

Algorithm 1 Acoustic Transition-Appliance Association Algorithm

Input: acoustic transition set $\mathcal{T}$, non-primarily monitored appliance set $\mathcal{A}$
Output: acoustic transition-appliance association

$C = \emptyset$
for transition k in $\mathcal{T}$ **do**
  find sensor i with the largest absolute change of signal energy in k
  if sensor i is a primary sensor **then**
    associate k with the corresponding primarily monitored appliance
  else
    $C = C \cup \{k\}$
  end if
end for
cluster the transitions in C using k-means algorithm based on their absolute power changes, with $|\mathcal{A}|$ as the number of clusters
sort clusters according to their centers
sort appliances in $\mathcal{A}$ in terms of power
associate the sorted clusters with the appliances in $\mathcal{A}$ in order

absolute power changes (i.e., Δ_k) of the events in cluster m. The total error is defined as $E(\alpha, \beta, A) = \sum_{\forall m, \forall j} a_{m,j} \cdot e_{m,j}$. Based on this error metric, we formulate the problem as:

Light Cluster-Appliance Association Problem. *Find α, β and A to minimize $E(\alpha, \beta, A)$, subject to that $\forall m, \sum_{\forall j} a_{m,j} = 1$ and $\forall j, \sum_{\forall m} a_{m,j} = 1$.*

The constraint in the above formulation means that A is a one-to-one mapping. To solve the above problem, we first fix α and β and then find A to minimize $E(\alpha, \beta, A)$ under the constraint, which is a *linear assignment problem* [Burkard et al. 2012]. We employ the Hungarian algorithm [Burkard et al. 2012] with a time complexity of $O(M_L^4)$ to solve this sub-problem. Henceforth, the final solution can be found by enumerating α and β in their possible ranges. Therefore, Supero automatically learns the values of α and β in a specific deployment such that the association minimizes the discrepancy between the measurements and the decay model. This is desirable since determining their exact values through *in situ* calibration would be labor-intensive.

The association algorithm requires the sensor-appliance distances, which can be estimated by a sonic laser tape, arm span, or even rough visual estimation. As long as the order of the distances is preserved in the estimation, the association result will most likely remain unaffected. Hence, the association algorithm is robust to small errors in the distance estimation. In the evaluation reported in this paper, all the distances were visually estimated and we do not observe any association errors caused by inaccuracies of the visual estimation.

6.2. Acoustic Transition-Appliance Association

Although acoustic signals follow power law decay, they are typically side effects of the appliances' operation. Hence, the scaling factor β can vary significantly across different acoustic appliances and the association algorithm developed in Section 6.1 is not applicable to the acoustic modality. We now propose a heuristic association approach to solve the problem. Sensor i is defined as the *primary sensor* of appliance j if the absolute change of signal energy of sensor i is always the largest when appliance j changes its state, and must not be the largest when any other appliance changes state. Appliance j is defined as a *primarily monitored appliance*. The complement set of primarily monitored appliances comprises *non-primarily monitored appliance*s. Different from a *dedicated sensor* that can only sense one appliance, a *primary sensor* can sense multiple appliances. The primary sensors can be identified by user intuition based on the sensor and appliance locations. When a sensor cannot be accurately classified as a primary sensor, it can be conservatively excluded from the set of primary sensors. The pseudo code of the association is given in Algorithm 1. The algorithm first identifies the acoustic transitions generated by primarily monitored appliances and directly associates the transitions with the appliance (Line 3 to 5). The remaining acoustic transitions are associated with the non-primarily monitored appliances according to power (Line 10 to 13). Note that the extra prior information required by Algorithm 1 is the order of the non-primarily monitored appliances with respect to power, which is used in Line 12.

6.3. Unattended Appliance Association

The power of the appliance that generates the unattended power events in cluster m, denoted by P_m, is estimated as the median value of the absolute power changes of those events. Supero associates the clusters with appliances by matching P_m's with the rated powers. The association is a linear assignment problem [Burkard et al. 2012], which aims to minimize the total error of power. The error of associating cluster m with appliance j is defined as $e_{m,j} = |P_m - P_j^*|$, where P_j^* is the rated power of j. This optimization-based association is resilient to small deviations between the true working power and rated power. We create a virtual *background* appliance to represent all the appliances that consume little but variable power, such as laptop computers. The association error of the background appliance is always zero, i.e., $e_{m,j} = 0$ for any cluster m. In other words, the background appliance can be associated with any cluster such that it will not affect the association of other unattended appliances.

For various acoustic appliances that have complex signal patterns, the sensors may miss important events. For instance, the sound of a water boiler becomes detectable in a couple of seconds after being turned on. The delayed acoustic event may be falsely removed by the data correlation due to little associated power change. To address the issue, we treat such an acoustic appliance as an unattended appliance as well and then merge the acoustic transitions and power events. Supero is expected to become more robust to event misses if more acoustic appliances are jointly monitored and their rated powers are provided.

7. DUTY-CYCLED HEATING APPLIANCES

As discussed in Section 3.2, heating appliances such as stove burner and oven are major electricity consumer in homes. Most modern heating appliances duty-cycle to achieve the desired heat level. For instance, the top part of Fig. 7 shows the total household power readings when a GE JB710ST2SS burner is working. As the cycle can be short (e.g., several seconds), the EDF-based detector discussed in Section 4.3 may have poor performance. In this section, we propose a new approach to detect the duty-cycling pattern from the total power readings and calculate the related energy consumption.

Duty-cycled appliance rapidly switches between on and off, causing large variation in power readings. Thus, we detect the duty-cycling pattern based on the standard deviation of the windowed power readings. By denoting P and $\gamma \in (0,1)$ as the power and duty cycle of the appliance, the standard deviation of the power readings can be derived as $P\sqrt{\gamma - \gamma^2}$. We choose a threshold of $P\sqrt{0.05 - 0.05^2}$ by conservatively assuming that the duty cycle is greater than 5%. When P is unknown, we can choose a default value of $1.5\,\text{kW}$ for P because most duty-cycled heating appliances have a rated power around $1.5\,\text{kW}$ [TPCDB 2015]. As a result, the default threshold is $0.327\,\text{kW}$. To

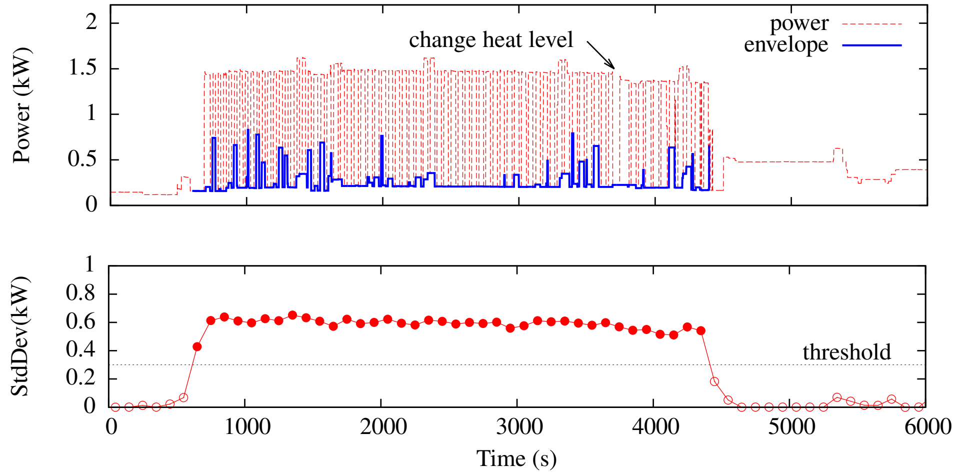


Fig. 7. Detecting stove burner. (1) Red curve: Total household power readings when a burner is working; Blue curve: The reconstructed lower envelope. (2) Standard deviation of power readings and threshold-based detection results (detection window size: 100 s).

suppress the false alarms caused by other high-power non-duty-cycled appliances, we further require that the zero crossing count of the mean-removed power readings in a window is at least 2. The bottom part of Fig. 7 shows the standard deviation of the power readings in the top part and the detection result. We can see that the time duration that the burner is working can be accurately detected. For the power readings in a window that has a positive detection, we apply the k-means algorithm with $k = 2$ and then interpolate the power readings in the cluster with a smaller average to reconstruct the lower envelope of power consumption (i.e., the background power), as shown in the top part of Fig. 7. With the lower envelope, it is easy to calculate the energy consumption of the duty-cycled appliance. In typical U.S. homes, stove burner and oven are the major duty-cycled heating appliances and they are often the components of a range. Supero does not differentiate the duty-cycled heating appliances and attributes all energy consumption to the range. To address multiple simultaneously working duty-cycled appliances, the number of clusters, i.e., k, can be first determined by the technique presented in Section 5.2.

The rapid duty-cycling can cause significant errors to the EDF-based power event detection (cf. Section 4.3) and the second tier of the multi-modal data correlation (cf. Section 4.4). Hence, when a duty-cycled appliance is detected, Supero disables these two components and the power changes of the light/acoustic events in this period are set to be missing values. Although such a design can cause errors to other appliances, it is worthwhile to give priority to the high-power duty-cycled appliances since they usually dominate the total power consumption of a household.

8. IMPLEMENTATION AND DEPLOYMENT

8.1. Prototype System Implementation and Settings

8.1.1. Sensors and Smart Meter. The sensors are implemented using TelosB and Iris motes [Memsic Corp. 2011]. TelosB has a light sensor only while Iris has both light and acoustic sensors. According to our lab tests, the light sensors on TelosB and Iris have satisfactory isotropic sensitivity in a considerably large range of incoming angles, which can mitigate the impact of sensor orientation on the accuracy of the power-law-based association algorithm. The signal sampling and event detection algorithms described in Section 4 are implemented in TinyOS 2.1. The parameters used in these algorithms are carefully tuned offline and then fixed for different deployments. The sensors communicate directly with the base station. Such a single-hop topology suffices for our deployments in three apartments and two multi-story houses. TED5000 [Detective 2015] is used to measure the total household power consumption.

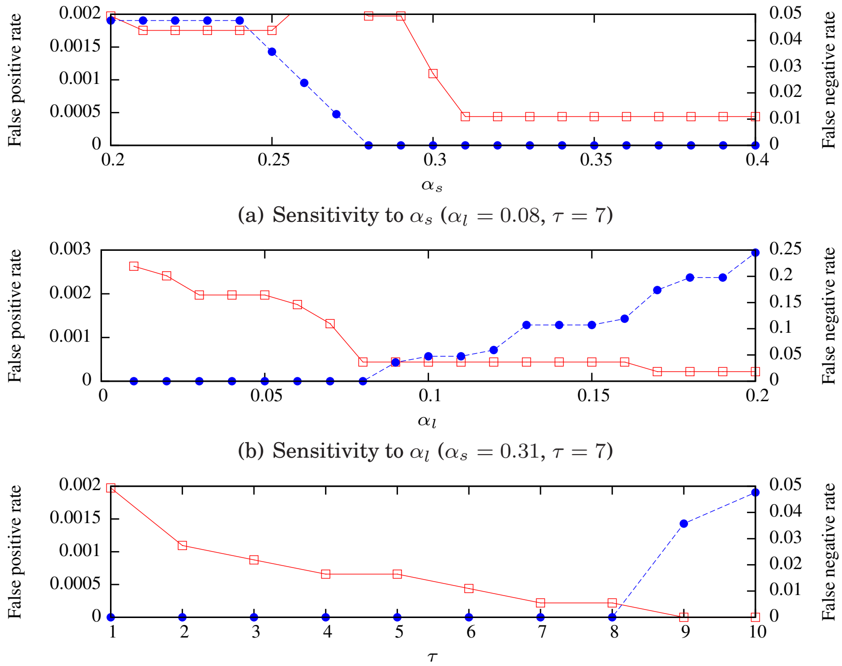
8.1.2. Base Station. The base station is a TelosB mote connected to a laptop computer. A daemon service on the computer retrieves real-time power readings from the TED5000 and stores the received event messages. The data correlation, clustering, and association algorithms are implemented in GNU Octave. The energy consumption of an appliance is computed by integrating estimated power over time. Note that this simple energy calculation method can be easily replaced by the regression-based method developed in [Kim et al. 2009] to improve robustness.

8.1.3. Groundtruth Kill-A-Watt Meters. In order to evaluate the accuracy of Supero, we built 14 power meters based on the P3 Kill-A-Watt (KAW) Model P4400 [P3 International Corp. 2012] to provide groundtruth power usage data of individual appliances. We connect two ADC channels of a Senshoc mote to two pins on the internal circuit board of a KAW to sample the voltage and current signals. Senshoc is a TelosB-compatible mote implementation with significantly reduced cost [Li et al. 2011]. The Senshoc mote computes and transmits the real-time power usage data to the base station for storage. Each modified KAW is carefully calibrated to output accurate power readings.

8.1.4. Parameter Settings. The parameters of algorithms in Supero are determined by offline experiments. Note that this process does not need to be repeated for different deployment environments. All the deployments in our experimental evaluation use the same parameter settings.

The first group of parameters are the coefficients of the EDF for power and light event detection presented in Section 4, i.e. α_s, α_l, and τ. Their settings are determined via a set of sensitivity analysis based on a data set collected in a controlled experiment that will be detailed in Section 9.2. We now use the sensitivity analysis conducted for power event detection to illustrate. Fig. 8 shows the false positive and negative rates in detecting power changes larger than $30\,\mathrm{W}$ versus the setting of a parameter when the other two parameters are fixed. From Fig. 8(a), the detection performance is insensitive to the setting of α_s when $\alpha_s \in [0.31, 0.4]$. From Fig. 8(b), a satisfactory trade-off between the two error rates can be achieved when $\alpha_l \in [0.08, 0.12]$. From Fig. 8(c), when $\tau \in [3, 8]$, the detector misses no power events and gives false positive rates lower than 0.1%. From the results in Fig. 8, we set $\alpha_s = 0.31$, $\alpha_l = 0.08$ and $\tau = 7$. We adopt a similar sensitivity analysis approach to determine the settings for light event detection as $\alpha_s = 0.18$, $\alpha_l = 0.074$, and $\tau = 4$.

The second group of parameters are d_0 and outlier cluster size in the light event clustering presented in Section 5.1. We set $d_0 = 2$, i.e., two feature vectors should be classified into two distinct clusters if the number of sensors that can only detect the first event is 2 more than that of the second event. Moreover, we set the outlier cluster size to be 2, i.e., we ignore the appliances that generate two or fewer events in a long period such as several days. As other parameters can be either easily set (e.g., η_A for acoustic sampling and δ in (1)) or autonomously adapted and/or optimized (e.g., η_L in Section 4.1, α and β in Section 6.1), we omit the details here.



(a) Sensitivity to α_s ($\alpha_l = 0.08$, $\tau = 7$)

(b) Sensitivity to α_l ($\alpha_s = 0.31$, $\tau = 7$)

(c) Sensitivity to τ ($\alpha_s = 0.31$, $\alpha_l = 0.08$)

Fig. 8. Sensitivity of power event detection performance to parameter settings.

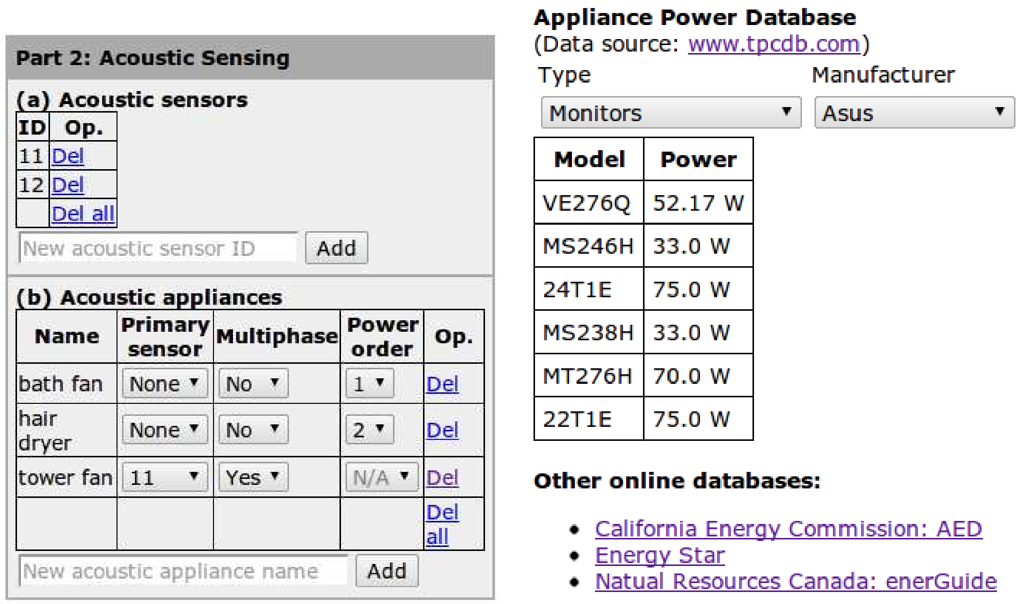
8.2. System Deployment and Configuration

This section discusses the sensor deployment and initial configuration of Supero.[1]

8.2.1. Sensor Deployment Strategies. A necessary condition for correct clustering and association is that every light/acoustic appliance can be detected, which is referred to as the *coverage requirement*. A conservative deployment strategy is to place a sensor close to each appliance. The number of sensors can be reduced by incrementally placing sensors close to appliances, starting with those that emit dim light/acoustic signals, until the coverage requirement is met. In our implementation, the coverage is checked by switching appliances on and check the sensors' LEDs that blink to indicate detection. Note that this coverage check is different from supervised training processes (e.g., [Patel et al. 2007]) that are typically conducted after system deployment and involve labelling the events with the source appliances. After coverage requirement is met, a few extra sensors may be deployed in regions without any sensors to provide redundancy and improve robustness. The effectiveness of the above conservative and incremental deployment strategies will be evaluated in Section 9.4.

8.2.2. User Inputs. First, Supero needs a list of the monitored appliances, which are categorized as lights, acoustic, or unattended appliances. Supero also needs to know whether an appliance has multiple working states although the exact number of the working states is optional. Second, for the light modality, Supero requires roughly estimated line-of-sight distances between the sensors and lights. Third, for the acoustic modality, Supero needs to know whether an acoustic appliance has a primary sen-

[1]An online video illustrating the system deployment and configuration can be found at https://youtu.be/4sSZaaV0Kv4



(a) Acoustic configuration (b) Rated power database

Fig. 9. Web configuration interface.

sor or not. All the non-primarily monitored acoustic appliances need to be sorted by their powers. Such a ranking is usually straightforward to obtain, e.g., based on common sense. Finally, Supero requires the rated powers of the unattended appliances, which can be obtained from the labels on the appliances or from a database of appliance rated powers. Supero only needs to be reconfigured occasionally, e.g., when sensors/appliances are relocated.

8.2.3. Configuration Interface. We have developed a web configuration interface using JavaServer Pages served by the base station computer to help the user input all the required information. For instance, Fig. 9(a) shows the configuration for the acoustic sensing, where the user can input the acoustic sensor IDs, appliance names, and other required information. In addition, we leverage TPCDB [TPCDB 2015], which is an online collaboratively edited database of appliance powers, to help the user input the required rated powers. Currently, TPCDB comprises the information of more than 500 appliances. Fig. 9(b) shows our interface of querying TPCDB through its web service API, where the user can find the rated power by appliance type, manufacturer and model. The case studies presented in Section 9.6 shows that this interface can be easily used by non-professionals.

9. EXPERIMENTAL EVALUATION

9.1. Deployments and Evaluation Methodology

We deployed and evaluated Supero in five real households. We first deployed Supero in a $40\,\mathrm{m}^2$ single-bedroom apartment (Apartment-1). As most of the appliances in Apartment-1 can be monitored by groundtruth KAW meters, this deployment allows us to extensively evaluate the accuracy of Supero. We then evaluate the sensor deployment strategies (cf. Section 8.2) in an $80\,\mathrm{m}^2$ apartment (Apartment-2). In addition, we deployed Supero in a one-story three-bedroom ranch house (House-1) to evaluate the

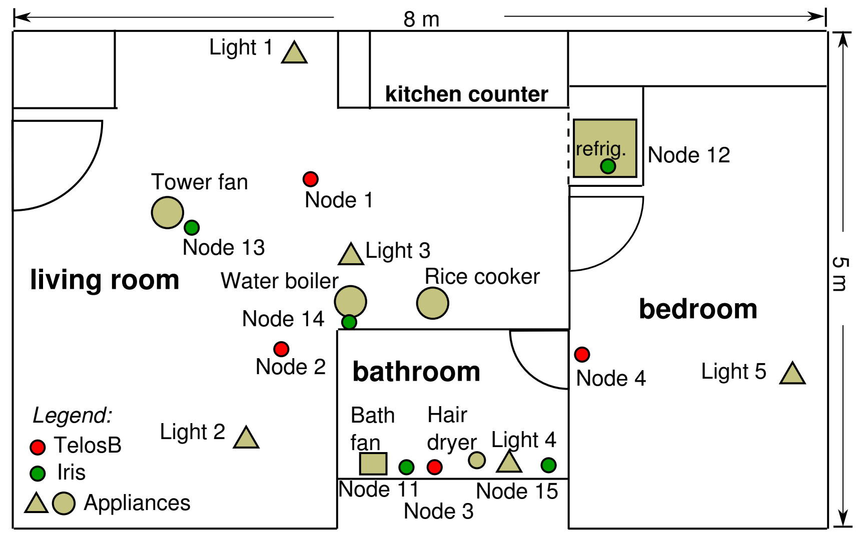


Fig. 10. Apartment-1 deployment.

portability of Supero to larger homes. Lastly, we recruited two homeowner volunteers to deploy Supero in their homes, an apartment (Apartment-3) and a two-story house (House-2). The Apartment-3 and House-2 deployments evaluate if non-professionals can deploy Supero easily.

We compare Supero with two baseline approaches. The first baseline approach (referred to as *Oracle*) uses appliances' groundtruth states and then applies the regression-based energy calculation method in ViridiScope [Kim et al. 2009]. In the second baseline approach (referred to as *Baseline*), the state of each appliance is detected by the sensor closest to the appliance and then the regression is applied. The results of *Baseline* will help us understand the challenges brought by an *ad hoc* sensor deployment.

9.2. Controlled Experiments in Apartment-1

9.2.1. Experimental Settings. The electrical appliances in Apartment-1 include 5 standing lights, a refrigerator, a water boiler, a 3-speed tower fan, a rice cooker, a bath fan, a hair dryer, 3 laptop computers, and a WiFi router. The apartment uses a natural gas range and a steam-based central heating unit that do not draw electrical power. The deployment consists of 4 TelosB and 5 Iris motes. The Iris motes only detect acoustic events. The laptops and router cannot be easily detected by sensors. However, as the router's rated power is known and it is always on, Supero can estimate its energy consumption. The residual energy consumption is thus mainly attributed to the laptops. The rice cooker, water boiler, and refrigerator are treated as unattended appliances, because they do not emit light or stable acoustic signals. The water boiler and refrigerator are also monitored by acoustic sensors. Fig. 10 shows the floor plan and sensor positions. The sensors are placed on the floor, a nearby table, chairs, and a toilet. The positions of the sensors are not carefully chosen except for the tower fan, refrigerator, and water boiler. Sensors are deployed close to these quiet appliances. As the bathroom has complex sound patterns, two acoustic sensors are deployed and both of them can hear all the appliances and the sound of water flowing in the bathroom.

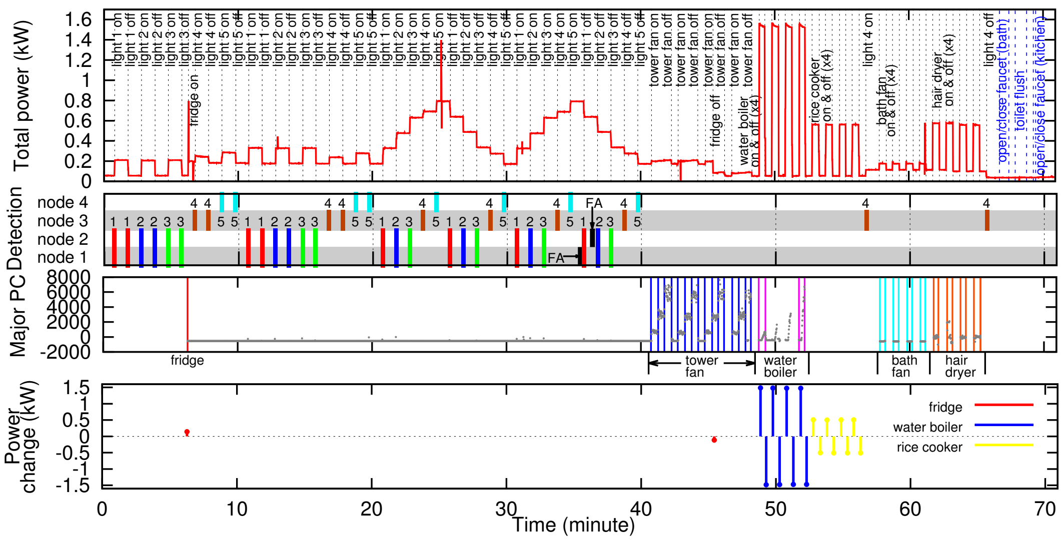


Fig. 11. Results of the controlled experiment in Apartment-1. (1) The top chart shows the power readings labeled with ground truths of the events. (2) The bars in the second chart show the detections of the light sensors. Two black bars at around the 35^{th} minute are false alarms (labeled “FA” in the chart) identified by the multi-modal data correlation. Clusters are differentiated by colors and the overhead numbers are the IDs of the associated light. (3) The third chart shows the major principle component given by PCA and the detected acoustic transitions. The acoustic transitions of the same color are associated with the same appliance. (4) The bottom chart shows the clustered and associated power events of the unattended appliances.

Table I. Energy breakdown for the 1-hour controlled experiment in Apartment-1.

Appliance		KAW		Supero			Oracle			Baseline		
Name	Rating (W)	Power (W)	Energy (kW·h)	Power (W)	Energy (kW·h)	Error* (%)	Power (W)	Energy (kW·h)	Error* (%)	Power (W)	Energy (kW·h)	Error* (%)
Light 1	150	152	0.0307	154	0.0309	0.7	152	0.0305	0.7	153	0.0310	1.0
Light 2	150	148	0.0298	150	0.0300	0.7	150	0.0300	0.7	151	0.0305	2.3
Light 3	150	151	0.0300	153	0.0304	1.3	153	0.0306	2.0	152	0.0307	2.3
Light 4	50	60	0.0211	61	0.0210	0.5	60	0.0210	0.5	62	0.0219	3.8
Light 5	100	102	0.0207	103	0.0205	0.5	100	0.0200	3.4	102	0.0206	0.5
Water boiler	1500	1472-1524	0.0490	1479	0.0456	6.9	1481	0.0481	1.8	232	0.0289	41.0
Tower fan	N/A	23-40	0.0031	N/A	0.0029	5.3	{23, 28, 35}	0.0028	9.7	30	0.0045	45.1
Rice cooker	500	498	0.0163	508	0.0168	3.1	507	0.0168	3.1	508	0.0163	0.0
Hair dryer	N/A	442	0.0158	462	0.0150	5.1	459	0.0150	5.1	5	0.0018	88.6
Refrigerator	N/A†	117-146	0.0784	129	0.0841	7.3	122	0.0795	1.4	119	0.0848	8.2
Bath fan	N/A‡	N/A	N/A	60	0.0020	N/A	61	0.0020	N/A	55	0.0048	N/A
Router	12	12.5	0.0147	12	0.0142	3.4	13	0.0154	4.8	13	0.0154	4.8
3 Laptops	N/A	37-63	0.0468	36	0.0430	8.1	31	0.4840	3.4	53	0.0472	0.9
Average error						3.6			3.1			16.5

*Error is the relative error of energy, in percentage, with respect to the KAW measurements.
‡Bath fan is hardwired to the power line and hence no KAW is applied for it.
†Refrigerator's rated power is not available. However, its power events can be correctly associated when a rated power of 80 W to 400 W is given to Supero.

9.2.2. Energy Estimation Accuracy. This section presents the results of a *controlled* experiment, in which we intentionally turned on and off the appliances. It allows us to understand the micro-scale performance of Supero. Fig. 11 shows the groundtruth information, power readings, event detection and clustering results. Both of the two light false alarms are identified by the multi-modal event correlation. No light event is missed. All the light events are correctly clustered and associated. For the acoustic modality, the non-power sounds such as a toilet flush and running tap water can be identified by the multi-modal data correlation. From the third chart in Fig. 11, Supero fails to detect the off event of the refrigerator and four events of the water boiler. The miss detections of the water boiler are caused by the delay of sound. However, as discussed in Section 6.3, by jointly treating the refrigerator and water boiler as acoustic and unattended appliances, these misses can be successfully recovered by the events detected from the power readings. Other detected acoustic transitions including the phase changes of the 3-speed tower fan can be correctly associated.

Table I shows the groundtruth measurements by KAWs and the estimation results of the various approaches. Both Supero and *Oracle* can accurately estimate the power and energy of each appliance. The average errors of energy consumption estimate are lower than 4%. For a few appliances, Supero outperforms *Oracle*. This can be caused by small errors in the groundtruth measurements by KAWs and the adoption of different energy calculation methods in Supero and *Oracle*. As Lights 1, 2, and 3 have no nearby sensors, *Baseline* uses the groundtruth states of Lights 1, 2 and 3. For other appliances, *Baseline* uses the closest sensor to detect the state of an appliance. As *Baseline* does not perform data correlation and event clustering, it generates excessive false alarms. For instance, as the hair dryer is very noisy, all the acoustic sensors raise detections when the hair dryer is on, which causes false alarms for all the other acoustic appliances. Hence, *Baseline* yields wrong power and energy estimates for several appliances. In fact, it is quite difficult to deploy dedicated acoustic sensors as they can be easily triggered by any noisy appliances. Acoustic data from multiple sensors must be jointly processed to produce correct detection results.

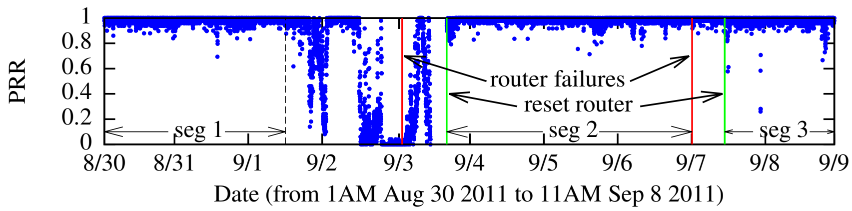
9.2.3. Impact of Distance Errors. This section evaluates the robustness of the association algorithm in Section 6.1 with respect to the errors in the light-sensor distances. The distances given to Supero are distorted as follows. First, we proportionally increase all the distances. As the association algorithm can find a best fit scaling factor β, the association remains correct even if we multiply the distances by 10. Second, we add a random bias to a particular distance in each test. The result shows that if the bias is within 70% of the true distance, the association remains correct. Finally, when we exclude Node 2 from the evaluation, the results remain the same as long as the order of the distances from Node 1 to Light 1 and Light 3 is consistent with reality, i.e., Light 1 is farther from Node 1 than Light 3. These results demonstrate that Supero is robust to the errors in the light-sensor distances.

9.3. 10-Day Experiment in Apartment-1

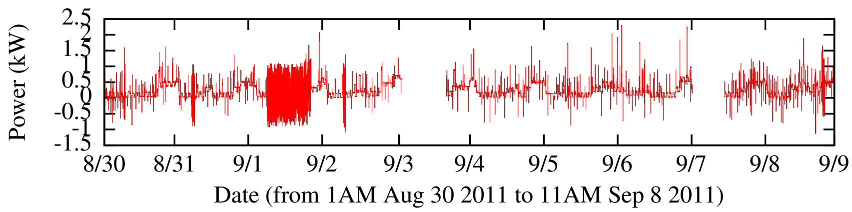
We conducted a 10-day *uncontrolled* experiment, during which two residents led normal lives in their apartment. In this section, we first discuss our experiences and learned lessons, and then present the evaluation results.

9.3.1. Experiences and Learned Lessons. We experienced the following three issues during the 10-day experiment.

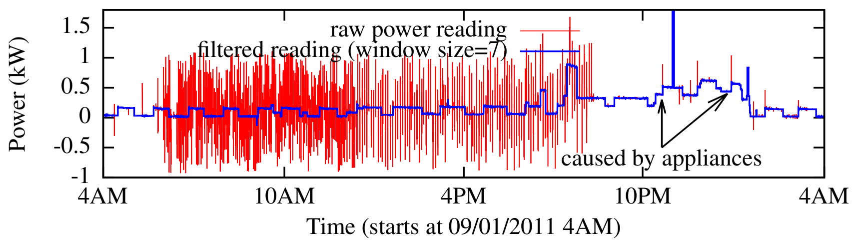
Power spikes. Power spikes are typical dynamics in power lines, which can be caused by bad weather conditions and turning on/off appliances in the tested home and even neighbor homes. Power spikes may cause errors in the appliance power estimation. In



(a) PRR of a KAW



(b) Power trace



(c) 12 hours of power trace on September 1, 2011

Fig. 12. PRR and power traces in 10 days.

the controlled experiment, we can see a few power spikes in the top chart of Fig. 11 when an appliance changes state. As we apply a guard region for computing the power change as discussed in Section 4.4, the power spikes do not affect the results. However, in the 10-day experiment, we observe excessive power spikes as shown in Fig. 12(b) that can affect the calculation of power changes for the detected events. We suspect that the power spikes observed on September 1 were caused by the thunderstorms during the period of the experiment. An expanded view of the power trace on that day is shown in Fig. 12(c). Almost all power spikes can be removed by a median filter with a window size of 7 seconds. We also apply the median filter with the same setting to the power traces collected in other experiments.

Router failures. The probe of TED5000 installed on the power panel sends real-time readings through power lines to the TED5000 gateway, which was attached on a power outlet and wired to the WiFi router to deliver readings to the base station computer. However, the router failed twice during the 10 days, leading to disruptions to the collection of power readings. We had to reset the router manually to restart the data collection. We suspect that the failures were caused by bugs in the router. As power readings are critical information to Supero, it is crucial to adopt a high-quality and stable router. Moreover, when the base station fails to receive power readings for a period of time, it can raise an alarm sound to remind the user to reset the router.

Communication performance. The quality of wireless links between the base station and sensors can affect the performance of Supero. Each Supero sensor only sends a packet when an event is detected while each KAW meter continuously transmits

Table II. Energy breakdown during 7 days in Apartment-1*

Appliance Name	KAW E (kW·h)	Supero P (W)	Supero E (kW·h)	Supero Error (%)	Oracle P (W)	Oracle E (kW·h)	Oracle Error (%)	Baseline P (W)	Baseline E (kW·h)	Baseline Error (%)
Light 1	4.14	154	4.17	0.5	152	4.11	0.9	152	4.11	0.9
Light 2	4.96	150	4.96	0.1	149	4.92	0.8	149	4.92	0.8
Light 3	6.15	155	6.24	1.4	155	6.25	1.7	155	6.25	1.7
Light 4	1.45	62	1.45	0.1	62	1.45	0.1	63	1.48	1.7
Light 5	0.39	105	0.39	0.2	105	0.39	0.7	110	0.41	5.5
Water boiler	0.48	1493	0.48	0.5	1491	0.48	1.6	0	0	100
Tower fan	0.15	30	0.21	50	26	0.17	17.9	24	0.24	66.2
Rice cooker	1.00	499	0.98	2.2	513	1.01	1.2	511	1.01	0.8
Hair dryer	0.09	467	0.07	19.2	467	0.09	0.4	3	0.02	73.2
Refrigerator	12.22	143	11.8	3.7	127	11.8	3.2	127	11.8	3.2
Bath fan	N/A	50	0.12	N/A	57	0.17	N/A	0	0	N/A
Router	2.12	12	2.03	4.3	18	3.04	43.3	18	3.04	43.3
Average error				7.5			6.5			27.0

*Error is relative error of energy with respect to KAW measurements.

groundtruth power usage to the base station by the attached Senshoc mote equipped with a CC2420 radio. Therefore, we use the data traces of KAWs to examine the packet reception ratio (PRR). Fig. 12(a) shows the PRR of a KAW during the 10 days. We can see that the communication performance significantly degraded and fluctuated between the evening of September 1 and the noon of September 3. As the residents watched online videos over WiFi during this period, we suspect that the poor link quality was caused by the interference from WiFi. We also examined the traces of other KAWs. Similarly, their link quality degraded during this outage period. We were able to repeat this phenomenon during an extra experiment using Senshoc motes and two laptops that transferred a large file over WiFi. Although the channel of Senshoc was set to 11, which is well separated from channel 6 used by WiFi, the PRR of Senshoc still significantly degraded. However, we did not observe significant degradation of PRR when experimenting with TelosB and Iris motes. Hence, we suspect that the performance degradation is caused by the imperfect antenna design of Senshoc.

Nevertheless, after the 10-day experiment, we have enabled packet acknowledgment and added retransmission mechanism to enhance the reliability of communication. Due to the router failures and lost groundtruth information from KAWs, we only use three data segments (“seg 1”, “seg 2” and “seg 3” shown in Fig. 12(a)). The total length of the three segments is more than 7 days. The three data segments are concatenated and then fed to the clustering and association algorithms.

9.3.2. Evaluation Results. Table II shows the results based on the seven days of data. During this period, 713 false alarms out of a total of 859 light events were raised by the light sensors, in which 703 of the false alarms are identified by the multi-modal data correlation. All the remaining false alarms are identified as outliers by the event clustering algorithm (cf. Section 5.1). In addition to the acoustic transitions generated by the refrigerator, 60 acoustic transitions were detected. We see that Supero can accurately estimate the energy consumption of lights. The tower fan was turned on and off twice and all its transitions were detected. However, two bath fan transitions were incorrectly associated with the tower fan, because Node 13 (i.e., the primary sensor for the tower fan) heard loud noises in the living room at the same time. The two false associations introduced errors in the energy estimates of the tower fan and hair dryer. As shown in Table II, the average error of Supero is only 7.5%. The average error of *Oracle* is 6.5%. Therefore, the performance of Supero is close to that of *Oracle*. *Baseline* still fails to estimate the energy consumption of several appliances due to excessive false alarms, leading to an average error of 27%.

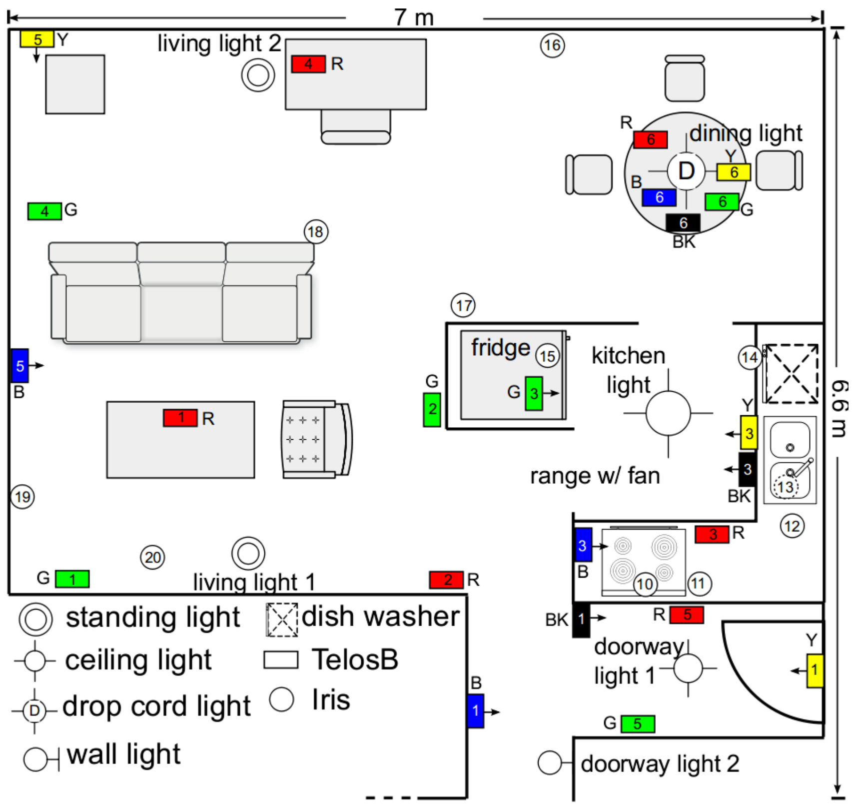
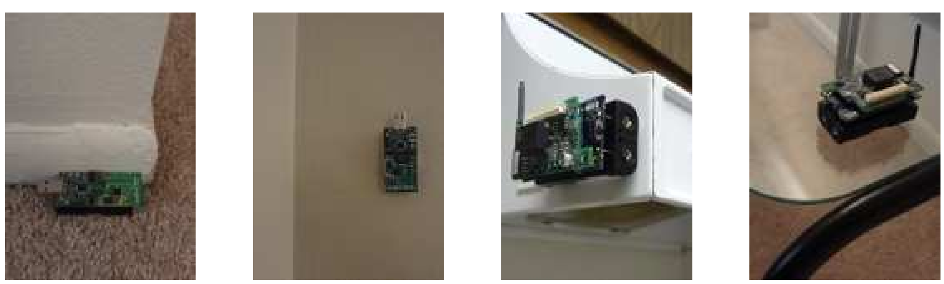


Fig. 13. Sensor placements in Apartment-2. The numbers in the squares and circles are the sensor IDs of TelosB and Iris, respectively. If a TelosB does not face upward, the arrow represents its facing direction.



(a) Node 2 (2nd placement) (b) Node 5 (4th placement) (c) Node 11 (d) Node 20

Fig. 14. Sensor installation examples. Sensors were placed on the ground, in the corner of walls, on the fan of a range, and on a table.

9.4. Experiments in Apartment-2

This section evaluates the performance of Supero under different sensor placements. We deployed 6 TelosB and 11 Iris motes in the doorway, living room, and kitchen of Apartment-2, as shown in Fig. 13. As the two doorway lights are controlled by the same switch, they are regarded as one light. As shown in Fig. 14, sensors were placed or attached on the ground, walls, appliances, and furniture. Note that the positions of

Table III. The set of sensors detecting a light (i.e., R_m) and clustering/association results

Light	Red	Green	Blue	Yellow	Black
Dining	{6}	{6}	{6}	{6}	{6}
Kitchen	{3}	{3}	{3}	{3}	{3}
Doorway	{5}	{5}	{1}	{1}	{1}
Living 1	{1,2,4}	{1,2,4}	{5}	{5,3}	{3}
Living 2	{1,2,4,6}	{1,2,4,6}	{5,6}	{5,6}	{6}
Result	✓	✓	✓	✓	X

sensors were chosen by common sense without careful planning. We also varied the positions of sensors in several trials and similar results were observed, as shown later in this section. We first evaluated the light modality. We conducted five sensor placement trials to monitor 6 lights including incandescent bulbs and fluorescent lamps. Different colors of the TelosB motes in Fig. 13 represent different placements, which are also labeled with the initials of color names, i.e., 'R', 'G', 'B', 'Y' and 'BK'. In the red and green placements, a sensor was placed close to each appliance. The blue and yellow placements follow the incremental strategy to reduce the number of sensors from 6 to 4. In the black placement, no sensor was deployed in the living area. All the placements ensure the coverage requirement. We conducted a controlled experiment to evaluate each placement. Table III shows the set of sensors that can detect the same light (i.e., R_m defined in Section 6.1). The clustering and association results of the red to yellow placements are correct. In the black placement, although all the events can be detected, they cannot be correctly clustered. For instance, although Node 6 can detect the near dining light ($13\,\mathrm{W}$) and the farther "living light 2" ($150\,\mathrm{W}$), the changes in light intensity from them are similar, leading to incorrect clustering.

To further demonstrate the flexibility of sensor deployment, we deployed 11 Iris motes and select four different subsets of them as sensor placements, which are $S_1 = \{\text{All Iris motes}\}$, $S_2 = \{10, 12, 14, 15, 16, 18, 20\}$, $S_3 = \{10, 12, 14, 19\}$, and $S_4 = \{10, 14\}$. All the subsets satisfy the coverage requirement. However, they represent very different deployment strategies. S_1 and S_2 use redundant sensors and hence are conservative. S_3 follows the incremental deployment strategy. As there is no sensor in the living area, S_4 does not follow any proposed deployment strategy. The acoustic appliances covered in the experiment include an exhaust fan over the range, a waste disposer in the sink, a dish washer, and a vacuum cleaner. During the experiment, we used the vacuum cleaner in both the dining and living areas. The exhaust fan has two speeds and Node 10 is designated as the primary sensor for the fan. For the other appliances, the order (rather than the actual values) of their power consumption is provided to Supero. The event detection and association results for S_1, S_2, and S_3 are correct. For S_4, although all the acoustic events can be successfully detected, some of them cannot be correctly associated. For instance, when the vacuum cleaner ran in the living area, Node 10 received the highest signal energy, which is inconsistent with its designation as the primary sensor for the exhaust fan.

The results in this section show that both the conservative and incremental deployment strategies can effectively ensure the sensing results. Moreover, the data correlation and the unsupervised clustering/association algorithms adopted by Supero allow the sensors to be deployed in an *ad hoc* manner with considerable flexibility.

9.5. Experiments in House-1

House-1 is a one-story three-bedroom ranch house with a living space of about $150\,\mathrm{m}^2$. Compared with Apartment-1, it has more lights of various types (incandescent bulbs and standard/compact fluorescent lamps). The deployment consists of 7 TelosB and 3 Iris motes. The Iris motes detect both light and acoustic events. We conducted a

Table IV. Energy breakdown in House-1*

Appliance Name	**Groundtruth** P (W)	E (kW·h)	**Supero** P (W)	E (kW·h)	Error (%)
Entry light	32	.0079	33	.0081	2.3
Hall light	38	.0112	38	.0109	1.9
Kitchen light	24	.0059	23	.0056	5.8
Dining light	76	.0149	77	.0113	24.6
Living light	43	.0041	41	.0040	3.1
Master bed light	33	.0065	31	.0061	6.0
Master bath light	22	.0054	21	.0052	3.6
Master bath fan	47	.0069	47	.0068	2.3
Guest bed light	29	.0071	29	.0056	21.2
Guest bath light	20	.0070	20	.0070	0.6
Guest bath fan	41	.0097	40	.0097	0.0
Stove burner	1356	.4603	1379	.4675	1.6
Water dispenser	N/A	N/A	140	.0518	N/A
Average error					6.1

*Error is relative error of energy with respect to KAW measurements.

controlled experiment for more than 5 hours. Groundtruth information was manually recorded and then rectified by checking the total power readings. In the experiment, each light sensor could detect multiple lights, and 40 false alarms out of totally 127 light events were raised by the light sensors, where 38 of the false alarms were identified by multi-modal data correlation. The remaining two false alarms were identified as outliers by the clustering algorithm. Table IV shows the results. For one of the dining light events, a sensor monitoring the light missed the event, which resulted in a misclassification and error in estimating the energy of the dining light. From the background cluster of unattended power events, we observed that an unknown appliance with a power of 140 W was turned on for one minute about every 10 minutes. The appliance turns out to be a hot water dispenser at a sink. Moreover, the dispenser caused a missed detection of a guest bed light event, as the dispenser and the light were once turned on/off at the same time. The average error of Supero is 6.1%.

9.6. System Usability

We now present two case studies on how easily Supero can be deployed and configured by non-professionals. We recruited two homeowner volunteers to deploy Supero in their homes including a single-bedroom apartment (Apartment-3) and a two-story house with basement (House-2). We first introduced Supero and explained the deployment strategies to the volunteers, which took less than one hour. They then installed the sensors and configured the system using our web interface without any further instructions from us. For safety reasons, they did not install the TED5000.[2] In Apartment-3, the volunteer deployed 5 TelosB and 3 Iris motes to monitor all the appliances including 5 lights, a refrigerator, a microwave, and a fan. The deployment and configuration took only about half an hour. In House-2, the volunteer took about one hour to survey the appliances and another hour to install the sensors. He finally deployed 12 TelosB and 10 Iris motes to monitor 12 lights, an exhaust fan in the kitchen, a waste disposer, a dish washer, a refrigerator, a microwave, and three fans in three bathrooms respectively. The base station on the first floor could reliably receive data packets from sensors distributed on the two floors and basement. After the system deployments, we conducted controlled experiments to evaluate the deployments and

[2]The TED5000 probe needs to be hardwired to electrical service wires to get powered and connected to the gateway. Contactless power sensors [Patel et al. 2010], which are more friendly to non-technical end users, can be used instead.

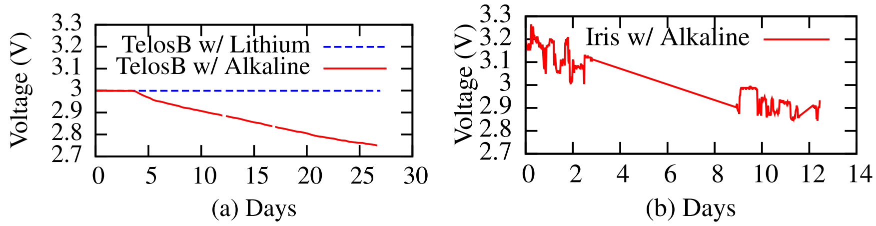


Fig. 15. Battery voltage traces of TelosB and Iris.

configurations. We generated total power readings according to gathered groundtruth to run the algorithms. The event detection, clustering, and association results of the controlled experiments are correct in both deployments. These two case studies show that the non-professional users were able to quickly deploy Supero and ensure correct sensing results. We also find that both users preferred the conservative deployment strategy discussed in Section 8.2.

9.7. System Lifetime

This section evaluates the lifetime of the battery-powered Supero sensors. In this experiment, we force the CPUs of the motes to stay active even though they would operate in low duty cycles (e.g., $\leq 5\%$ for Iris) in Supero. The radios are turned on only when there are packets to transmit. The TelosB motes report their battery voltages to the base station every minute. Fig. 15(a) plots the battery voltages of two TelosB motes with Alkaline and Lithium batteries, respectively, over time. The projected lifetime with Alkaline batteries is 79 days by conservatively setting the minimum operating voltage (MOV) to be $2.2\,\text{V}$ although it is $2.1\,\text{V}$ in datasheet [Memsic Corp. 2011]. With the high-capacity Lithium batteries, there is no observable voltage drop in one month. For the tested Iris mote, we enforce it to always work in the fast sampling mode. It piggybacks voltage reading to the acoustic feature packet. Fig. 15(b) plots the battery voltage of the Iris with Alkaline batteries. The tested Iris kept working from the 4^{th} to the 9^{th} day. Regression analysis shows that the projected lifetime is 40 days by conservatively setting the MOV of Iris to be $2.2\,\text{V}$, since the MOVs of the RF230 radio chip and ATmega1281 8MHz MCU on Iris are $2.1\,\text{V}$ and $1.8\,\text{V}$. We note that the lifetime can be further extended by simply using Lithium batteries and duty-cycling the CPU of motes.

10. CONCLUSION AND FUTURE WORK

This paper presents Supero – a sensor system for unsupervised residential power usage monitoring. In Supero, the multi-sensor fusion can effectively reduce sensing errors in complex household environments. By using unsupervised event clustering algorithms and a novel appliance association framework, Supero can autonomously estimate the power and energy usage of each monitored appliance. Extensive evaluation in five real homes shows that Supero can be deployed with considerable flexibility and provide accurate monitoring results.

Complementary to Supero, a few direct meters (e.g., the Zigbee-enabled KAW) can be applied to handle certain other appliances that have highly complex light/acoustic signal characteristics (e.g., TV) and power consumption profiles (e.g., furnace). In our future work, we will explore the use of other sensing modalities (e.g., infrared, seismic, and magnetic) to monitor these complex appliances. We will explore privacy-preserving strategies to prevent information leakage due to the wireless communications in Supero. Moreover, we will study the applications of Supero in non-residential environments such as legacy public infrastructures without permanent appliance-level monitoring capability.

REFERENCES

Alertme. 2015. AlertMe. (Aug. 2015). Retrieved August 5, 2015 from http://www.alertme.com

Rainer Burkard, Mauro Dell'Amico, and Silvano Martello. 2012. *Assignment problems*. SIAM, Philadelphia.

The Energy Detective. 2015. The Energy Detective. (Aug. 2015). Retrieved August 5, 2015 from http://www.theenergydetective.com

Steven Drenker and Ab Kader. 1999. Nonintrusive monitoring of electric loads. *IEEE Computer Applications in Power* 12, 4 (1999), 47–51.

Richard O. Duda, Peter E. Hart, and David G. Stork. 2012. *Pattern Classification*. John Wiley & Sons, New York City.

Linda Farinaccio and Radu Zmeureanu. 1999. Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses. *Energy and Buildings* 30, 3 (1999), 245–259.

Sidhant Gupta, Matthew S. Reynolds, and Shwetak N. Patel. 2010. ElectriSense: single-point sensing using EMI for electrical event detection and classification in the home. In *12th ACM International Conference on Ubiquitour Computing (UbiComp)*. 139–148.

George W. Hart. 1992. Nonintrusive Appliance Load Monitoring. *Proceedings of IEEE* 80, 12 (1992), 1870–1891.

Bo-Jhang Ho, Hsin-Liu Cindy Kao, Nan-Chen. Chen, Chuang-Wen You, Hao-Hua Chu, and Ming-Syan Chen. 2011. HeatProbe: a thermal-based power meter for accounting disaggregated electricity usage. In *13th International Conference on Ubiquitous Computing (UbiComp)*. 55–64.

Insteon. 2015. Insteon. (Aug. 2015). Retrieved August 5, 2015 from http://www.insteon.net

Xiaofan Jiang, Stephen Dawson-Haggerty, Prabal Dutta, and David Culler. 2009a. Design and implementation of a high-fidelity ac metering network. In *The 8th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. 253–264.

Xiaofan Jiang, Minh Van Ly, Jay Taneja, Prabal Dutta, and David Culler. 2009b. Experiences with a high-fidelity wireless building energy auditing network. In *The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 113–126.

Deokwoo Jung and Andreas Savvides. 2010. Estimating building consumption breakdowns using ON/OFF state sensing and incremental sub-meter deployment. In *The 8th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 225–238.

Younghun Kim, Thomas Schmid, Zainul M. Charbiwala, and Mani B. Srivastava. 2009. ViridiScope: design and implementation of a fine grained power monitoring system for homes. In *The 11th International Conference on Ubiquitour Computing (UbiComp)*. 245–254.

Liqun Li, Guoliang Xing, Limin Sun, Wei Huangfu, Ruogu Zhou, and Hongsong Zhu. 2011. Exploiting FM radio data system for adaptive clock calibration in sensor networks. In *The 9th International Conference on Mobile Systems, Applications, and Services (MobiSys)*. 169–182.

Andrea H. McMakin, Elizabeth L. Malone, and Regina E. Lundgren. 2002. Motivating Residents to Conserve Energy without Financial Incentives. *Environmental and Behavior Journal* 34, 6 (2002), 848–863.

Memsic Corp. 2011. TelosB, Iris Datasheets. (2011).

P3 International Corp. 2012. P4400 Kill A Watt ^{TM} Operation Manual. (2012).

Shwetak N. Patel, Sidhant Gupta, and Matthew S. Reynolds. 2010. The Design and Evaluation of an End-User-Deployable, Whole House, Contactless Power Consumption Sensor. In *ACM Conference on Human Factors in Computing Systems (CHI)*. 2471–2480.

Shwetak N. Patel, Thomas Robertson, Julie A. Kientz, Matthew S. Reynolds, and Gregory D. Abowd. 2007. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line. In *The 10th International Conference on Ubiquitous Computing (UbiComp)*. 271–288.

Alexander Strehl and Joydeep Ghosh. 2003. Relationship-based clustering and visualization for high-dimensional data mining. *INFORMS Journal on Computing* 15, 2 (2003), 208–230.

Z. Cihan Taysi, M. Amac Guvensan, and Tommaso Melodia. 2010. TinyEARS: spying on house appliances with audio sensor nodes. In *The 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*. 31–36.

TPCDB. 2015. The Power Consumption Database. (Aug. 2015). Retrieved August 5, 2015 from http://www.tpcdb.com

U.S. DoE. 2006. Annual Energy Outlook. (2006).