ERNEST ORLANDO LAWRENCE
BERKELEY NATIONAL LABORATORY

Automated Measurement and Verification and Innovative Occupancy Detection Technologies

Phillip Price, Bruce Nordman, Mary Ann Piette, Rich Brown, Janie Page, Steven Lanzisera and Jessica Granderson

Environmental Technologies Area

September, 2015

Automated Measurement and Verification and Innovative Occupancy Detection Technologies

September 20, 2015

Phillip Price, Bruce Nordman, Mary Ann Piette, Rich Brown, Janie Page, Steven Lanzisera and Jessica Granderson

Building Technology and Urban Systems Department

Lawrence Berkeley National Laboratory

Prepared for:

Marina Sofos,

US Department of Energy Building Technologies Office

Acknowledgement

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Building Technologies Office, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Executive Summary

In support of DOE's sensors and controls research, the goal of this project is to move toward integrated building to grid systems by building on previous work to develop and demonstrate a set of load characterization measurement and evaluation tools that are envisioned to be part of a suite of applications for transactive efficient buildings, built upon data-driven load characterization and prediction models. This will include the ability to include occupancy data in the models, plus data collection and archival methods to include different types of occupancy data with existing networks and a taxonomy for naming these data within a Volttron[1] agent platform.

This research was conducted to:

- determine desired characteristics of, and technical feasibility of, new sensors that can inexpensively monitor the number of building occupants;
- explore how existing systems in buildings can be used to estimate the number of occupants as a function of time; and
- use energy savings Measurement and Verification (M&V) methods to quantify changes in building energy performance, both with and without the use of occupancy data.

Virtual Sensing

We have identified more than a dozen potential data sources for virtual occupancy sensing in buildings, and collected sample data on eight of them from LBNL buildings. Specifically, we acquired data from LBNL's telephone system, its Wi-Fi infrastructure, and several sources from the IP network infrastructure. Each source has its own advantages, disadvantages, and peculiarities. A general feature of most sources is that data could be extracted as frequently as desired, and it is almost as easy to analyze results for many buildings as it is to do so for a single one. Since all hardware required is already present in buildings, the implementation cost is close to zero. The technology appears to be highly replicable and scalable. In the primary study buildings, occupancy patterns are readily visible in the data, particularly arrival, departure, and lunchtime. Weekends and holidays are also similarly quite obvious in the data.

Measurement and Verification (M&V) Agent

Current Practice: No occupancy data

We used the M&V Agent that was developed for the Transactional Network project that contains a baseline model to a) predict load based on historic building load and weather data, and b) use the load predictions to quantify changes in energy use over time. Therefore, the Agent can be

[1] Volttron, developed by PNNL, is an agent based transaction platform to support communications on the smart grid. See http://transactionalnetwork.pnnl.gov/volttron.stm for more details.

used to detect abnormal operations, *and* to conduct measurement and verification of energy or demand savings. This Agent was applied to analyze the energy use patterns in energy use patterns in a building at LBNL (one of the eight above mentioned sites) and a building at PNNL. In the LBNL building the change from normal operation to heating season operation was easily detected, and a particular feature of the load shape was investigated and determined to be caused by inefficient controls settings. The PNNL building exhibited load behavior that may indicate the potential for substantial energy savings. Researchers at Google investigated the model used in the M&V agent, comparing its results to those generated by other models, with a specific interest in predicting the peak daily load. As of the end of January, 2015, they reported that the LBNL model was the best performer by a small margin. In a larger study, the LBNL model was one of 10 models (both proprietary, and published) that were tested to evaluate their accuracy in predicting energy, specifically for M&V applications (Granderson et. al 2015 a; Granderson et al. 2015 b). The LBNL model performed well, demonstrating solid accuracy relative to the current state of the art.

M&V with proxy occupancy data

Occupancy data are typically not available for most buildings. Most common occupancy sensors used for lighting controls are not communicating sensors. They are commonly used in conference rooms, office areas, and rest rooms, but the data are not archived. Such data were not available for any of our test buildings, but "virtual sensing" (see above) provides a solution: data on the number of Wi-Fi connections as a function of time can serve as a proxy for occupancy data. We extended the statistical M&V model described above to add the number of Wi-Fi connections as a predictive variable. For two of the LBNL buildings with the virtual sensing we used the model with input data from November and December 2014 and January 2015, excluding the holiday period from December 23-January 1. We used the model predict the electric load as a function of time during the holiday period that was excluded from the input data. The model substantially outperformed the previous model: in both buildings, the use of the occupancy proxy variable reduced. The root-mean-squared error(RMSE) changed from 3.2 to 1.7 for one building, and from 21.7 to 17.5 in the other. These are 46 and 20% improvements respectively.

We also compiled load data and Wi-Fi data from five UC Berkeley buildings, spanning four months in summer 2015. Four of the buildings showed little occupancy-related variation in load (and, indeed, little variation for any reason). In the fifth building, the occupancy proxy variable substantially improved the model predictions.

In total, occupancy data substantially improved baseline electric load prediction in three of the seven test buildings. The other buildings showed little variation in load from day to day and week to week, in spite of moderate changes in occupancy patterns.

Since the energy in most buildings is used to provide services to occupants, such as lighting and space conditioning, in an efficiently operated building the load would vary substantially with occupancy. The new capability to identify buildings whose load is insensitive to the number of occupants may allow those buildings to be targeted for energy savings measures such as occupancy-controlled lighting and ventilation.

Conclusions and Future Directions

While both of the occupancy-related aspects of this work (virtual occupancy sensing, and use of that virtual data to improve baseline model accuracy) are preliminary, they seem promising and merit further work in virtual sensing (also known as “proxy” or “inferential” sensing, and in the literature as “implicit sensing”) and monitoring to generate an equivalent value to traditional sensors and yield a value that is actionable for building operations.

We have identified promising future directions for each topic area. Among these will be to validate the IT system data with ground truth occupancy data, to compare the various methods with each other, and extract key metrics. Further work is also needed to use these and related occupancy data to better understand the linkages between occupancy and energy use for measurement and verification, fault diagnostics, energy forecasting, and improved building control.

Disclaimer

Acknowledgments

The authors are grateful for support from Joe Hagerman and Marina Sofos in the Building Technologies Office at the US Department of Energy. Many people contributed to this report, but special mention needs to be made of Michael Smitasin of LBNL's network group, who provided much data, quickly, and easily. Also critical to the project were Daniel Fuller and Greg Thomas.

Table of Contents

Background and Introduction

The Transactive Network (TN) project, funded by the Department of Energy's (DOE's) Building Technologies Office (BTO), is a multi-laboratory effort led by Pacific Northwest National Laboratory (PNNL), with Lawrence Berkeley National Laboratory (LBNL) and Oak Ridge National Laboratory (ORNL) also contributing. This report provides a summary of the LBNL work in FY14 and FY15 related to occupancy sensing and measurement and verification.

Measurement and verification (M&V) generally refers to the process of quantifying energy savings. One common way of conducting M&V is to compare actual measurements of energy use against a model of anticipated energy use based on what is known about the things that drive energy consumption. Historically, M&V models have considered building or equipment characteristics and weather patterns. This report examines a modification to the M&V model to account for temporal variation in energy use associated with changes in building occupancy.

In 2013 LBNL developed a series of new applications for the TN focused on characterizing the energy savings associated with short- or long-term operational changes in a building (Piette et al, 2014). A demand response (DR) event is an example of a short-term change whereas an energy efficiency (EE) measure is a long-term change. Demand response is a change from normal patterns of electric energy consumption by end-use customers in response to changes in electricity price or incentive payments designed to induce lower electricity use when wholesale market prices are high or when the supply system reliability is jeopardized. The energy and power savings associated with these actions can be quantified and measured against the electric load that might reasonably be anticipated in the absence of those changes. These changes can be translated into economic terms based on an electricity tariff associated with a particular site. Specifically, LBNL developed applications to:

- Convey demand response (DR) events using a DR event scheduler.
- Calculate a baseline electric load shape that is used to estimate the short-term peak demand reduction from DR events (kW) or long-term savings from energy efficiency measures (kWh).
- Conduct measurement and verification (M&V) of energy and demand savings. Baseline loads are compared to actual metered energy use to determine the savings during DR events managed by applications such as PNNL's automated DR agent, or from energy efficiency interventions such as changes in RTU operations based on information from PNNL's fault detection agent.

One of the emerging concepts in the transactive agent platform development is to explore how data from, and interoperable access to, end-use controllers can be leveraged to allow building energy use to be better managed.

One example demonstrated in 2014 was to explore new ways to measure and continuously diagnose the operation of occupancy- and scheduling-based lighting controls (Granderson et al. 2015 c). Two key goals are present in this concept. First, energy use can be reduced overall if the agent systems are able to evaluate and identify energy waste. Energy waste may be present if the HVAC or lighting systems are operating outside design parameters or if the systems are running when there are no occupants present. This is a common problem in buildings. Second, by building and demonstrating control systems that are able to maintain fault-free efficient operations, and also report savings achieved over time, industry can take the needed steps to scale adoption of efficient controls. The ability to account for changes in building occupancy—that is, the number of people in the building—should allow major improvements in identifying and quantifying energy waste and in recognizing and diagnosing faults. Most building energy consumption is, or should be, used to provide services to building occupants, so in a building that is operating correctly and efficiently there should be a strong relationship between occupancy and load. For example, high HVAC load may represent wasted energy in a building that is sparsely occupied, whereas high load is necessary in a heavily occupied building. High night-time lighting load may indicate a fault if the building is unoccupied or sparsely occupied, but indicates correct behavior in a building that is heavily occupied at night.

As a first step in recognizing wasteful consumption or faulty systems are faulty, LBNL developed approaches to sense or estimate occupancy in buildings and use the resulting data to better predict building energy consumption.

Occupancy sensing solutions that are low-cost, widely applicable, and highly granular are needed as highlighted in (Brambley et al., 2005). Only 10% of commercial buildings use an energy management and control system (EMCS) and these tend to be the larger buildings in the stock. As a result, using approaches that do not rely on an EMCS existing can be more broadly applicable in the market (Katipamula et al., 2012). The market barriers of installation and other costs as well as poor interoperability and proprietary systems are highlighted in the two previously cited reports. Virtual sensing is well situated to address these. Virtual sensing does not require new hardware and IT networks are based on standard technology so can bypass these barriers. Virtual sensing uses devices installed and maintained for other purposes so that configuration, commissioning, and maintenance are not issues that the sensing function needs to address. Despite all these advantages, virtual sensing is not available on the market as there is no standard protocol to communicate such data between the sources of the information and the devices that could receive it. Creating such a standard is a near-term priority for future work. This could be used by any source of virtual sensing data. Research efforts such as this project have explored virtual sensing data (Melfi et al., 2011; Nordman et al., 2014). Several manufacturers of Wi-Fi access points sell hardware and software to obtain high-resolution tracking of individuals by monitoring the Wi-Fi footprint of phones, with retail shopping the primary target market. However, these systems generally require extra hardware and are relatively expensive. The LBNL portion of the multi-lab TN project focused on new diagnostic intelligence, highly-granular, and interoperable occupancy sensing through IT network traffic.

During FY15 we compiled data from several buildings on the UC Berkeley campus, and analyzed them using an improved statistical model. The model improves on previous versions by (1) allowing a different relationship between load and outdoor air temperature during the “startup” period of each day than during other periods of the day, and (2) allowing occupancy proxy variables to be included as predictive variables. Another way of saying this is the root mean square error in the model is reduced. Details of the model and its implementation are provided in an appendix to this report.

This report documents LBNL work on M&V and related issues for the Transactive Network project in FY 2014 and FY 2015 by:

- Describing virtual sensing principles and providing a number of disparate examples; summarizing the state of software and statistical models that have already been developed for the project;
- Illustrating the application of the software for recognizing and quantifying changes in building energy behavior;
- Discussing ongoing research in improving the software by taking data on time-varying building occupancy into account; and
- Outlining a proposed work plan for future work.

The Transactive Network (TN) project has a variety of goals related to recognizing how buildings are operating and how those operations affect energy use:

- Predict baseline load for short time intervals. Necessary for quantifying DR effectiveness.
- Predict baseline load for long time intervals. Necessary for evaluating savings from EE measures, i.e. Measurement and Verification (M&V) of energy savings.
- Operate buildings more efficiently (lighting and HVAC). Necessary for saving energy and money.
- Improve Fault Detection and Diagnostics (FDD). Necessary to identify and fix operational problems.

A major focus of our work in these areas is to develop ways to collect and incorporate data concerning the number of occupants in a building at a given time. The focus of this report is on baseline models, however all of the applications listed above benefit from data about occupancy (Brambley et al, 2005 Katipamula et al., 2012).

- Occupants affect energy use, so occupancy data can be used to improve baseline predictions, and therefore reduce errors in quantification of EE and DR.

- Occupancy data make it possible to optimize the amount of lighting and HVAC provided, thereby increasing efficiency.

- Occupancy data can be used to distinguish energy use that is necessary to provide services from energy use that is not, thus detecting faulty equipment or operations.

Ideally, the absolute occupant count as a function of time could be measured for each area of a building, such as an HVAC zone. There are many techniques for absolute occupancy measurement with cameras and badge readers. Li et al (2012) describe occupancy measurement with RFID tags for example.

In light of the costs to install absolute occupant measurement and counting systems, we are pursuing “virtual sensing” to solve this problem, with a goal to making it broadly available to building owners (it is not available today as a low-cost capability). This involves compiling data that are already collected, or that can easily be collected, and that are related to occupancy, from which occupancy can be inferred. Such data are “proxies” for occupancy data. Examples include the number of phone calls made, number of Wi-Fi connections, and various measures of IT network usage (Melfi et al., 2011; this study only collected the virtual sensing data from several sources and compared it to manual occupancy counts and building total energy use). Ideally future work will compare absolute counting systems with virtual occupancy data. However this calibration and comparison is beyond the scope of this study. Our main objective in this research was to evaluate if the virtual sensing data improved the goodness of fit of baseline models. If they do, we assume the virtual data helps explain variations in energy use.

In this report we first review the importance of having information about the number of occupants in a building (or a portion of a building) for energy savings and comfort, and what approaches we are taking to obtain data about occupancy. We then explore virtual sensing as a promising technology for understanding building energy use, and for actively controlling buildings. We give brief examples of how our current TN agents perform in the absence of data on occupancy, and discuss our fledgling attempts to create statistical models that include occupancy data. Finally, we discuss our proposed future work.

Why Occupancy Matters

The number of occupants in a building can have a large influence on the amount of energy used in the building, particularly in buildings that are relatively more efficient and/or well controlled. In many cases, as occupancy increases, modest but observable increases in electricity use are seen.

In this work, the M&V approach follows the principles of the International Performance Measurement and Savings Protocol (IPMVP) (EVO 2014). It focuses on whole-building level M&V (IPMVP Option C), but could also be extended to measure isolation approaches based on submetered data (IPMVP Option B). In this approach, the energy saved by an ECM is defined as the amount of energy the building would have used in the absence of the ECM, minus the amount of energy the building actually used. The amount of energy the building actually used is measured by the meter, but the amount of energy the building would have used must be obtained from a prediction. Typically, the prediction is generated by using data from the period prior to the implementation of the ECMs to create a statistical model that can be used to predict the building's energy use in the absence of the ECMs; this model prediction is known as the "projected baseline energy use" or just the "projected baseline." Then the ECMs are implemented and the building's power consumption is measured and compared to the projected baseline. The difference between predicted baseline load and actual load is assumed to represent the savings from the ECMs, and is referred to as avoided energy use.

Changes in load due to occupancy changes can lead to an over- or under-estimate of the effect of the conservation measures if those occupancy changes are not taken into account. For example, suppose an ECM reduces the building load by 5% on average, over a 6-month post-ECM evaluation period, but that the building's occupancy increases by 10% during the same period. The ECM might appear to have no effect on the building's energy consumption, or even to increase it, since the occupancy-induced increase in energy consumption may overwhelm the effect of the ECM. The opposite effect can occur as well, with a load reduction being incorrectly attributed to an ECM when in fact it is due to decreased occupancy. In common practice, these effects would be quantified by an engineer, through the application of 'non-routine' adjustments to the baseline-predicted energy use. The ability to adjust for occupancy-induced changes in load patterns within the baseline models themselves would reduce the error in the predicted baseline and thereby substantially improve the measurement and verification of energy conservation, and the quantification of Demand Response effectiveness. Since this occupancy data is not routinely available in today's buildings, most models do not include occupancy as a predictor variable.

The ability to adjust for occupancy also has important implications for fault detection and diagnostics. Some fault detection and diagnosis (FDD) methods detect faults by looking for anomalous changes in load patterns, such as an increase in night-time load, which can indicate a failure in lighting or in HVAC system scheduling. An increase in building occupancy—at night, in this example—can also cause an increase in load. The ability to recognize, and adjust for, occupancy-induced changes in load patterns can enable improved FDD by allowing occupancy-induced load changes to be distinguished from genuine faults.

In some applications, low-time-resolution occupancy data are sufficient, especially for M&V. For example, if an office building loses a tenant so that one of the floors becomes unoccupied for several months, simply knowing that the average daytime occupancy of the building has decreased by 20% can allow the baseline prediction to be adjusted downward. Even when the size of this adjustment is only a rough estimate, it is much better than not taking the change into account at all.

Alternatively, some buildings have energy consumption that is nearly insensitive to the number of occupants: lights and HVAC systems are on pre-set schedules, and ventilation airflow is not based on carbon dioxide concentrations or other occupancy-related measures. The ability to identify these buildings may represent an opportunity to perform energy conservation measures such as installing motion sensors to control lights, and carbon dioxide concentrations to control ventilation.

In other applications, such as fault detection and diagnostics (FDD), and quantifying DR effectiveness, high time-resolution occupancy data (or data on proxy variables) are needed because the comparison between actual load shape and baseline load shape is made at a timescale of minutes or hours rather than weeks or months.

The discussion above is presented in terms of whole-building occupancy and whole-building electric load, but is equally valid for spatial subsections such as an entire floor, or a wing of a building, or smaller groupings such as an office suite or even an individual office. And of course, in some applications data must be high-resolution in both space and time. For example, some buildings have tenant submetering and large changes in occupancy can explain large changes sub-metering if a tenant vacates a whole floor of an office building. Also, if a building has controls to turn off lights in unoccupied areas, and to turn them on when people are present, then the correct operation of those systems can only be verified by knowing which areas were occupied at which times, and which lights were on at which times.

Virtual sensing

While conventional sensors use devices dedicated to that purpose, "virtual sensing" leverages hardware and communications infrastructure already installed in most buildings. In the literature it is referred to as "implicit sensing", the act of obtaining occupancy (or other) data from existing networks or devices, that were installed for some other primary purpose (Melfi et al., 2011; Nordman et al., 2014). The sensing is implicit in behavior of a device not intended to be a sensor. Virtual sensing is often low-to-no cost to begin using, and is already present in many locations around the world. To accomplish using virtual sensing to save energy or help understand energy use patterns, the sensing data must be collected, processed, communicated, and then interpreted for use.

Building occupancy is a critical parameter to understand for building energy efficiency in general, and the most promising initial application of virtual sensing[2]. It is also the factor most useful to know for improving measurement and verification of energy and demand savings. For all these reasons, occupancy was the focus of our work on virtual sensing.

Many types of data that can be used for virtual sensing are already collected, or could readily be collected, in most commercial buildings. Appendix A presents a catalog of such data types. Examples of virtual sensing that can give an indication of occupancy include the following, and with each the timescale at which the data are commonly recorded.

- Number of phone calls from wired phones (hourly to monthly)
- Number of Wi-Fi device connections (hourly or less)
- Various measures of computer or network usage (typically hourly or less)
- Carbon dioxide measurements in the ventilation system (hourly or less)
- Detections by motion sensors[3] (minutes to hours)
- Gallons of water used in a given time period (typically monthly)

For this project, we focused on data that are already being collected at buildings at LBNL, and at the University of California, Berkeley (UCB). Part of the research was to understand the level of effort required to compile and process the data so they can be useful. Some data can be acquired in real-time, or nearly so; these can be used in dynamic building operation, such as in feeding into a Transactional Network. We also sought to understand the relative reliability, granularity, and latency of each source. Some sources are easier to obtain, and some are available in more buildings. Some data are retrieved only periodically, sometimes requiring manual effort, so suitable only for retrospective analysis, or future planning.

Another part of the research was to understand the utility of different types of data. For example, there are several types of data that relate to IT network usage and that may be

[2] The focus in this paper is "Tier 1" virtual sensing in which no new hardware is needed to procure the data. There are examples of "Tier 3" virtual sensing in which hardware is added to a buildings. Using submetered electricity data for equipment health monitoring is an example of that.

[3] Motion sensor data is not virtual sensing data but we collected it to validate the data from other sources.

logged, such as total network traffic, the number of Wi-Fi connections, the number of web pages requested, and so on. We compiled data of many of these types in order to investigate which appear to be most useful for reliably reflecting actual occupancy.

A principal way to use virtual sensing data is to feed it directly into dynamic building operation. The largest opportunity here is climate control, so that buildings can be run on the basis of actual occupancy, rather than fixed schedules of expected occupancy. A building could only initiate workday temperature and ventilation when a nominal fraction of the building occupants have arrived (perhaps 5%), or start normal operation early, but cease it if expected occupancy does not occur. This can automatically detect holidays and daylight saving time changes, and also account for anomalous occupancy, either more or less than expected. For buildings that can vary the amount of ventilation, that amount can be based on the fraction of normal occupancy that occurs at each moment rather than a constant value as is more typical.

To be able to feed the results of virtual sensing into a transactional network, we have posted sample data from our analysis to a sMAP[4] server. Volttron agents could then pick up such data on a dynamic basis, or for off-line analysis as for M&V or understanding scheduling. We have tested the sMAP posting but not the use of the virtual sensing data by Volttron agents.

Moving forward, virtual sensing data – and in fact all sensing data – should be forwarded to an “Occupancy Server” for each building. This entity would gather occupancy and related data from other devices, process and aggregate the data, and then provide it to devices that can use the information. This could be for dynamic building operation and retrospective analysis such as M&V. Granularity needs over space and time vary so the occupancy server can provide data of the form needed by the requestor, combine the best insights of all sources, be resilient to sources becoming unreliable or disappearing entirely, and easily add new sources. Ideally an occupancy server is not a stand-alone device, but rather a function of a device that already exists for some other purpose. This can make the cost (and energy use) of an occupancy server low.

Figure 1 shows an example series of virtual sensing data. It covers one typical workday and shows the number of “Wi-Fi associations” seen by the network management system on 10 minute intervals. The building in question is building 90 at LBNL. All devices that are connected to the Wi-Fi network are included in the total, primarily phones, with notebook PCs making up most of the rest. The base load of about 20 devices is likely PCs that are left on 24/7 (potentially including some desktop PCs that use Wi-Fi for convenience). The peak value of just under 300 corresponds to the number of people we expect to be in the building at any time. While the number of building occupants in principle is considerably higher – over 350 – work

[4] sMAP (Simple Measuring and Actuation Profile) is a RESTful web service which allows instruments, sensors and other producers of physical information to directly publish their data. The data are stored in a time-series database and can be accessed through a simple API. See https://people.eecs.berkeley.edu/~stevedh/smap2/ for additional information

travel, vacation, sick days, and telecommuting all reduce this, even when visitors are added. While some people have two Wi-Fi devices, a notebook PC may not be awake and connected some of the time, and some people may have no Wi-Fi device at all. The pattern over the day also matches our expectations, with significant variation in arrival and departure times, and a noticeable lunch dip. The graph also shows how the HVAC system in the building is operated, with it being either all-on or all-off on a fixed schedule, supplemented by an optimal-start period before the 8a.m. start time to ensure that the building operates within the anticipated thermal range at 8am.

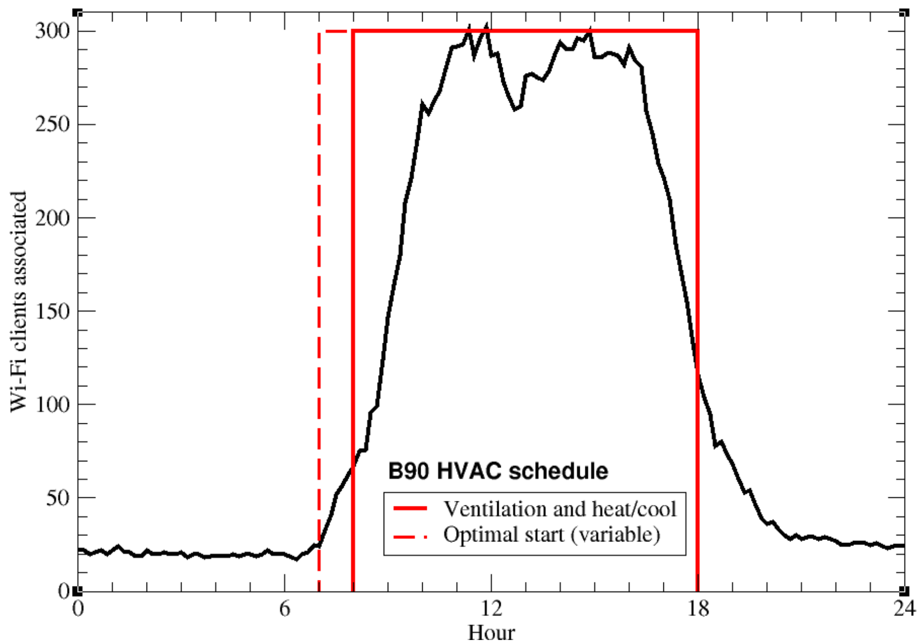


Figure1. Wi-Fi Associations over an example day for B90 (LBNL)

Virtual sensing extends our traditional sense of occupancy in several dimensions, as shown in Figure 2. A traditional occupancy sensor provides a single yes/no result, for a single location, and only a single point in time (lacking memory or network communications). Virtual sensing can extend this for people to give a count, identify individuals, and specify their activity; can provide time-series data for analysis; and can provide visibility across rooms or zones or a whole building rather than just a single location.

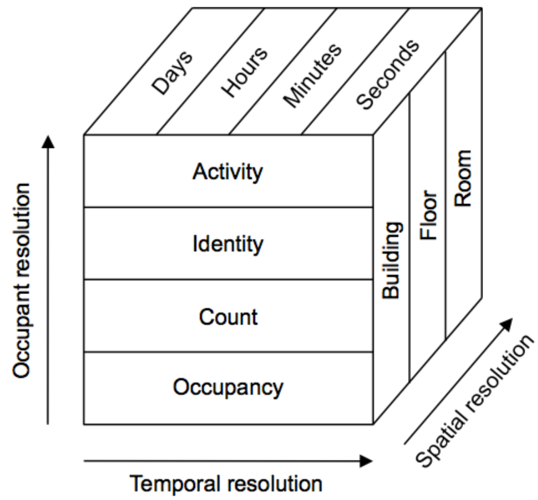


Figure 2. Virtual sensing characteristics (source: Melfi et al., 2011)

The virtual sensing mechanisms we explored fall into the following categories (**bold text** indicates items for which we have example data, *italicized text identifies* items we have explored in some detail):

<u>Internet Protocol Network Presence</u> — Address Resolution Protocol (**ARP)**, Dynamic Host Configuration Protocol (**DHCP)**, *Ping*, *Port Status*, **Wi-Fi networks**.

These methods rely on whether and how end-use devices are connected to IT networks, and retrieve data from network equipment (ARP and Port Status), network infrastructure servers (DHCP and Wi-Fi), or directly from end-use devices (Ping). The fact of each *device* being connected to the network - or rather how that changes over time - can be an indicator of *human* occupancy. Depending on how these data are acquired, some methods may have delays on when presence is recognized; due to how protocols operate, other methods will keep devices on the list for times--often several hours--after the device has left the network. These limitations are a problem for dynamic building operation, but not for retrospective analysis. These methods all, with additional information, provide granularity of occupancy data, often down to the individual device (and hence office). Some of them apply differently to wired and wireless network technologies.

<u>IT Network Traffic</u> — **DNS**, **Web browsing**, *direct traffic analysis*.

These methods use data on the network traffic itself to provide information indicating occupancy. This can be available from servers or from analysis of data traffic from individual devices. The latter is like the reverse of a network security firewall; a firewall keeps bad data out of a local network—this technology creates good data within the local network. Some data on the network only exists when a person is present and using a device or is found in much greater quantities. Since these all involve active analysis, the results can be available immediately. The beginning of occupancy is apparent immediately, but the end is only apparent when no activity is detected for an extended time. The spatial resolution is about the same as with the first category. These are all generic to any organization or building.

<u>Enterprise Applications</u> — Databases, email, *authentication*.

Systems for managing information within a company create traces of activity and hence occupancy. These are usually specific to a particular company, and while focused on the identity of the individual accessing the system, often also provide a network address (e.g., a fixed IP address), which can indicate specific location. As with the IT Network Traffic methods, these involve active analysis so lack significant latencies.

<u>Other IT Systems</u> — *Access control*, **wired phones**, *cameras*.

These may or may not use the IP network, but do not fit into the previous three categories. Cameras could track both people entering and leaving a building; access systems may apply to both, or only to entry.

Building Infrastructure — **Electricity meters**, elevators, water, gas, chilled water, hot water. These systems traditionally were not connected to IT networks to make their data available, but that is changing. When people are present in a building, they use services, which require resources that can be tracked. These are commonly scoped to an entire building, but could be more fine-grained.

A key finding from our work is that data from Wi-Fi systems is the best single opportunity for virtual sensing at this time. Among the reasons for this are:

- Easy to understand for people who don't understand network technology
- Most widespread method available—applicable to nearly any building type
- Simple to implement from an IT perspective
- Modest number of key manufacturers (for commercial sector at least)
- Low latency of detecting arrival and departure of device[5]

This produces the type of data shown earlier in Figure 1. In commercial buildings other than those that can be covered by a single access point, individual access points are scattered throughout the buildings to provide coverage, with a central access point controller device that coordinates their operation (including handoff of moving devices from one AP to another), provides a common point for authentication/security, and manages the entire collection, including archiving data about usage. It is the controller that can provide the data to the building energy systems. The controller can be set to actively "push" the data out on a regular basis, or to respond to queries from the outside to "pull" the data out. Which is possible depends on the manufacturer. A need to make this more available is to make the mechanism as simple as possible, use standard protocols, and be consistent across manufacturers.

Wi-Fi data do have limitations in potentially unknown (and slowly varying over time) conversion of device counts to people, incomplete coverage in some buildings, and possible departure time latency. That said, Wi-Fi data should always be seen as part of a foundation of data for building operation and management, to be supplemented as justified by other methods;

While the underlying principle of Wi-Fi based virtual occupancy sensing is sound, and the data we collected match our expectations about what occupancy data should look like, a critical future research step is to collect some amount of ground truth data to confirm this, and to determine conversion factors from Wi-Fi device counts to people. This should be done on

[5] We have not verified the arrival latency for these systems (the time between when a device enters radio range of an access point and the time it appears in the list of devices that are associated or authenticated), but in a test with data being obtained every 7 seconds, devices appeared to show up in the list without perceivable delay. We believe that leaving time (the time between a device being last seen and the time it is removed from the device list) is 30 minutes by default for the access point control system at LBNL. A lower time-out would be preferable for occupancy estimation purposes, but some other IT data streams have a time-out of one to several hours. On a campus, when a device is seen by an access point at a different building, perhaps even one that covers outside spaces, it will be removed from the earlier access point list so that not all devices take the full time to leave the list.

several building types and locations to observe any variance. The conversion factor may also change over time.

M&V Analysis

Current Practice

Measurement and Verification (M&V) is normally used to quantify the effectiveness of energy conservation measures, and as part of the Transactive Controls project, we wrote software for this purpose. The M&V Agent uses a statistical model that accounts for the regular weekly pattern of building load, as well as adjusting for outdoor air temperature, to predict the building load if there is no change in building behavior. The predicted load is then compared to the actual load. If the actual load is consistently lower than the prediction, then the energy conservation measure has been successful. In subsequent Transactive Controls work, we also wrote software for M&V of non-temperature dependent, scheduled lighting controls; the work in this report focuses on whole-building level M&V applications, however occupancy is an important parameter in the control of lighting end uses as well.

Researchers at Google are investigating the model used in the M&V agent (described here) to compare its predictions with those of other models, with a specific interest in predicting the peak daily load. As of the end of January 2015, they reported that the LBNL model was the best performer by a small margin, but they did not share the project details. In a larger study, the LBNL model was one of 10 models (both proprietary, and published) that were tested to evaluate their accuracy in predicting energy, specifically for M&V applications (Granderson et. al 2015 a; Granderson et al. 2015 b). The LBNL model performed well, demonstrating solid accuracy relative to the current state of the art.

LBNL Building 46A

One of the test case buildings for the Transactive Controls project is LBNL's Building 46A. Even though no energy conservation measures have been undertaken in the building, the M&V Agent routinely calculates the predicted baseline load and compares it to the actual load. This comparison revealed a change in building behavior that, when we investigated, revealed an energy saving opportunity. In essence, the M&V Agent served as a Fault Detection agent.

The case is illustrated in Figure 3. The upper panel shows the first three weeks in October, and the lower panel shows the following three weeks. In both cases, the blue line shows the predicted load from a statistical model based on data from September and the black line shows the measured load.

For the first three weeks in October the model matches the actual load fairly accurately: the RMSE is 1.6 kW. The model noticeably under-predicts the load on a few afternoons but otherwise performs well.

The situation changes dramatically sometime around the end of October (the middle of the second panel): the RMSE in the predictions doubles to 3.2 kW. As the figure shows, this happens because building now uses much more energy in the mornings than it did previously. The large change in the prediction error prompted us to look at plots like those in the figure, which revealed the change in behavior in the morning. We investigated and found that this occurred because the building was put into a winter heating mode that caused the building to warm up to a comfortable temperature every morning before working hours. The building could have achieved its warmer temperatures efficiently and gradually by using its heat pumps, which is the way it had been behaving previously. Instead, in its winter heating mode it was set to warm the building as quickly as possible, so it used its supplementary system of resistive heating, which is energy-intensive.

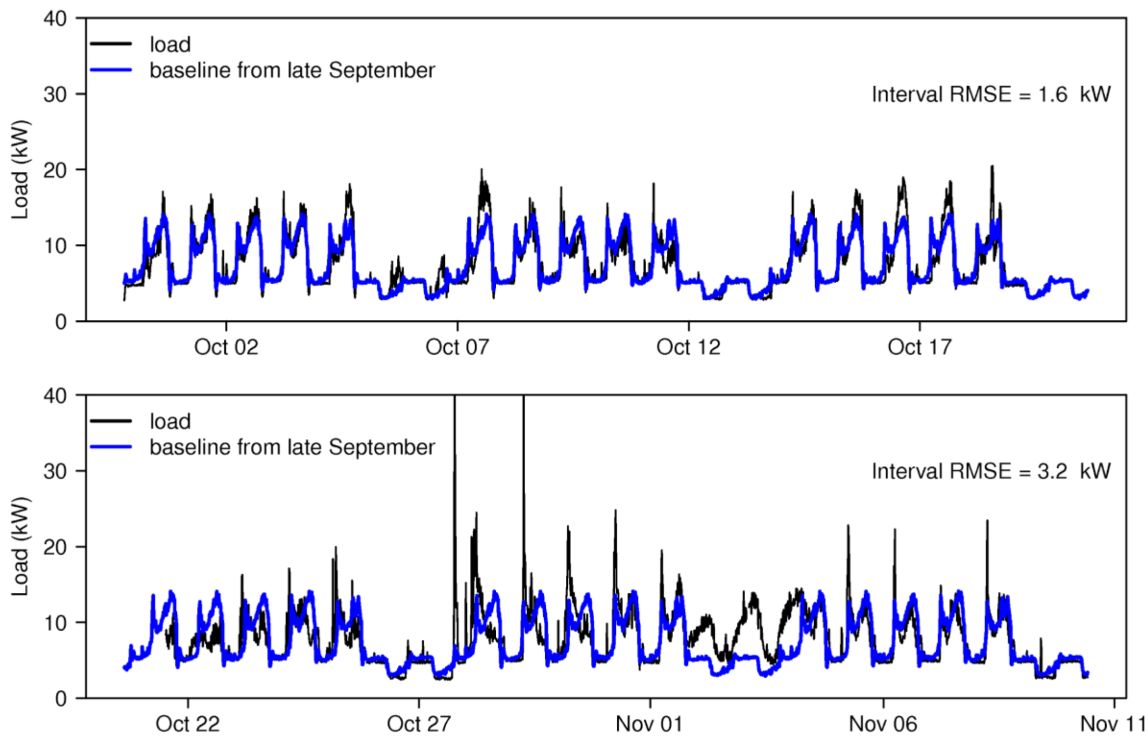


Figure 3. Load vs. Baseline in LBNL Building 46A (resistive heating)

PNNL building

The M&V Agent was also used to look for changes in behavior in the building PNNL is using as a test case for the Transactive Controls project. Results are shown in Figure 4. The upper panel shows data from two weeks in April, and the lower shows two weeks in May. As before, the blue line shows the baseline prediction and the black line shows data. In the upper panel the load follows the baseline prediction rather closely for almost the entire two weeks, with just a few short intervals that deviate substantially. In the lower panel something has changed and the load differs greatly from the prediction. Specifically, in the upper panel both the actual and predicted load show a large morning peak, similar to the resistive heating peak in LBNL's Building 46A, but in the lower panel the morning peak no longer occurs and the peak load has shifted to the afternoon. We have not yet worked with PNNL to understand the cause of this change, but it is another illustration of how the M&V Agent can be used to monitor building performance and recognize when a change has occurred.

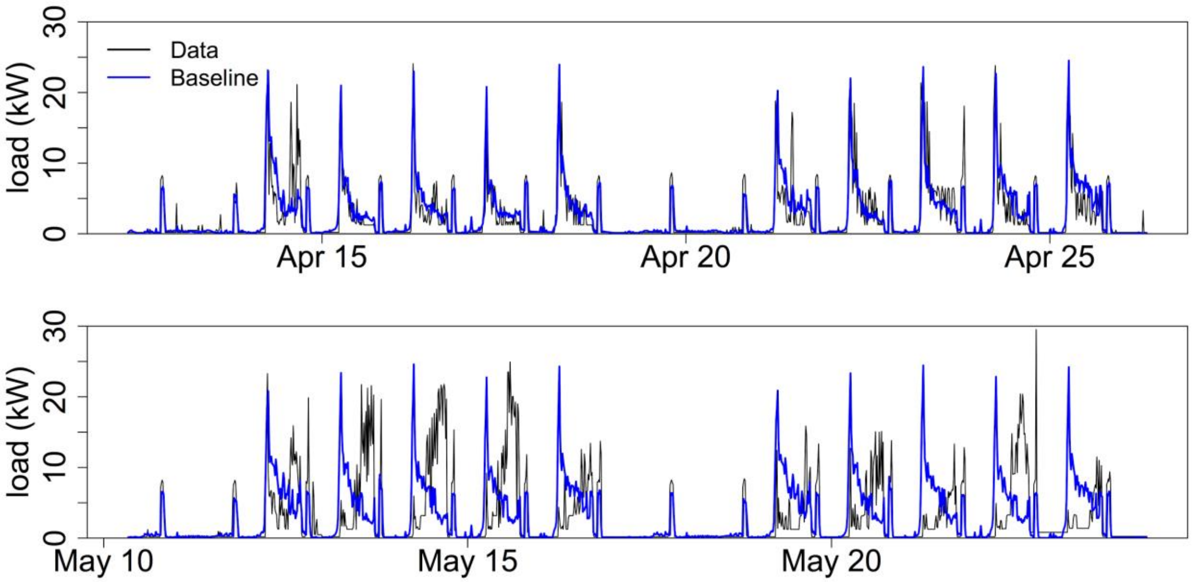


Figure 4. Load vs. Baseline in PNNL building

As can be seen from the two examples above, even without data on occupancy the M&V Agents produce baseline predictions that are of useful accuracy. But some of the unpredicted variation in load is undoubtedly due to changes in the number of building occupants, and the ability to adjust for such changes would improve the accuracy of the models. We now discuss the steps we have taken so far to incorporate occupancy proxy data. This work is in its early stages.

M&V with Proxy Occupancy Data

In this project we sought to identify sites at UC Berkeley and at LBNL where we could collect both energy data and virtual sensing data. To test the concept of "virtual sensing" of occupancy, we wanted to find buildings whose occupancy changes with time, and for which we can obtain load data and proxy occupancy data. We selected two buildings at LBNL for the test, and selected the end-of-year holiday of 2014-2015 as a period when we knew occupancy would be low in both buildings. Our approach is to create a statistical model based only on load data from the *non*-holiday period, and then use that model to predict the load during the holiday.

For both buildings, we obtained the load data, and data on the number of Wi-Fi connections in the building for each 15-minute interval. The Wi-Fi data serve as a proxy for occupancy data.

LBNL Building 46A

LBNL's Building 46A is a single-story office building with about 25 offices. Its electric load as a function of time is shown in Figure 5, along with outdoor air temperature, for six weeks in late 2014 and early 2015. During the holiday period, when LBNL is nominally shut down, the building still experienced quite high morning peaks in electric load...higher, in fact, than on typical non-holidays. What's more, even the mid-day load is higher during the holiday than it is on weekends in the non-holiday period. Another recognizable feature is that the two days before the holiday (Dec. 22 and 23) have lower load than typical workdays for most of the day, although their peak loads are far exceeded on many days during the holiday break.

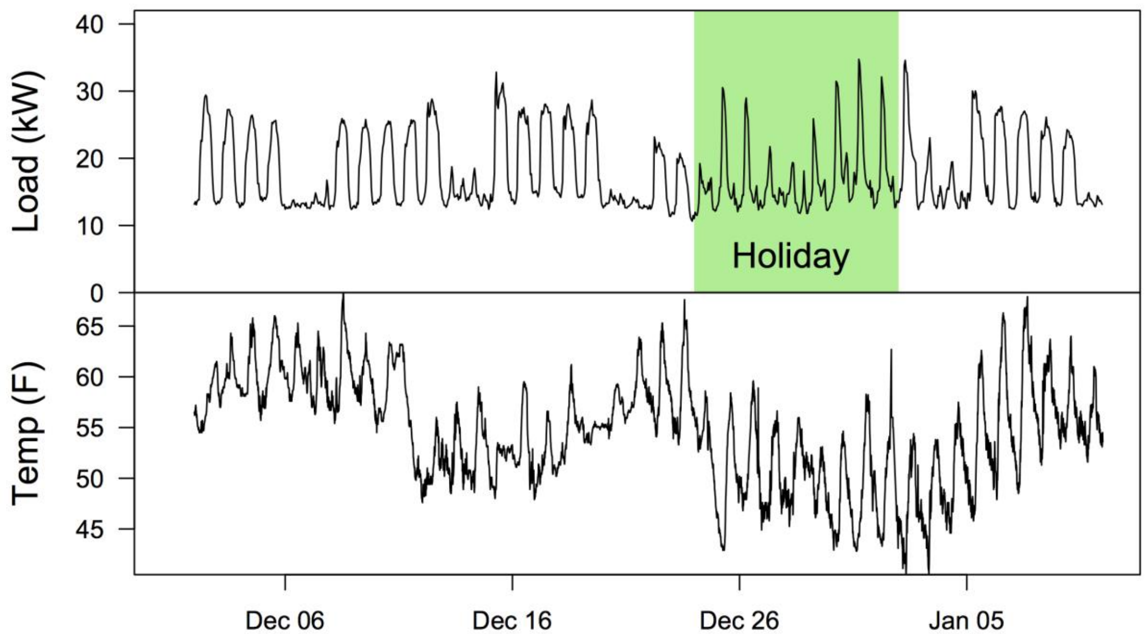


Figure 5. LBNL Building 46A Load and Outdoor Air Temperature, December 2014 – January 2015

Without implicit sensing of occupancy, it's difficult to know how much of the electricity consumption in this building depends on occupancy, how much depends on outdoor air temperature, and how much is independent of either (because it is constant or is on a schedule). Fortunately, in this building the number of connections to the local Wi-Fi network is recorded. This means laptops and phones that automatically connect to the network are counted even if they are not being used for Internet access at a specific time.

Figure 6 repeats the plots shown above, and also shows the number of Wi-Fi connections. The Wi-Fi data confirm that the building was almost or completely unoccupied during the holiday period; the explanation for the high daily load peaks does not involve people coming in to finish work before the end of the year.

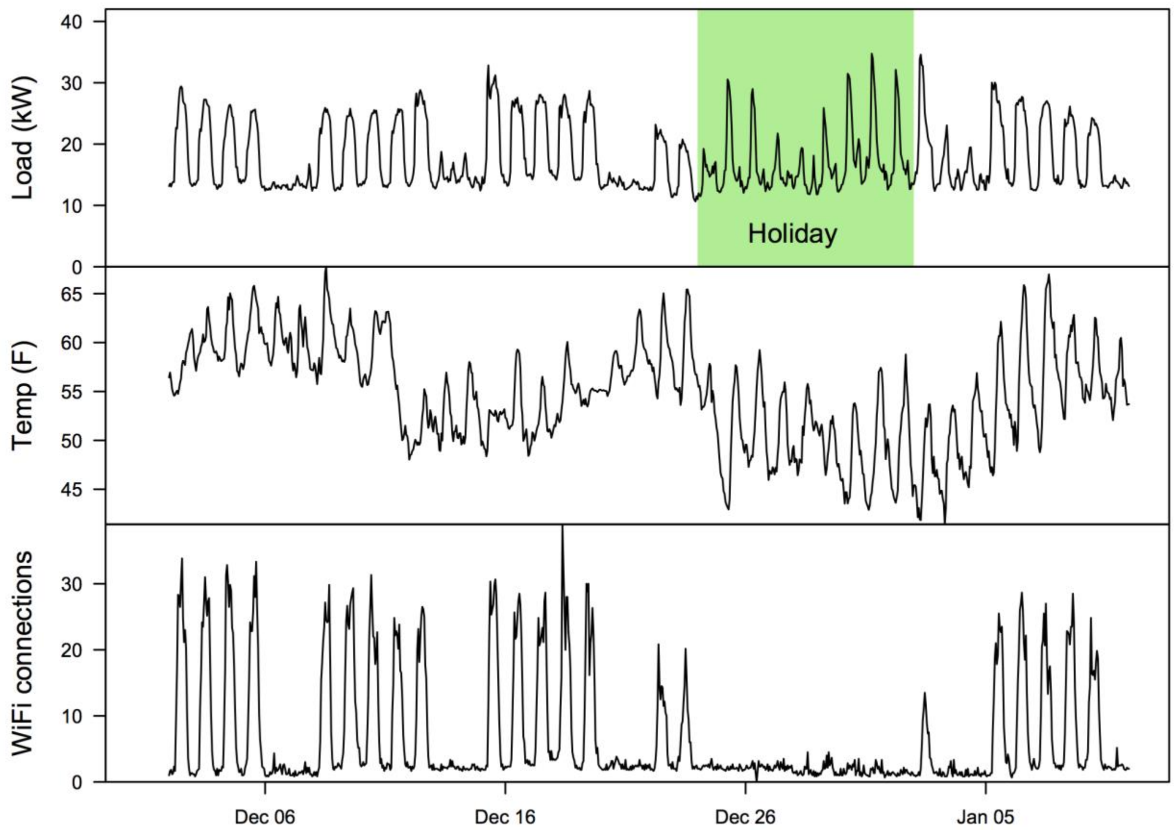


Figure 6. Load, Temperature, and Wi-Fi Connections in LBNL Building 46A

We extended our standard statistical baseline model to include the following components, which sum to the total predicted load:

- A repeating weekly load pattern
- A portion of the load for which there is a piecewise-linear relationship between temperature and load during the first few hours of the workday
- A portion of the load for which there is a separate piecewise-linear relationship between temperature and load during the rest of the workday
- A portion of the load that is proportional to the number of Wi-Fi connections

We fit the model to the data from before and after the holiday period -- referred to as the "training period" -- and then used that model to predict the load during the holiday, This mimics the procedure that would be used to perform M&V. Results are shown in Figure 7, both for the new model (blue line) and for an identical model that does *not* include Wi-Fi data (red line). The upper panel shows the models predictions during the training period; the lower panel shows the fit to the prediction period. As is evident in the plots, both models do about equally well at fitting the load during the training period, but the model that includes the Wi-Fi data performs far better during the holiday period. The model that does not use Wi-Fi data over-predicts the total energy used during the holiday period by 21%; the model that uses the Wi-Fi data over-predicts by only 11%

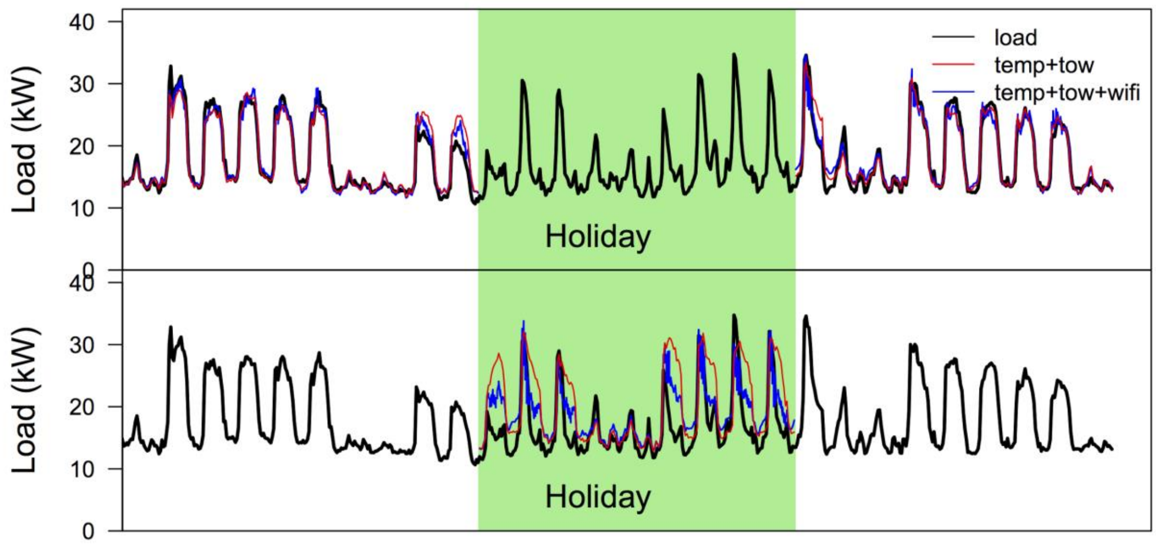


Figure 7. Load vs. Model Predictions for LBNL Building 46A with and without Wi-Fi data

LBNL Building 90

LBNL's Building 90 is a large office building: it has over 100,000 square feet of occupied space and hosts several hundred occupants at peak hours. Figure 8 below shows building load as a function of time (top panel), as well as outdoor air temperature (middle panel) and WiFi connections (lower panel). Only five weeks of data are shown, out of a much longer series.

Several features of the occupancy data from Building 90 echo those from Building 46A: judging from the number of Wi-Fi connections, occupancy was substantially lower on December 22 and 23, and January 2, than on typical workdays. However, the load data series is strikingly different from that for Building 46A: the load on December 22 and 23 was almost as high as on other workdays, in spite of the low occupancy. And, unlike Building 46A, there are no spikes in load during the holiday, although the daily load peaks are much higher than on weekends.

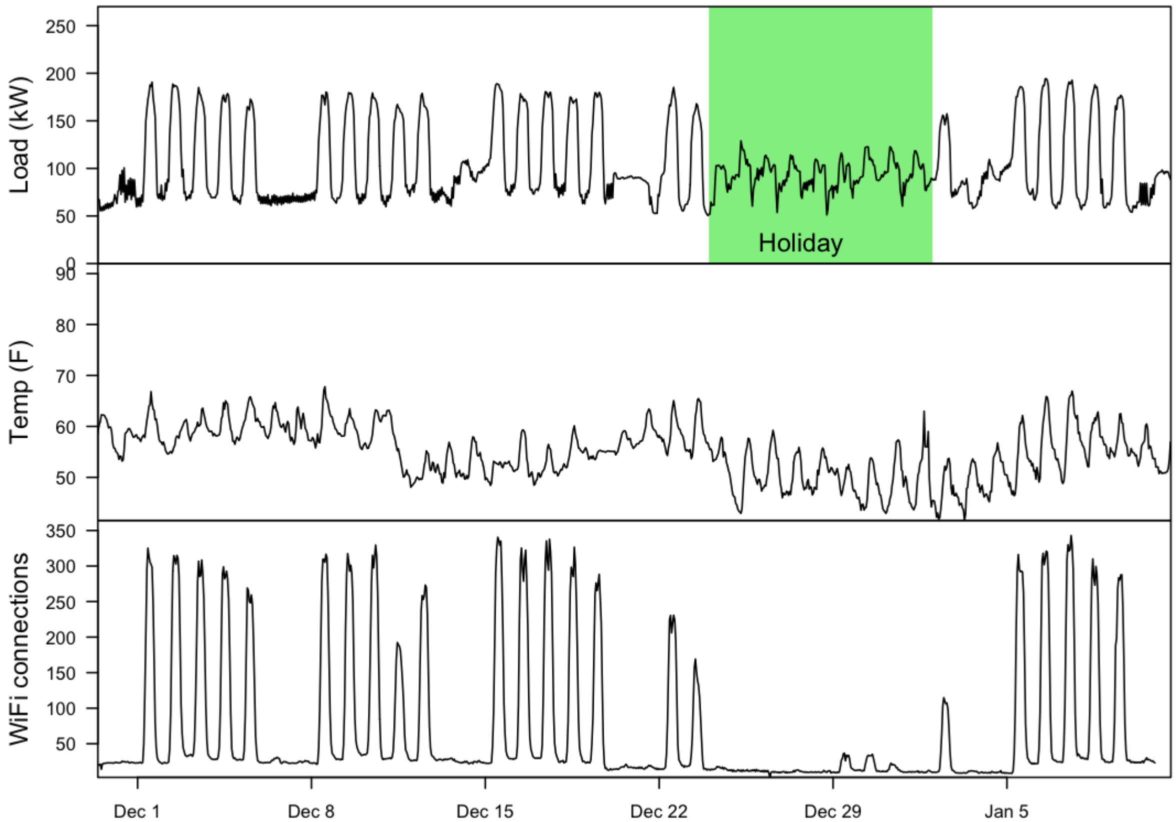


Figure 8. LBNL Building 90 load as a function of time, temperature, and Wi-Fi connections

We fit the same two statistical models for Building 90 that we used for Building 46A. Results are shown in Figure 9. As with Building 46A, the model that includes Wi-Fi data performs much better at predicting the energy used during the holiday period: without the Wi-Fi data the prediction is 23% too high, but with the Wi-Fi data it is only 7% too high.

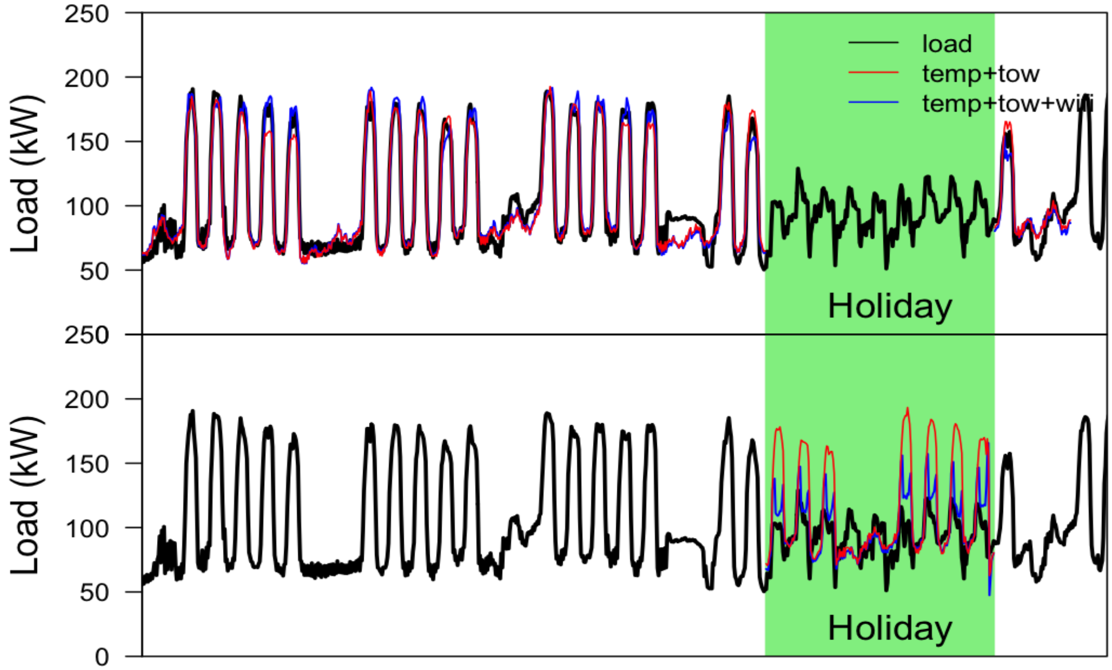


Figure 9. Model Predictions for LBNL Building 90

Discussion of both buildings

Predicted energy consumption and load shapes in both Building 46A and Building 90 are greatly improved when the number of Wi-Fi connections is used as a predictive variable. This makes sense because the number of connections is presumably closely related to the number of people in the building, and people use energy, primarily through lighting and computers. Tables 1 and 2 below summarize the calculated errors in the predicted energy used during the holiday period, in both buildings, with and without using the Wi-Fi data, according to two metrics - relative bias (how big was the error in the predicted energy use as a percentage of the actual energy use) and root mean squared error (RMSE).

Table 1. Relative bias comparisons of model predictions

Building	Relative Bias	
	Without WiFi Data	With WiFi data
LBNL Building 46A	21%	11%
LBNL Building 90	23%	7%

Table 2. Root Mean Squared Error (RMSE) comparisons of model predictions

Building	RMSE in hourly predictions	
	Without WiFi Data	With WiFi data
LBNL Building 46A	26 kW	16 kW
LBNL Building 90	40 kW	21 kW

Even when Wi-Fi data are used, the statistical models over-predict the amount of energy used in both buildings. Although it's hard to judge from a small sample --- perhaps if we looked at a third building, the predictions would be too high rather than too low --- we suspect that over-prediction will be the norm for extreme cases in which the building is completely unoccupied, as in this example. This is expected because lighting energy use, in particular, is not expected to be proportional to the number of occupants: in both buildings, some areas are lit by a bank of lights controlled by a single switch, and in those areas the same amount of lighting energy will be used whether only a single person is present, or several people. If the training period included some workdays when the building is nearly unoccupied --- or if we included one or two of the holiday days in the training period --- then the model could be extended to include a binary effect determined by whether the number of Wi-Fi connections is or isn't above its base value. This would lead to a better-fitting model, but we have not pursued this approach because the use case would be rather artificial.

Overall, the results confirm the expected outcome that data that track the number of building occupants can substantially improve the predicted baseline energy use in buildings for which occupancy varies with time. This is an important finding because unknown or un-quantified changes in occupancy often lead to large errors in baseline predictions, and therefore can complicate or invalidate efforts to measure and verify energy savings. The ability to incorporate virtual sensing data to reduce baseline errors will improve the accuracy of energy efficiency measurements. Future work could compare these results with models that include absolute occupant measurements, which were not available for these buildings.

UC Berkeley

Beginning in early May 2015, the UC Berkeley information technology team began logging Wi-Fi data from several campus buildings, at our request. Data from two of the buildings are shown in Figure 10 and Figure 11 below. Each figure shows the load data, Wi-Fi data, and outdoor air temperature for nearly seventeen weeks.

These buildings illustrate the extremes of this small collection: in Stephens Hall (Figure 10) the weekly pattern of electric load repeats for the entire four-month span with little variation, in spite of some variation in occupancy patterns. In North Gate Hall (Figure 11), on the other hand, the peak load on highly occupied weekdays is more than 30% higher than on weekdays with lower occupancy. We discuss North Gate Hall in more detail, below.

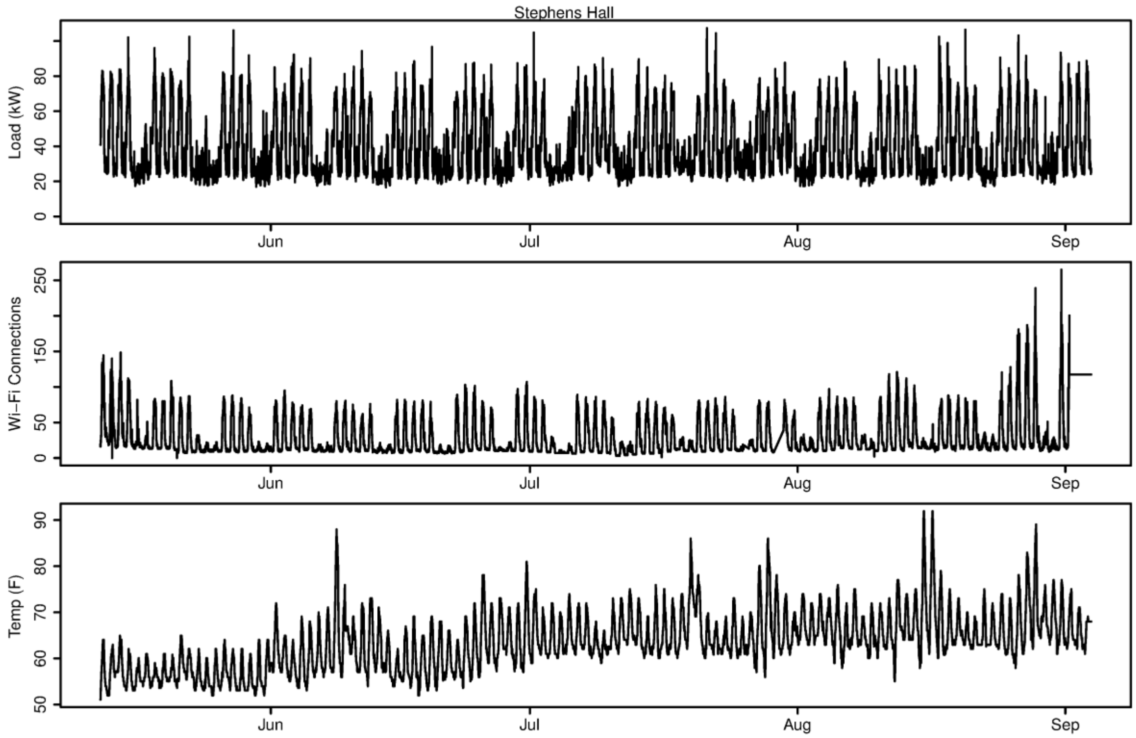


Figure 10. Load, Wi-Fi, and Temperature data for UC Berkeley Stephens Hall

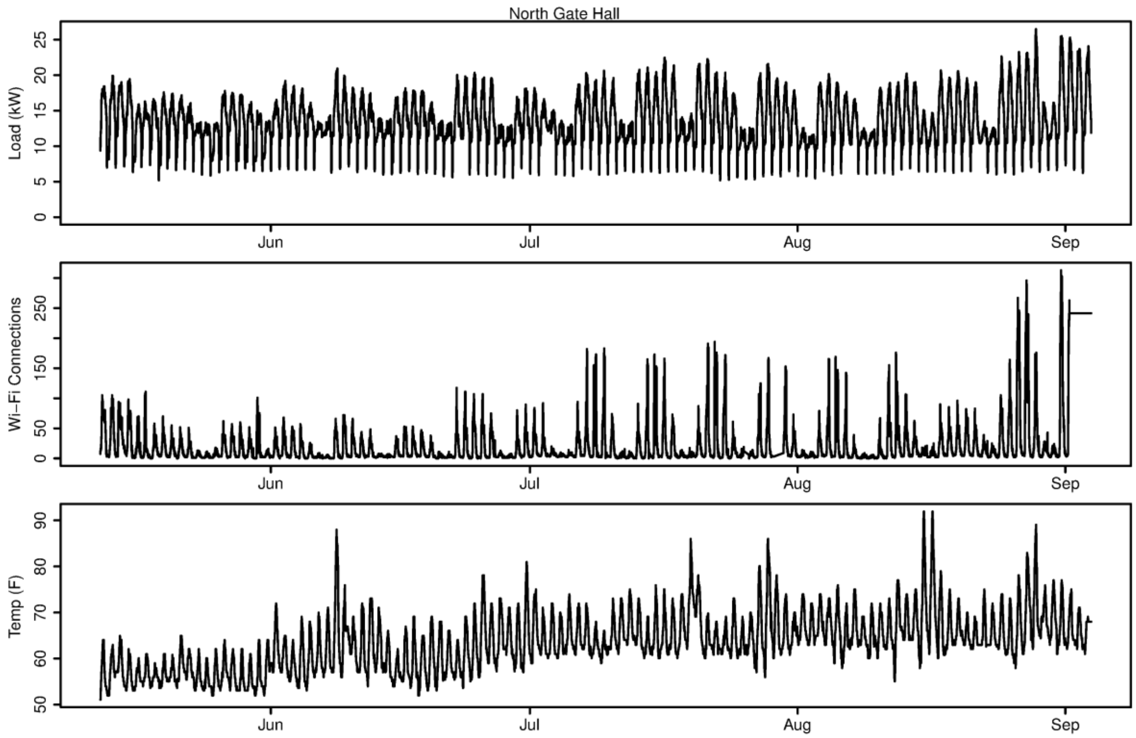


Figure 11. Load, Wi-Fi, and Temperature data for UC Berkeley North Gate Hall

For each of the UC Berkeley buildings, as with the LBNL buildings, we fit a model to predict 15-minute load from time of week indicator variables and outdoor air temperature, using different piecewise-linear functions of temperature for times when the building is and isn't in "occupied mode." We fit these models with and without using Wi-Fi data as an additional predictive variable.

Additionally, we tested a modified version of the model that uses a piecewise-linear model for the relationship between load and the number of Wi-Fi connections, so that the additional load per connection is estimated separately when the number of connections is below or above the median. The idea behind this model is that even a relatively small group of building occupants might be expected to have a large effect per person: For example, a room's lights need to be on whether there is one person present, or ten.

North Gate Hall

North Gate Hall at UC Berkeley is a small classroom and office building that houses the journalism department. In contrast to some of the larger buildings on campus, a substantial portion of North Gate's load appears to be related to the number of occupants.

In the figure below, we show the final four weeks of data, and predictions from the model with and without Wi-Fi data used as a predictive variable. When the new semester started at the end of August and building occupancy increased, the model that includes Wi-Fi data did a much better job at predicting daytime load.

For the dataset as a whole, the root-mean-squared error (RMSE) in the hourly prediction was 0.91 kW when Wi-Fi data were used as a predictor, compared to 1.09 kW when Wi-Fi data were not used. That is, errors were about 20% higher when Wi-Fi data were not used.

The improvement was even more dramatic for daytime hours: the RMSE was to 1.07 kW when Wi-Fi data were included in the model, and 1.40 kW (that is, 30% higher error) without Wi-Fi data.

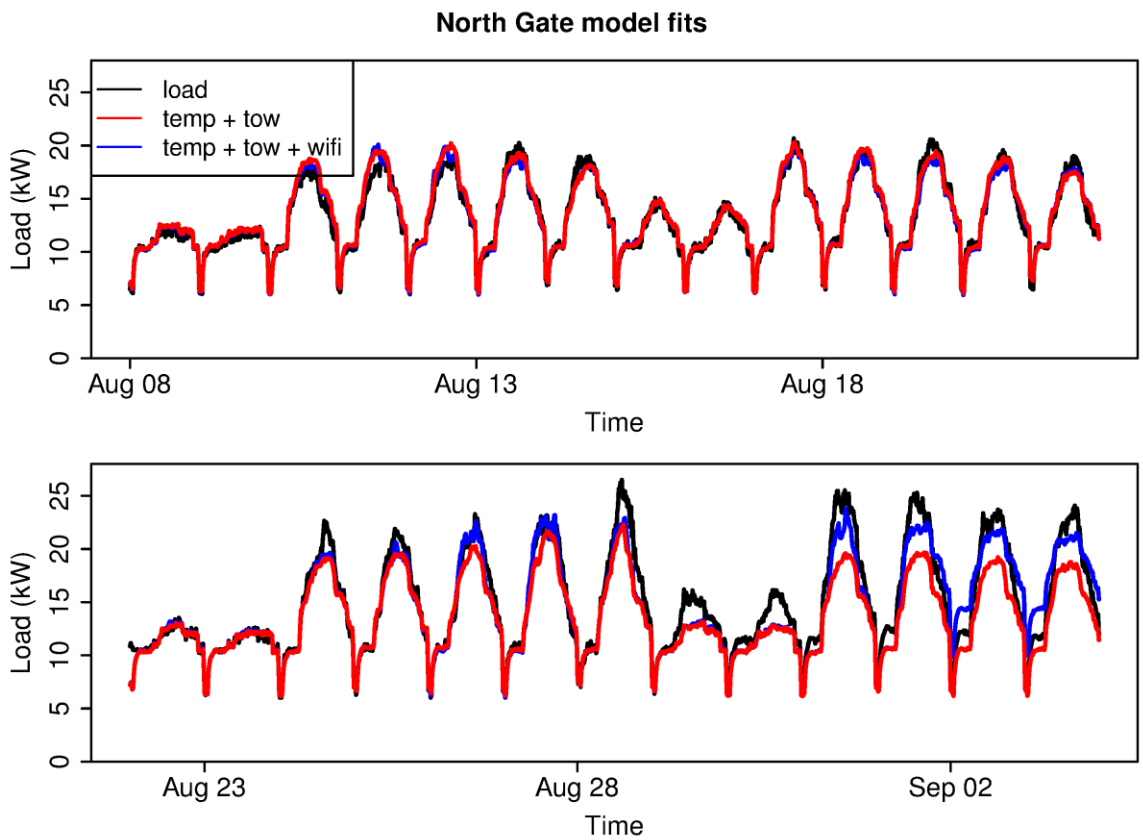


Figure 12. Load vs. Model Predictions for UC Berkeley North Gate Hall

The large amount of diurnal variation and weekday/weekend variation in the load can make it hard to see some of the regular patterns, so it can be helpful to focus on some specific weekdays and times. Figure 13 shows average load, Wi-Fi connections, and outdoor air temperature in North Gate Hall, just for weekdays between 12:00-14:00. Average load varies by only a few kW within a typical week, even though the number of Wi-Fi connections (and presumably building occupancy) is much higher on Tuesdays, Wednesdays, and Thursdays than on other days for most of July and August.

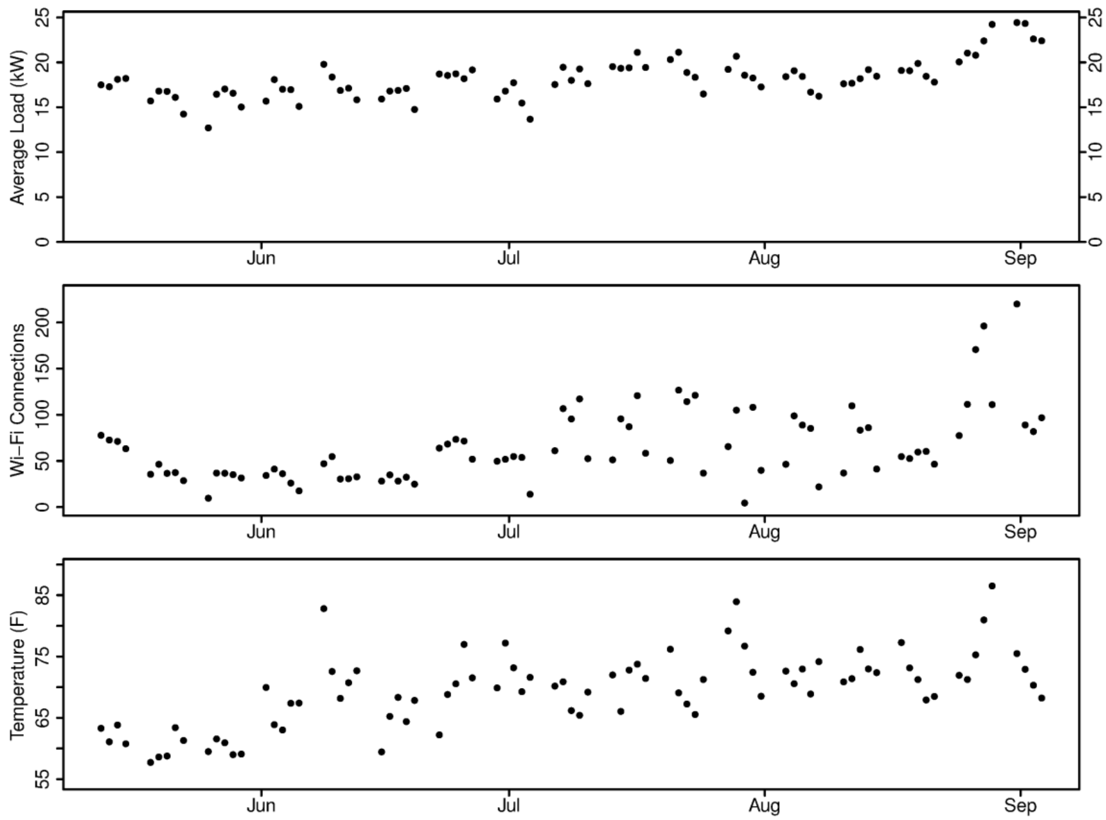


Figure 11. North Gate Hall: Average load, Wi-Fi connections, and outdoor air temperature, for 12:00-14:00 on weekdays only.

Cursory visual inspection of the load, Wi-Fi, and temperature time series does not suggest a strong connection between Wi-Fi connections and load, but in fact such a connection is present, as shown in Figure 14. Based on the regression coefficients from the model (described in an appendix) that includes time of week, temperature, and number of Wi-Fi connections, each of the 100 Wi-Fi connections is associated with 2.70 +/- 0.05 kW of load. Overall, occupancy variations account for roughly 2% of the load variability during these times of day, and on especially high-occupancy days about 4% of building load is attributable to occupancy.

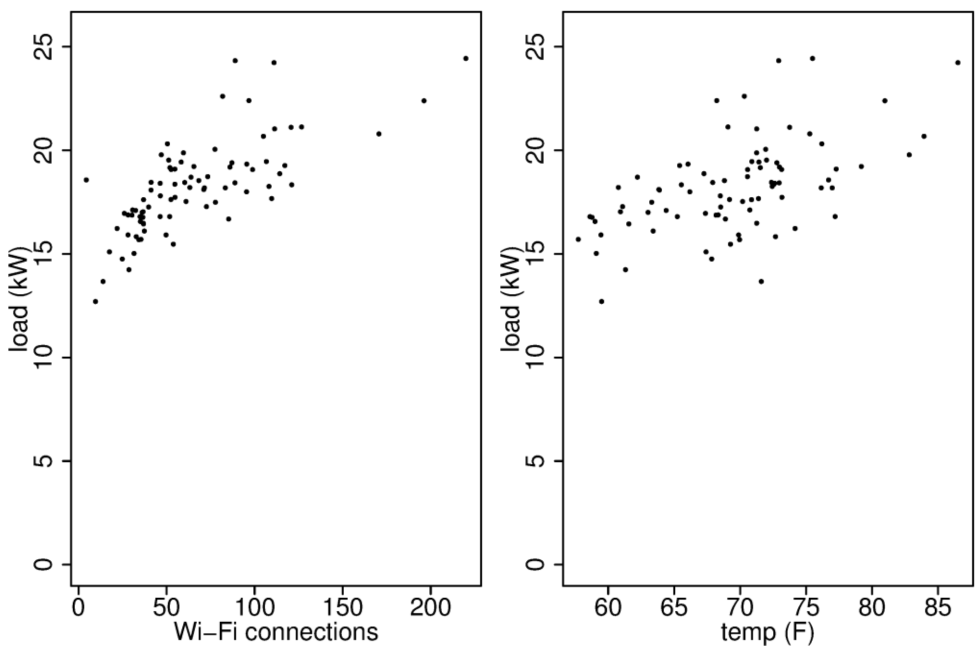


Figure 14. North Gate Hall: Average load vs. average number of Wi-Fi connections (left panel) and vs. average outdoor air temperature (right panel) from 12:00-14:00 on weekdays.

Table 3 summarizes model fit results from all five UC Berkeley buildings and both LBNL buildings. In this table we summarized the goodness-of-fit statistics for the entire multi-month dataset for each building. Note that this included holiday periods for the LBNL buildings but not for the other buildings.

The reduction in prediction error that is obtained by including occupancy data is not an intrinsic quality of a building: it will depend on the time interval covered by the model, because both the amount of occupancy variation and the amount of load variation will depend on the time interval. In the case of the LBNL buildings, we deliberately chose a time interval that includes holidays, thus leading to larger load and occupancy variation than typically occur in those buildings. Still, the LBNL analyses are informative because some more typical M&V applications also involve large variations in occupancy. For example, large occupancy variation can occur when a tenant vacates a multi-tenant building, or when an employer adds or eliminates a large number of employees. Both of those cases can be problematic for conventional M&V approaches, and the LBNL holiday examples demonstrate that it may be possible to use implicit occupancy sensing to greatly improve baseline predictions in such cases. Analyses of data from buildings that actually experience those types of occupancy changes will be needed to confirm that this approach works with useful accuracy.

Table 3. Goodness-of-fit statistics for models that do and don't include Wi-Fi data, for several buildings on UC and LBNL campuses.

Building	Coefficient of Variation (CV) of daytime occupancy	Mean daytime load (kW)	RMSE in predicted load during daytime intervals (kW)		Reduction in RMSE (%)
			Without Wi-Fi data	With Wi-Fi data	
North Gate	1.12	36	1.4	1.07	24
Wurster	0.84	154	17.5	13.9	22
Stephens	0.62	44	6.44	5.87	9
Moses	0.8	25	3.25	3	8
RSF	0.93	139	6.09	6.01	1
LBNL 46	0.87	12	3.17	1.7	46
LBNL 90	0.78	150	21.9	17.5	20

The UC Berkeley buildings experienced more typical occupancy variation, both within and between days. In only two of those buildings, too, the use of occupancy data led to a substantial decrease in prediction error, as measured by the root mean squared error in the load predictions.

A striking fact about all of the UC Berkeley buildings (and about the LBNL buildings during non-holiday periods) is that the load is not occupancy-dependent. This may seem to be contradicted by the fact that occupancy data do substantially improve the baseline predictions in several buildings, but there is no contradiction, merely two ways of looking at the same results. On one hand, the prediction error in Wurster Hall, for example, was reduced by more than 20%. On the other hand, the error was reduced from about 11.5% (17 kW out of 154 kW) to about 9.1%, by changing the prediction by a few kW at high-occupancy times. In short:

- When there are large changes in occupancy, including proxy occupancy data may be essential for reducing errors in the predicted baseline.
- However, in our small sample of buildings, only a small fraction of the load was sensitive to occupancy for the range of occupancy values that is normally experienced.

Summary, Conclusion and Future Research

Virtual sensing appears to be a promising source of readily available, low cost, fine-grained and reliable occupancy data for M&V. Other potential uses of these occupancy data include forecasting electric loads, fault diagnostics, and control algorithms and schedule. It also can be incorporated directly into a transactive network.

Buildings vary as to how occupancy and energy use is correlated. We found that virtual sensing data can substantially reduce baseline errors – by more than 20% in some buildings – and thereby improve the accuracy with which energy savings can be measured. Use of data from physical occupancy sensors could produce better results, but it is highly unlikely that installing such sensors would ever be justified by M&V purposes so that the choice in practice is between having virtual occupancy data or none at all. We do not have quantification of the difference in performance between physical and virtual occupancy sensors. However, since not all buildings have load that depends strongly on the number of occupants, some building baseline models are not substantially improved by including occupancy data. In the small number of buildings we have investigated to date, load in larger buildings was predicted fairly accurately without proxy occupancy data; whereas in smaller buildings the load was less predictable in general and proxy occupancy data can significantly reduce errors. If this result holds for buildings in general, virtual sensing may be important in M&V in small buildings, which is a building population that has historically presented special challenges.

Additionally, our results suggest that baseline models that include occupancy data may be useful beyond M&V. Ideally, the electricity patterns in buildings *should* depend strongly on occupancy since the majority of energy in office and similar buildings is used to provide services to occupants such as lighting, heating, cooling, and other services to occupants. A building with loads that are not occupancy-sensitive may be a candidate for energy conservation measures such as demand-controlled ventilation, or occupancy sensors to control lighting.

Further research is needed to better understand how to simplify the data collection and analysis methods for proxy data. Ideally these proxy data would be verified against absolute energy counting to better understand and evaluate the robustness of proxy data. Further research is also need to continue to collect data occupancy data for a larger sample of buildings and explore how to improve M&V baseline models. Finally, research is needed on methods to extend the use of the occupancy data into applications related to forecasting loads, demand response predications, control, and fault diagnostics. This technique will help reduce energy when fewer occupants are in a building or help better link and understand the relation between energy use and occupancy. Future work is also needed to evaluate the trade-offs between virtual sensing and absolute occupancy measurement. This comparison would include evaluating the costs for different types of occupancy data and comparing the performance and analysis of in baseline models. Our hypothesis is the virtual sensing will provide most of the value for minimal costs. This concept, however, needs to be further evaluated in pilot studies.

References

Brambley, M., P. Haves, S. McDonald, P. Torcellini, D. Hansen, D. Holmberg and K.Roth, 2005. Advanced Sensors and Controls for Building Applications: Market Assessment and Potential R&D Pathways. PNNL Report 15149.

Efficiency Valuation Organization (EVO). International performance measurement and verification protocol: concepts and options for determining energy and water savings, vol. I. January 2012. EVO 10000-1; 2012.

Granderson, J, Touzani, S, Custodio. C, Sohn, M, Fernandes, S, Jump, D. M&V shootout: Setting the stage for testing the performance of new energy baseline models. Proceedings of the 2015 International Energy Program Evaluation Conference, Long Beach, CA.

Granderson, J, Touzani, S, Custodio, C, Fernandes, S, Sohn, M, Jump, D. Assessment of automated measurement and verification (M&V) methods. Lawrence Berkeley National Laboratory, July 2015. LBNL#-187225.

Granderson, J, Price, P, Czarnecki, S, Page, J, Brown, R, Piette, MA. Development of diagnostic and M&V agents, and implementation in an occupied office environment. Lawrence Berkeley National Laboratory, March 2015. LBNL-6973E.

Katipamula, S, R Underhill , D Taasevigen , MA Piette, J Granderson, J Goddard, T Kuruganti. 2012. Small- and Medium-Sized Commercial Building Monitoring and Controls Needs: A Scoping Study. PNNL Report 22169.

LBNL, Innovative Occupancy Detection Technologies, Draft Plan for Specification, Development, and Testing (Sensors & Controls core task), March 10, 2014a.

LBNL, Innovative Occupancy Detection Technologies, Draft Specification (Sensors & Controls core task), March 15, 2014b.

Li N, Calis G, Becerik-Gerber B. Measuring and Monitoring Occupancy with an RFID Based System for Demand-Driven HVAC Operations. Automation in Construction 24:89-99, 2012.

Melfi R, Rosenblum B, Nordman B, and Christensen K. Measuring Building Occupancy Using Existing Network Infrastructure. International Green Computing Conference and Workshops, July 2011.

Nordman B, Christensen K, Melfi R, Rosenblum B , and Viera R. Using Existing Network Infrastructure to Estimate Building Occupancy and Control Plugged-in Devices in User Workspaces. International Journal of Communication Networks and Distributed Systems, Vol. 12, No. 1, pp. 4-29, January 2014.

Piette, M.A. R. Brown, P. Price, J. Page, J. Granderson, D. Reiss, S. Czarnecki , G. Ghatikar and S. Lanzisera. Automated Measurement and Signaling Systems for the Transactional Network. Lawrence Berkeley National Laboratory report LBNL-6611E. January 2014.Touzani, J., S, Custodio. C, Sohn, M, Fernandes, S, Jump, D. M&V shootout: Setting the stage for testing the performance of new energy baseline models. Proceedings of the 2015 International Energy Program Evaluation Conference, Long Beach, CA.

Appendix A:

Catalog of occupancy-related data streams

This Appendix reviews many potential sources of virtual sensing data for the underlying principles, and issues with acquiring, processing, and using the data. It is a "Catalog" of the types of existing networks that could be used for occupancy, including for each:

- The underlying "sensing phenomenology"
- Ease or difficulty of access to the data
- Data processing needed
- External data needed (e.g., mapping IT network nodes to physical spaces)
- Granularity in time, space, and people (e.g., anonymous vs. identified)
- Best application contexts
- How standards could make these more readily available

A need in buildings is for an "Occupancy Server" — the entity that gathers occupancy and related data from other devices, processes and aggregates this data, and provides it to devices that can use the information. Key example users are lighting and HVAC control systems, and into building operators. Ideally the occupancy server is not a stand-alone device, but rather a function (service) of a device that already exists for some other purpose.

This discussion oriented to office buildings, but many methods apply to other building types. Virtual sensing is best applied to larger buildings with relatively uniform networks and usage, but once developed can be applied to nearly any building type.

Results can be of several types:

- Individual result—typically for a single room or other defined space.
  May be derivative of data about a person or device. Individual results can be used to control lighting or HVAC.

- Collective result—for a group of rooms, zone, floor, or an entire building.
  We refer to these as "zone" results. Whole-building estimates are the most common of these, but the scope of a zone could be smaller than the whole building. Zone results are suitable for controlling HVAC and other building-wide services. They also can be used for automatic holiday and daylight saving time detection.

Virtual sensing data can be of a variety of forms: available always, or only intermittently; firm or probabilistic; more likely to overestimate or underestimate occupancy, etc. Any collection of individual results can be aggregated to a collective result (sometimes with a scaling factor to account for people not assessed). Some methods allow either type of result, with the collective result being simpler to obtain.

Virtual Sensing methods fall into several Tiers based on what is involved in the data becoming available, as shown in Table A-1. Most methods in this catalog are Tier 1 methods, but we also consider a few Tier 2 and Tier 3 methods.

Table A-1. Tiers of Virtual Sensing (Nordman et al., 2014).

Tier 1	requires no modification to existing systems other than a data collection and processing point.
Tier 2	involves the addition of software to existing infrastructure to make existing occupancy related data available.
Tier 3	involves the addition of software and hardware to introduce new sources of occupancy data to existing systems.

Each method is based on some principle of operation and has limitations and variations. They may require additional data, require calibration, raise privacy concerns, or suggest research issues.

For all methods, obtaining and analyzing example data clarifies some aspects of how to conduct the method, how to interpret the results, show how useful they are, and indicate what further research is needed. For this reason, some methods below do not include a listing of needed **future** research.

Virtual Sensing leverages systems installed for other purposes. The other systems addressed in this report are Internet Protocol Networks (Network Presence and Data Traffic), Enterprise Applications, Other IT Devices, and Building Infrastructure Devices.

Tier 1 Methods

Internet Protocol Network Presence

Presence of a device on a local IT network may indicate the presence of the user of that device in the building or in a specific room. Network presence can be observed through a variety of means, as listed below. Network infrastructure in a building is a combination of wired (mostly Ethernet) and wireless (mostly Wi-Fi), and these can be integrated to varying degrees (which affects what additional infrastructure is shared or separate). The hardware involved includes end-use devices, switches and routers, and servers, which provide network services.

The ideal case is that everyone in a building has a particular electronic device that they use (e.g. a PC), that has a power state corresponding to their occupancy. This discussion assumes that each device powers down at the end of a work day (manually, or automatically based on a timer for non-use), and the PC is powered up soon after the user arrives at work. Thus, the power state of the device is an indicator of the user’s presence. The methods listed in this section are based on this principle (the following section reviews those that rely on the active use of the device being evident on the network).

Some devices on a building IT network are not associated with individuals, such as network equipment, printers, and cameras. These need to be filtered out for data used to indicate occupancy. These are on the network continuously, so that a few days of observation can be used to automatically flag them as such. A PC left on continuously cannot be used as an occupancy indicator with device network presence methods so also should be also filtered out and can be recognized the same way.

In most cases, a network infrastructure device (router, switch, or server) has the needed data for all end-use devices in the building. The occupancy server may need to pull data from that device, so that a change will not be seen until the next pull, introducing some latency. The frequency of pulling the data could be as long as an hour, or as short as is needed. If a system with the data is able to push the data to the occupancy server any time the data changes, this delay can be eliminated. Latency is also introduced by people who don't power up their PC immediately on arrival. People will not be counted if they have no PC or if they leave their PC on 24/7. Over counts may occur if people have more than one PC and this knowledge is not understood by the occupancy server.

Zone data can be obtained from total counts of devices, but to get to individual locations or people, it would be necessary to have a table that maps from each network address to the physical location of the device it corresponds to, and perhaps also the person who uses that device. The table would need to be updated as people move or change devices. There are ways that this could be substantially automated, to minimize the burden that maintaining the database imposes.

For many of these cases, the values from the mechanism are undercounts of total building occupancy. So, to get a better result, a scaling factor is needed. The result will almost certainly be better if multiple methods are combined. For example, some people use only the building's wired network and others only the wireless network, so if they can be combined, the result will be superior to an extrapolation based on only one method.

In many buildings, wired and wireless networks are managed separately so that their results need to be combined for maximum benefit. Issues with their data availability, and the mechanisms to obtain data, may be different. Devices that get network connectivity only with mobile phone technology (e.g. 3G or LTE) are invisible to the local IT network. There are standards for reporting out data from network equipment, notably the IPFIX mechanism defined by the IETF. IT network presence data is inherently about individual devices that could be associated with individual people and so raises privacy concerns.

Research Questions

How easy is it to map device addresses to rooms? What data are useful for this process? Could network routers push some data to an occupancy server (e.g. a new address showing up on the network or one not being seen for a sufficient time period) rather than relying on an external device pulling such data?

Address Resolution Protocol (ARP)

Internet Protocol networks use the Address Resolution Protocol (ARP) to determine the presence of a computer or other device (specifically, its Internet Protocol (IP) address) on a local network, and to obtain its physical layer address (generally the Media Access Control (MAC) address). This is necessary to get data packets to their destination. A network router keeps a table (the “ARP Table”) of addresses that it has seen recently; this table shows what devices are present (or at least, have been present recently) on the network. Any device on the network, including PCs, need to respond to any ARP requests broadcast to stay on the network.

The general method to use ARP for virtual sensing is to first query the router ARP table to obtain the "last-seen" time for each IP or MAC address. Enterprise network equipment generally has the ability to export ARP table contents to a querying device over the network. Then, filter out always-on devices and any device not seen “recently” (perhaps within 15 minutes). A machine that has been seen on the network recently signifies occupancy. This can then be combined with the translation from network address to location to get the occupancy data by location, either to an individual room, or to a zone of network connectivity.

Research Questions

How frequently should ARP data be pulled? How frequently is the timestamp for a device’s ARP table entry usually updated (this addresses how “recently” a device needs to have been seen to conclude it is still present on the network)?

Ping

Ping is the colloquial name for an “Echo Request” message in the Internet Control Message Protocol (ICMP). In essence it is asking the question “are you there” with the device to then respond with “I am here” as the answer. This is a way to interact with a device without interacting with any specific application or functional goal. We group under Ping any method that employs this principal.

Historically, all machines replied to Pings, though with concern about security in the past decade or so, it has become more common for some to have the feature disabled by default or by local convention. That said, within a local network, there are other ways to accomplish the same goal. The most simple of these is to send a directed ARP packet (an ARP message sent to an individual machine rather than one broadcast as most are).

This approach requires actively collecting data from individual devices rather than harvesting it from network equipment, but the end result is quite similar. Since the data collection is active, it can be directed only to the devices of interest. Network security systems may see such probing as indicative of hacker traffic and so raise an alarm; virtual sensing efforts should be done in conjunction with the IT staff to avoid such issues.

In many cases it may be optimal to combine the ARP and Ping approaches. For example, the ARP table (or scanning the entire local subnet) is needed occasionally to identify new devices on the network, that can then be folded into the Ping list. Ping might be useful to identify when a device appears on the network during time someone is likely to arrive (this can be determined from previous day arrival times), with the ARP table used for departure detection (which is less latency-critical). Ping can be used for wired or wireless devices.

Since these methods are oriented to individual devices, they can be also used for related applications such as automatic inventorying of energy-using devices in buildings.

Research Questions

How frequently is Ping disabled on PCs? On other devices?

Dynamic Host Configuration Protocol (DHCP)

The Dynamic Host Configuration Protocol (DHCP) is used on many networks to allocate IP addresses to devices as they need them, on a day-to-day basis, rather than the traditional method of assigning “static” addresses. A network can have a mixture of static and dynamic addresses. An entity on the network called the DHCP server hands out addresses to devices that request them. The addresses are accompanied by a lease expiration time, after which the device is to stop using it. Devices begin to seek to renew the lease after half of the expiration time has passed. Lease times can be from a few hours to many days. When a device is replaced, the MAC address will change, but the human-readable name likely will stay consistent, which provides a way to track user associations with devices over time.

Many DHCP servers will aim to give devices the same IP address on successive lease requests, though as the MAC address will not change regardless, that can be used to track devices reliably over the course of time. The human-readable domain name may also provide insight on the device owner.

A DHCP server can be queried for the devices it has active leases with. This will indicate the last time a lease was renewed. When the device is powered down, it will not attempt to renew its lease. Any lease older than half of its duration is suspect. This gives a solid indication of when a device has come onto the network, but a fairly crude (crude if leases are of days duration) indication of when it leaves. Ultimately it is a similar method in result to ARP, though a single DHCP server may cover a larger amount of a building or campus network than does a single router.

Ethernet Port Status

Ethernet switches, the device at the other end of the Ethernet cable for an end-use device, know the status of each link. In general, when a PC powers down to sleep or off, it drops its link. Exceptions are for PCs that have Wake On LAN (WOL) enabled, or have Network Proxying enabled. These are not widely used today.

Typically, each cable from an Ethernet port runs to a static location. Thus, the table mapping ports to locations should be stable, unlike IP or MAC addresses. The port status information does not indicate which device(s) is connected to that port. Ethernet port status has some similarities as a method to Wi-Fi associations.

Wireless (Wi-Fi)
Wi-Fi is commonly used in office buildings for notebook PCs, smartphones, tablets, and other devices. By including phones, Wi-Fi covers people without a PC but also introduces the possibility of double-counting people who have multiple devices. Notebooks may be used with Wi-Fi always, or only when not at a user's office (where they plug into an Ethernet port). Access points for Wi-Fi can report what devices currently are connected to them. Many wireless networks can report which specific access point a device is connected to, providing some location information even when the owner of the device is not known.

Phones are a particularly interesting case; a user's phone may show up on the Wi-Fi network as the person is approaching their office so that lighting could be switched on before the user arrives at the office. In addition, when a person goes elsewhere in the building, the phone will still be present on the network, allowing the system to better understand the user context (temporarily leaving the office rather than leaving the building entirely). Devices on wireless networks may drop off when they sleep so be more intermittent.

An increasing number of phones support Bluetooth, so that if that term becomes widely used, it could be an additional source of data in buildings that have Bluetooth receivers. This would provide greater spatial resolution as Bluetooth is for shorter-range communication.

There has been extensive research on using signals from multiple Wi-Fi access points to determine the location of individual devices. In some cases it is the device itself that does the analysis; in other cases, data from the access points is collected and analyzed. Regardless, this is usually done for some functional purpose rather than assessing occupancy, so that it is worth adding hardware and/or software to the devices. The question here is what can be learned from devices and access points as they are.

Research Questions
How frequently do phones show up on wireless networks?

Supporting Application: Network Device Inventorying
For most of the methods above, it is desirable to have an automated way of determining what type of device is associated with each network address. These may be personal PCs, phones, tablets, printers, servers, cameras, or other devices. Those that do not indicate occupancy, or those that would lead to double counting, should be filtered out. There are methods (many proprietary) for an entity on a local network to send messages to other devices on the local network to gather information to identify what it is. These can be the types of protocols each device responds to, the organizational identifier of the MAC address, or content from some of the protocols they do respond to.

Such inventorying can be useful to make the virtual sensing data more useful, but can also be used for non-energy purposes, such as asset management. Such inventory can be gathered as often as desired completely automatically.

Research Questions

What are the capabilities and limitations of existing available software for inventorying devices over the network? How well can these map to a universal device classification system? Could simple, free, open-source software to do this be developed?

IT Network Traffic

Even when a device is present on the IP network, the user may still be not present. The actual IP data traffic emanating from end-use devices, or perhaps also going to them, will be different for systems that are actively in use than for those that are not active. For example, when a person is not present, we should not expect emails to be sent, or new web pages to be visited. There will be some traffic to and from a system not in use, such as network infrastructure traffic, hacker probes, software and security updates, data backups, remote user access, etc. However, there are differences between these two types of traffic that could be recognized. Using this method requires two elements: deciding on a traffic characteristic to use, and getting the data to an entity that can detect the signal.

The traffic characteristic may be a combination of indicators, and could include the volume of traffic, stability, types of protocols used, etc. It is unlikely that any examination of the data itself is needed, but only information about the traffic, or contained in packet headers. Some characteristics may be valid only for certain computing environments.

The other issue is what entity does the analysis and how it gets the data. They include:

- Analysis on the closest switch (or router) to each end-use device.
  This guarantees that all traffic is seen but involves a potentially large number of devices.

- Analysis on a central router.
  This will miss traffic local to subnets, but that traffic may not be necessary for the algorithm to properly function. For example, email traffic to a cloud server will go to that server even if the message is sent between two people on the same local network.

- Routing of all (or some) traffic from network equipment to a separate traffic analysis device.
  This is a capability that network equipment generally has (to copy traffic to a separate destination), but the large amount of data to send could be a concern.

- Sending select traffic characteristics from network equipment to a traffic analysis device.
  This could be as simple as the number of packets sent or received in a time period per IP address, so could be a relatively small amount of data.

- Monitoring traffic into and out of an entire building's network, e.g. across its broadband Internet connection.
  Many types of traffic (e.g. email from external servers and almost all web traffic) will cross this barrier and so this may be perfectly sufficient.

The analysis could be adaptive. In some cases it may be quite clear through simple methods that there is, or is not, a user using the computer. For example, if the amount of data or number of packets is below some threshold, then it may be clear that no one is using the machine for a network application. In other cases, if it unclear, then more sophisticated methods could be added. The latter are more complex but work for systems that are not powered down during off-hours.

Research Questions

How easy is it to obtain traffic characteristics from switches, routers, or other devices? Can the IPFIX protocol be used for this purpose? How feasible is it to route all traffic to an occupancy server, or some sufficient subset of traffic? What features of traffic best indicate occupancy, are most common, and most easily obtained?

Enterprise Applications

In many offices, there are one or more network applications that people log into when arriving at work, and log out of (for security and other reasons) when they leave. Examples are email servers, database systems, customer relationship management systems, etc. These could be used to indicate when a person is present or not. These enterprise application systems could be queried for user presence, on an individual or collective basis, or could report out the data automatically. In some cases people may be able to work remotely. The difference in this needs to be detectable. The IP address of the user should be different and so readily indicate this. Electronic calendar databases can show when people are expected to be working, and for meetings with an assigned room number, where they should be. This is most likely to be useful for estimating conference room occupancy. Some calendar data might not reliably differentiate between meetings that are on-site, or off-site.

Other IT Devices

Other sources of data are IT systems that are at least sometimes connected to the IP network, and even when not, may still be accessible electronically.

Building Access Control Systems

These systems, usually using RFID badges, provide tracking of the presence of individuals in a building. The use of the systems is varied. Some are to unlock doors and are intended for more than one person to enter on a single badge scan. Others require each individual to "badge in". Some are used on entry and exit, but many only on entry. Some are only used during non-business hours. Regardless, some useful information can likely be gleaned from these systems, particularly when calibrated against knowledge of ground truth occupancy.

Hardwired phones

Conventional phone systems can track incoming and outgoing calls by line. Many commercial buildings now use Voice over IP (VOIP) phones that have more intelligence and programmability than older technologies, and are inherently capable of IP connectivity. In principle, phone systems could be queried for the most recent time a phone was used. A table translating phone line to location is needed. It may be possible to report the physical port used, which will be more static than the phone number used. Some people use their phones frequently so that it is a reasonably good indicator of occupancy; for others the phone is only used occasionally, so that phone usage would be most useful for a building total estimate.

People may have other interactions with their phone useful for occupancy assessment, e.g. manually changing the outgoing message for missed calls from one for use during the day to one used in off-hours.

Cameras

An increasing number of buildings have cameras for security or other purposes. While early versions transmitted the data as analog signals via coaxial cable, modern systems usually use IP for ease of data transfer (and ease of powering, over Ethernet). The video signal could be relayed to the occupancy server for processing to count people going into and out of a space, commonly a building entrance. While identifying people is theoretically possible with this method, the disadvantages with trying to do that are substantial[6] so we do not further consider it here. For total building occupancy, some inaccuracy in counts is not a problem. One can often assume zero occupancy in the middle of the night and so reset the counter to zero so that errors do not accumulate. Most cameras will be at entrances/exits but a few may be on the occupied spaces themselves, requiring a different approach.

Research Questions

For most security cameras, what are typical values for resolution, frame rate, distance to people, etc.? Is this sufficient for counting people and their direction? How useful are infrared cameras compared to visible light cameras? How common are they?

Building Infrastructure devices

There are many parts of building infrastructure that don't usually communicate today, but perhaps will in future, or might get power meters installed to indicate their activity (and such meters would not need to be accurate for this purpose).

Elevators

Elevators provide only a coarse assessment of occupancy given that there may be other ways to transit between floors, and the number of people entering or exiting at a given floor may not be known (though getting weight information could show this). They also don't address occupancy of the building floor directly accessible from the outside.

Automatic doors

Power measurements of automatic doors could show the instances of opening, and how long they remain open for. Some doors are one-way (e.g. at supermarkets) which are even more useful for assessing occupancy. Some doors have a manual option that may be available all the time, or only some of the time.

Water heaters

Most hot water use is tied to direct occupancy, though as most occupancy activities don't involve hot water, it is primarily useful as a crude indicator in both time and type of occupancy.

---

[6] Identifying individuals is viewed by many people as an unwarranted invasion of privacy, and may be in conflict with laws, labor contracts, company policies, etc. Creating such data then creates a need to carefully guard it against inappropriate or unlawful access. In a retail environment, public knowledge of the practice might result in lost customers who object to being monitored this way.

Water use
Total water use in a building is a more direct indicator than only hot water. Some uses such as irrigation are independent of occupancy but these could be identified and filtered out.

Electricity metering
Whole building electricity can indicate occupancy so long as baseload and automatically controlled loads (e.g., HVAC) can be subtracted out (weekends and holidays can be particularly useful for this). Data from submeters can provide much better targeting of load types that indicate occupancy, and can be specific to particular parts of a building. It is critical to not create any unintended feedback loops, such as an automatic activity in a building being misinterpreted as occupancy, which then triggers one or more building services to initiate, which is then seen as further evidence of occupancy.

Existing occupancy data
Some building systems today have occupancy data they collect and use locally, but do not make it available to other building systems. Lighting controllers are an example. An occupancy server could query such systems, combine this with data from other sources, and then publish it to other building systems.

Tier II Methods
As defined in Christensen et al. (2014), Tier II of virtual sensing is methods that that require additional software on existing hardware, but no new hardware. This increases the burden of deploying virtual sensing, but can provide data and insight not otherwise available. They may be perceived as intrusive by building occupants, who would at a minimum need to be confident that the content of their actions are not being recorded or assessed, but only their presence. Sound-based methods could be confused by a loud sound emanating from other than the target room, e.g. a building public address announcement, or a nearby car backfire or plane passing overhead. However, an occupancy server could be set to filter out reports that occur in all or many rooms simultaneously. Some of the methods described in the Tier I section may be Tier II in some deployments, but could in principle be accomplished solely as a Tier I activity.

PC - input activity
For the many people whose work is dominated by interaction with their PC, use of the keyboard and/or pointing device (mouse or trackpad) is highly correlated with their presence. All that is needed is a small background application that observes such input activity and reports it on a periodic basis. This could be reporting on fixed periods or event-based. An event would be an input action after a long period of inactivity, or a long period of inactivity reaching a time threshold. Device power-up and power-down events should also be reported (though power-down can be unexpected and so missed). A PC that is powered down cannot report input activity, but of course is not being used anyways so this lack of capability is not a problem.

This method can be well combined with others such as tracking phone activity, and using the PC camera and/or microphone, to assess presence of times when input activity has ceased for a sufficient time period.

Fixed phones - microphone

The microphone on a fixed phone (e.g., a VOIP phone) could be always listening, or set to listen during times there is uncertainty about occupancy . Some sound patterns should be reliable indications of occupancy. Activities such as night-time cleaning might be differentiated from regular occupancy due to the timing (including of appearance in nearby offices) and nature of the sound. Sensing sound with the phone would be a way to control office lighting, since it can occur quickly after a person enters a space. Phones are typically on continuously making them particularly good for sensing when someone enters a space.

PC - camera and/or microphone

PCs with an integral or attached camera could take a picture any time there was uncertainty about occupancy, and have a simple algorithm to assess if the user of the PC (or really, anyone) was present or not. The same could be done with the microphone, providing a similar service as the fixed phone. A PC that is powered down cannot report such data, making it less suitable for assessing when occupancy begins for the day.

Displays

Some TVs, and more recently some computer monitors, have integral occupancy sensors. These are for the purpose of powering up and down the display, and are not generally (perhaps not ever) available externally. However, a software modification in these devices could enable the status of the occupancy sensor to be made available either by querying it, or by automatically pushing out changes to the occupancy sensor status. An increasing fraction of TVs sold today have Internet connectivity built in. There have been copiers sold with occupancy sensors (to automatically awaken them when a user approaches) but they were not seen as a successful feature and are rare.

Tier III Methods

Tier III of virtual sensing is methods that that require additional hardware as well as additional software. These work against the core appeal of low cost of virtual sensing, but may still have merit in some cases. As an example, a PC could have a USB-based occupancy sensor, with the PC providing both the power and communications for the sensor. The USB device could wake up the PC when occupancy was detected, providing user amenity.

An example from a different type of sensing is using a USB temperature sensor. This could avoid the need for dedicated communications for a temperature sensor. When the PC is powered down, it can’t provide real-time readings (it could relay them when the PC powers back up), but it is less important to have temperature data in unoccupied spaces than in occupied ones. A PC could be woken up occasionally to get a temperature reading if needed.

Some Wi-Fi access points have USB ports that might be able to accept sensors; these could be particularly useful in hallways that lack other potential sensing devices. For PCs without a camera or microphone, on could be added for purposes of virtual sensing (and then of course used for any purpose).

Summary

Many systems already existing in buildings today have, or could have, data that could indicate occupancy. A challenge is to extract this data, process it, and combine it with other sources. Many of these methods are likely to be imperfect so that a combination of several will produce a better result. Concrete examples for each method will help better understand the operations necessary to obtain them, features or limitations, and ways that they could be made easier to obtain or more accurate.

Appendix B: Examples of virtual sensing data streams

We identified several potential sources of data for Virtual Sensing that might be available for building 90 at LBNL. This building is advantageous for this effort in that the involved researchers work in the building so that it is convenient and well known, it is relatively large, so that privacy concerns are minimized, and has a dedicated portion of the LBNL IT network. The selected sources were available, did not raise privacy concerns, and appeared to be promising as indicators of occupancy. These include calls from the landline phones in each office, data from the Wi-Fi system, and data from the IT network generally.

While we focused on building 90, all of the data sources are readily available for the entire laboratory, with only modest additional effort needed to produce similarly occupancy patterns for many buildings.

 shows two initial examples of results from our data collection. The first graph shows outgoing phone call counts (not including intra-lab calls) by hour, for a week in June 2014. The second shows associations of devices (mostly mobile phones) to Wi-Fi access points in the same building, with counts per ten minute period (this only indicates presence of the devices on the network; it does not indicate whether or not the device is in active use). In the case of Wi-Fi, the presence of a mobile device is highly associated with the presence of the person. In the phone case, it is user activity that is being observed, not just presence. Activity relies on occupancy, but is one step removed, as a person can be present but not active.

The rest of this data shows examples of data streams we have collected. As these are exploratory analyses, some of the axes are labeled in seconds out of convenience. The daily patterns of operation are generally visible in the data so the x-axis not of importance.

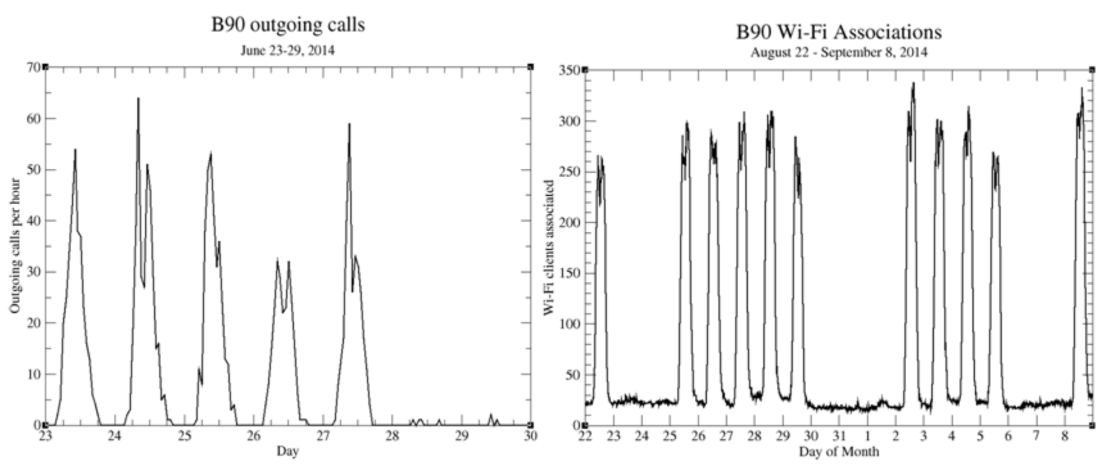


Figure B-1: Outgoing wired phone calls and WiFi associations for Building 90

Wired Phone Calls

LBNL tracks outgoing phone calls (that is, calls to locations outside the laboratory), for billing purposes. Not tracked are incoming calls, calls between lab phones, nor incomplete calls.

Thus when a phone call is logged, there is a reasonable guarantee that someone is in the space where the phone is. While the overall average daily calls per person is less than one (because during most hours most people don't make a call at all, even though some may make more than one), there is a reasonable association between call volume and building occupancy. The data we acquired also include the duration of each call, but we have not yet identified a good use for that for virtual sensing data.

Figure B-2 shows hourly counts of calls for a three-month period at B90, plus a one week subset of this. Lunchtime is visible on several days. Holidays (in the left figure Memorial Day and the Fourth of July) show up just like a weekend day, with a maximum of two or three calls in an hour. There is a fair amount of randomness in the data. A threshold of perhaps 5 or 10 calls/hour could be set as the indicator of when the building is occupied. This would allow identifying the beginning and end times of substantial occupancy.

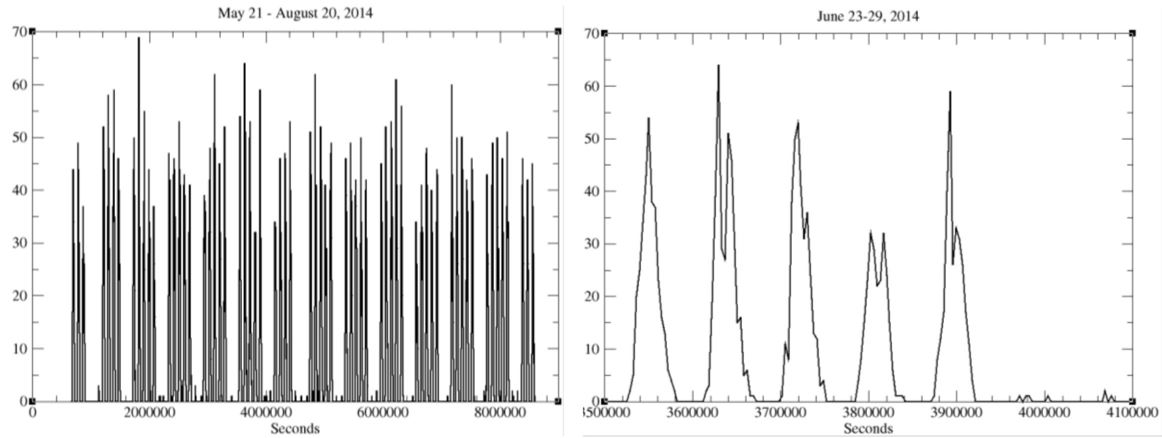


Figure B-2: Outgoing wired phone calls for Building 90

Figure B-3 shows call data for B46A, for periods of time of more than a year, and about three weeks respectively. There are far fewer occupants of this building (according to the lab phonebook, 17 versus the 357 in B90); this makes the resulting load shape much coarser and less reliable. That said, overall patterns such as weekends, holidays, and daylight saving time, all appear as expected.

Wi-Fi

In commercial buildings, it is common to have a Wi-Fi system composed of many access points (APs) that bring the Wi-Fi data to a wired Ethernet port for further routing, plus a facility-wide AP Controller that manages security and moving connections among access points when the connected device changes location.

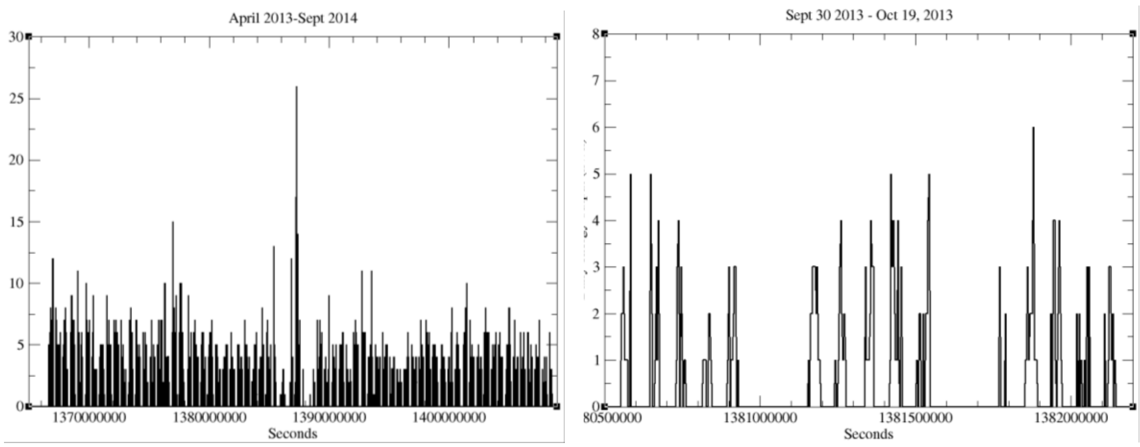


Figure B-3: Outgoing wired phone calls for Building 46A

Building 90 has 32 APs spread over the four main floors plus the basement. With about 400 occupants of the building, this is an average of about one AP per twelve people. The lab-wide AP controller can be queried any time for a list of devices connected to each AP. The LBNL network staff set up a system to do this every 10 minutes. These data include many aspects of the device and connection, but notably the IP and MAC address. To not risk intruding on privacy, we only obtained a count of the number of devices per AP, not any data about each individual device. There are actually two figures of merit potentially available. One is the number of devices that have actively connected to the Wi-Fi network or “authenticated”. The other is just devices in the area that have made no attempt (or made a failed attempt to login) to connect, but are known to the AP to be in the area by virtue of how Wi-Fi works. We think the LBNL data is associated, but data from UC Berkeley where both are available show they are not significantly different.

Figure B-4 shows 18 days of B90 Wi-Fi counts. The Labor Day holiday is clearly visible in the middle of the graph, as is the lunchtime dip. Having a 10-minute resolution makes such fine-grained patterns such as lunch more apparent. People might have zero, one, or two Wi-Fi devices (two if they have both a phone and a notebook using Wi-Fi). It is likely that on any given day a significant number of B90 occupants are never present, or present for only a portion of the day, so that the peak values shown are probably similar to a ratio of one device per actual occupant.

There is a quite consistent background level of about 23 associated devices. This could include some desktop PCs using Wi-Fi because Ethernet was either unavailable or inconvenient. It could also include notebook PCs that are left fully on. Finally, there could be printers or other devices in the buildings that are always on the network. For purposes of occupancy detection, this base load can simply be subtracted from the current value.

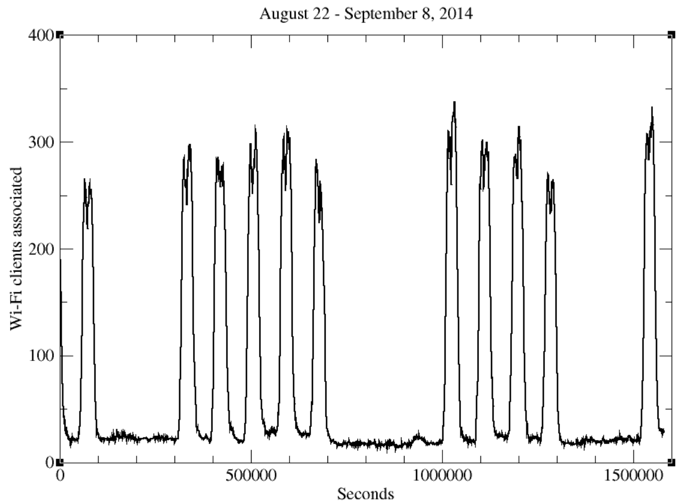


Figure B-4: WiFi associations for Building 90 over 18 days

The AP control system that covers LBNL B90 (from Aruba) reports the Device Type, if known. Queries for web pages often include data about the hardware and operating system of the device; this can help the web server that creates the page for the device to know what the screen size and capabilities of the device are. The AP can observe this data to identify many devices. Another line of inquiry is to separate the data by floor, to see how similar or different occupancy patterns are within the four building floors.

In more exploratory work, data on the time of associations to APs for one of the author's phones was obtained for many months. This showed it moving around B90, as well as occasionally at other lab buildings. It also showed up sometimes at the AP for a building on the vehicle route up to B90, which also has a major shuttle bus stop. However, since the snapshots were only every 10 minutes, many short associations to that AP were not captured. This ability to track individual behavior is both promising as well as raising significant privacy concerns.

One issue for this source of data is that devices stay in the list of associated devices for about 30 minutes after they have been last seen. To some degree this is needed as one doesn't want a device to be knocked off if it loses contact for a few seconds as a person moves around. That said, providing a count that well-reflects occupancy would likely benefit by using a shorter time period. Experimentation with this would be helpful.

Address Resolution Protocol (ARP)

One of the most basic protocols on an IP network is ARP (Address Resolution Protocol). It is used by a device to announce to the network infrastructure that it exists, and where it is located on the network. LBNL's routers (as all routers do) maintain a table of "current" machines that have assigned addresses. Data for these is published on the LBNL network once per hour, and for this project we set up a system to archive this data for later analysis. Figure B-5 shows about one week of ARP count data for B90. This reflects only the wired network in the building (the wireless devices are mixed in with wireless devices from throughout the lab for ARP purposes). The system keeps devices in the list for about four hours after they are last actually seen on the network. This distorts the data making 8 hours of presence on the network appear as 12. As we do have the identity for each device in each hourly list, it is possible to correct the data to remove these extra hours, to get a more accurate shape, though this can only be done retrospectively, not in real time.

There is a high base load in these data, which reflects PCs left on continuously, servers (of which there are a modest number in the building), and printers. The baseload can be simply subtracted from the total for the estimate of human occupancy. Note that the weekend base load is lower than it is for weeknights, reflecting some systems being powered down only on weekends. Finally, there are some PCs (principally from Apple) which can respond to ARP packets even while asleep, if configured that way. This only became available in recent years. Thus, the base load overestimates the number of PCs on continuously, though by how much is unknown.

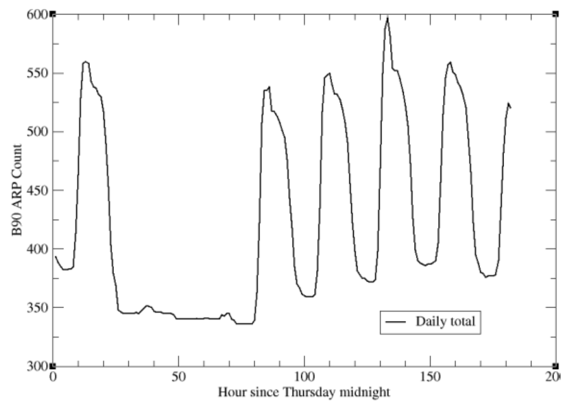


Figure B-5: ARP data for LBNL Building 90 wired network, single week in late August 2014

DHCP

The Dynamic Host Configuration Protocol (DHCP) is used in an IP network to allocate (potentially) scarce IP addresses to devices on demand, rather than in advance and semi-permanently. Figure B-6 shows the overall count of communications between devices in B90 and the lab's DHCP server. There are several communications to initiate a DHCP lease, and several more in each transaction to renew it, which occurs at LBNL every two hours (different networks can establish different lease times; LBNL's are for four hours, which means renewal happens at two, at half the lease time). In any case, the multiplicity of communications per device is why the total is so much larger than the number of devices as reported by ARP. The data could be processed to eliminate duplicates and obtain more accurate hourly data. As with ARP, there is a high base load value, for the same reasons. That the base load shifts markedly several times is curious, but not relevant to occupancy detection since the night base load is simply subtracted from the total.

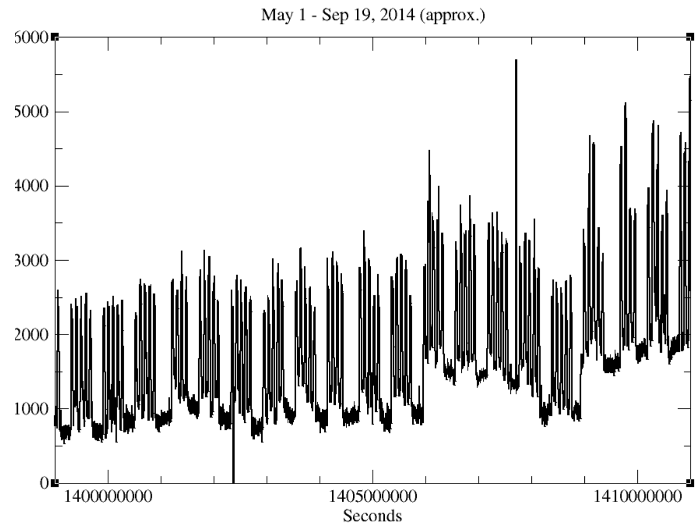


Figure B-6: DHCP transactions for Building 90 over several months

Hypertext Transfer Protocol (HTTP)

The Hypertext Transfer Protocol (HTTP) is used to issue requests for web pages, and for responses to deliver the desired content. While its major use is with web browsers and web servers, any application can use it if it wants to. Figure B-7 shows hourly counts of HTTP requests from a single machine. For the first five days the employee was away but the machine was on. Some application(s), or the OS, were using HTTP while no one was there--for several days at a consistent rate of about 50 per hour. On return, the employee put the machine to sleep at the end of each day, which is why there are zero HTTP requests nights and weekends. On the last evening of data reported here, the machine was left on overnight once again.

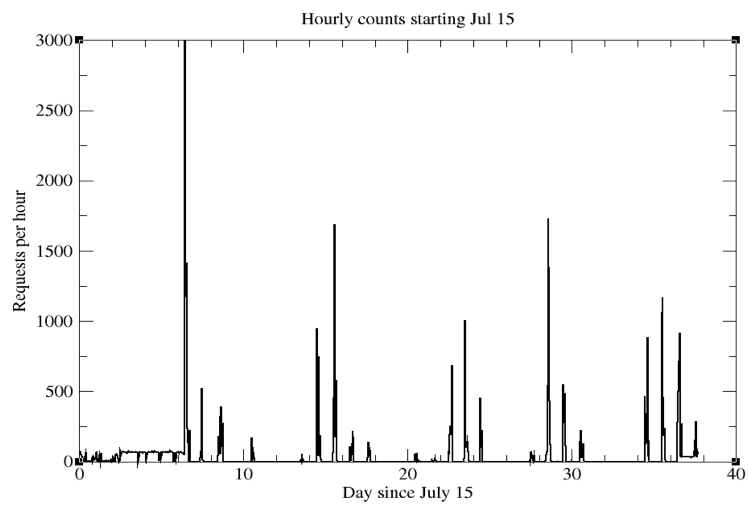


Figure B-7: HTTP requests for a single PC over 5 weeks

While there is substantial variation from hour-to-hour for this machine, it is possible that the total of several hundred machines would average-out these variations and be a reasonable proxy for occupancy. Or, if the data were accumulated per-IP address, even if the address itself was not included, an algorithm could determine present-or-not for each device, then accumulate these to get a total for the building. The number of HTTP requests from a machine that is on and in use is not important - just the fact that it IS on and in use. Such fine-grained analysis should produce a significantly higher quality result. There is probably additional data in the HTTP logs that could help identify the type of device (e.g. PC vs. server vs. printer) to further improve the result, and the address being queried might indicate if it is associated with human occupancy or not.

DNS

The Domain Name Service (DNS) is the mechanism by which human-readable ("domain name") addresses (e.g. www.energy.gov) get converted to numeric IP addresses (e.g. 199.167.76.13). Facilities such as LBNL operate their own DNS servers for efficiency and security (they acquire data from external DNS servers as needed).

Figure B-8 shows DHCP counts for a single machine—the same on as used in the HTTP graph above. It has the same characteristics as the HTTP source, though with less erratic results. The same processing could be done on the entire DNS stream for B90 to isolate individual systems. If the name of the system being asked for was also known, there might be ones that could be readily filtered out, such as those for sites that do only or principally back-up services.

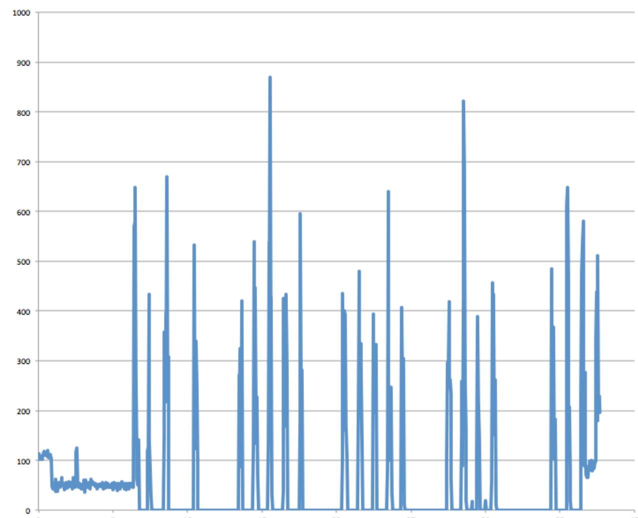


Figure B-8: Counts of DNS requests for a single machine in LBNL Bulidng 90 for over 5 weeks

Additional Research Needs

This collection and analysis process could be extended to more types of information, and many of the ones that we have collected could be studied in more depth. For example, knowing the type of device on a network rather than just the count can provide a better picture of occupancy. This is particularly true for Wi-Fi.

In addition, we need to do cross-correlations between the various sources, to see how consistent or not they are, to explore ways to adjust each source to make it more reliable. Some, such as HTTP transactions and DNS queries have a direct relation to each other. Others, such as wired phone use, are independent. We can then make recommendations on the value of each source, in isolation, and incrementally when some other sources are already at hand.

The data we have all are available for all of LBNL. Thus, many buildings could receive the same sort of analysis. In some cases, small buildings are combined with others for their wired network, so that those data streams can only be used to assess the building collection as a whole[7].

There are other methods of data collection we have not employed to date but which may have promise. One is to study the actual data packets that are sent to and from a PC when it is on and being actively used, on and not being actively used during the day, on and not being actively used nights and weekends, and when asleep. Each of these may have signatures that could be observed. Another is to actively interact with machines on the local network. Printers, for example, will respond so some protocols that other devices will not likely do, or respond differently.

The origin of virtual sensing is the realization that status on the IT network provides information about the energy use of PCs and other electronic devices. It would be helpful to determine what percentage of electronics and miscellaneous energy use in buildings. This involves inferring what type of device is present on the network (to get average power levels by mode) and then determining the operating pattern for each and doing the arithmetic.

---

[7] This may overstate the case. It may be that querying the Ethernet switches that devices are connected to could reveal the MAC (or IP) addresses of the connected devices, and then applying knowledge of which switch ports go to which buildings, disaggregate the data by building. Or, even the on/off status of each port might be able to be matched with the appearance and disappearance of devices on the network, to enable inferring which device is connected to which port, to get the same result.

Appendix C: Additional Virtual Sensing Issues

This appendix covers several aspects of virtual sensing that do not fit neatly into the main report or the previous two appendices.

Ground Truth

Two key questions remain: (1) How do we measure exactly how many people are occupying a building in a general sense, and how many people are actually in the building at any given time. The first question has many potential answers. For building 90, there are 357 wired phones; the lab's facilities department considers that there are 446 people assigned space; and there are about 530 offices and work stations. Some spaces have no one occupying them. Some people have no wired phone. Some, such as custodians, have no assigned workstation. Some people work part-time and may, or may not, share a space. Some are on extended vacations or work travels. Some people come to meetings and so are in the building for a time without having any permanent association. Luckily for our analysis, we are aware of no people who are regularly in the building in the middle of the night.

The second question is how many people are in the building at a single given time, to be able to compare that figure to the one provided by the virtual sensing system. Such 'ground truth' data for a period of time (even a few hours during a morning or evening) can both determine a conversion ratio from the value that virtual sensing provides to actual occupancy, and validates that such a ratio is stable over the course of a day. This needs to be done for a few buildings to validate virtual sensing in general, and to provide reference values for different building types. Developing inexpensive ways to collect episodic ground-truth data would be helpful. That said, for many purposes, it may not matter how accurate on an absolute basis, or so consistent over the course of the day, implicit sensing data are, as even rough estimates may get most of the benefit of moving from no data to perfect data.

Privacy and Security

An issue which quickly arises as one delves into virtual sensing is the potential conflict between obtaining data useful for energy and perceived and real concerns with privacy and security. At its extreme, virtual sensing is about using our IT infrastructure to spy on people—to track and record their movements, activities, and electronic fingerprints. This could exceed bounds on what is allowable by policy or law, or be quite unsettling to people who work in places where it is used. That said, the goals of virtual sensing in general do not require the most sensitive data, so it is critical to delineate what is, and is not, being collected, be transparent about this, and seek to build systems wherein sensitive data are filtered out before being passed on. In this project, we consulted extensively with LBNL's staff in charge of privacy and security, to assure that our activities were well short of the line at which concern could be legitimately raised for either of these topics. Research in this area does require some exploration of sensitive data, such as to confirm that masked or aggregated data reasonably reflect what the detailed data would indicate. Crossing the line to a limited and controlled degree can help establish where the line of limit should be for technologies.

For many purposes it isn't important to know who is in the building, but only how many people. In these cases, counts of devices are sufficient, and individually identifying information can be filtered out. In other cases, it may be necessary to be more specific. For example, the telephone of someone appearing on a campus Wi-Fi AP likely means that that person will soon appear in their office. This could be used to adjust HVAC, lighting, and electronics operation. However, it does require the Virtual Sensing system to have a persistent association between the device and the owner, which then allows the system to track the owner around the building and campus.

More Analysis of Building 90

LBNL's Building 90 has been the prime focus of our investigation into virtual sensing. This section provides additional insight and analysis. As background, B90 has four main floors (the fourth floor has a smaller footprint than the other three), basement area, two "temporary" buildings in front (90C and 90P), and the FLEXLAB buildings (90X). The discussion here does not include the temporary buildings or FLEXLAB, but does include the basement; the FLEXLAB wired network is separate from the rest of building 90 and at the time of this data collection at least had no Wi-Fi infrastructure.

B90 has 32 Wi-Fi access points (APs) in it, scattered across the five floors. All of the data below reflects totals from all of them, but additional analysis could look at data from individual APs or groups of them. Every ten minutes, a snapshot is taken of the devices associated with each AP; these are mostly, but not entirely, phones and notebook PCs. From what we know about the actual population of the building, the Wi-Fi total is similar to actual occupancy, suggesting that there is rough equivalency between the number of people with no devices vs. the number with two.

Figure C-1 shows a count per ten-minute period of the number of associated Wi-Fi devices. There is a baseload of about 22 in the middle of the night that likely reflects PCs fully on. As we believe that normally there is no one in the building during this time, these should be discounted at all times. The graph shows a reasonable pattern of arrivals in the morning and departures in the evening, along with a dip at lunch presumably showing people leaving to eat. Some of the evening tail is likely due to PCs that are on for some time period after the user has left, before going to sleep.

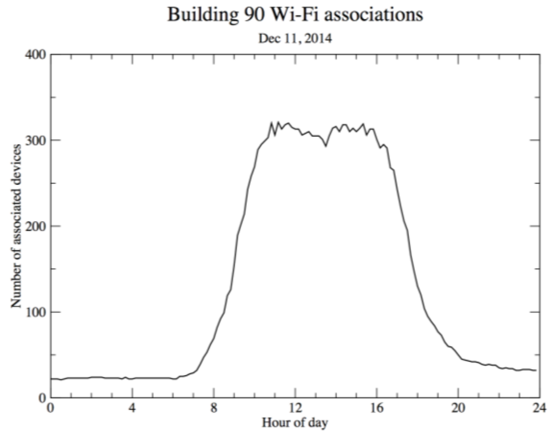


Figure C-1: WiFi associations for LBNL Building 90

Figure C-2 shows 18 days of data, including one week with a holiday on a Monday. That day is almost the same as a weekend day, with just a few extra devices seen. The peak does vary, with Friday's notably lower; this is towards the end of summer so not surprising.

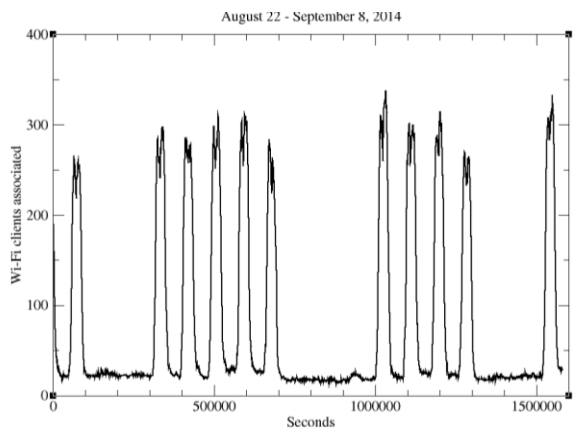


Figure C-2: WiFi associations in Building 90 over eighteen days

Figure C-3 shows a different sample day, with the building's HVAC schedule superimposed. From 8am to 6pm, the ventilation and temperature control are full-on. There is an optimal-start algorithm that begins earlier than 8am to bring the building to the intended range by 8am.

To understand broader patterns, it is necessary to apply some statistical analysis to the data. This will also be needed when longer time periods and more buildings are assessed. If we ignore the lunchtime dips, considering daytime occupancy as relatively flat, then the issues are what is the peak occupancy level, what is the nature of transition from unoccupied to peak, and the nature of the reverse transition at the end of the day.

Figure C-4 shows two Load Duration Curves of data in December and January that cover ordinary workdays, workdays impacted by the holiday break, holidays, and weekends. The black line shows all ten-minute time periods, sorted from highest to lowest. There are 144 periods per day. The red line takes only the peak period for each day. The lowest red days all are during the holiday break (a few PCs likely powered down making them lower than ordinary weekends). The next five days are ordinary weekend days. The highest two days in that section are Monday and Tuesday between Christmas and New Years, so forced vacation days, but showing a few people coming by. The days between the two roughly horizontal sections include the actual workdays during the two-week holiday period; many people take these as vacation days (one day is erroneous and reflects a part-day at the beginning of the analysis period). The lowest ordinary day in this time period has a peak of 284.

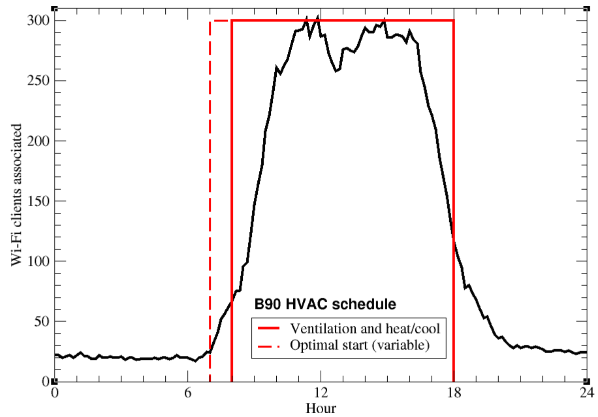


Figure C-3: Outgoing wired phone calls and WiFi associations for LBNL Building 90

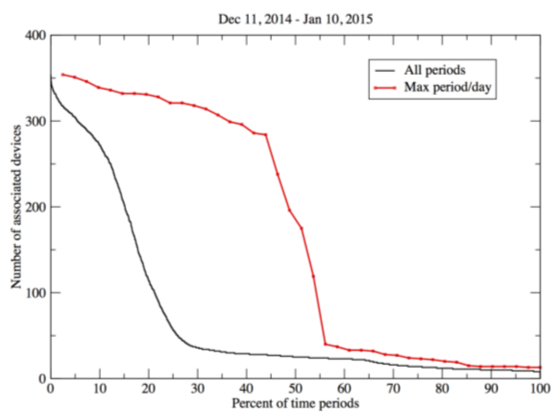


Figure C-4: Load duration curve of WiFi associations for LBNL Building 90

We want to identify statistical periods for each day and want to assure that the fully-occupied state is reached during each regular workday. For the analysis below, 280 was taken as the peak.

The points we chose to analyze are those that are 10%, 25%, 50%, 75%, and 90% of the way between the night-time base and the value chosen for peak. We found the first time of the day that each of these percentile values was exceeded, as well as the last time during each day it was exceeded. A graph of these values is shown in Figure C-5. It is notable that there is a significant group of people who arrive promptly at 8am each day, with never more than 10% arriving before that time and most of the time over 25% there before 8:10; this is shown by the red line being on top of the black line for arrival for most days in the graph. Leaving times are neither so regular nor so closely clustered. Figure C-5 includes two holiday weeks with much lower occupancy

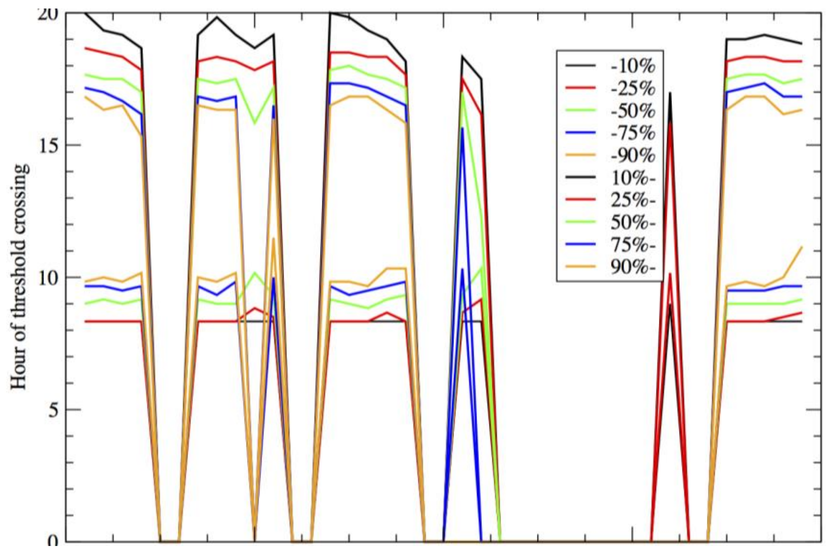


Figure C-5: Percentile times for Wi-Fi associations for LBNL Building 90

We sought to test the phone data against the Wi-Fi data, so for a period of four weeks plotted these against each other with the result in Figure C-6. If the two were perfectly correlated, points would fall along a line sloping up to the right. However, it moves clockwise, with many more phone calls proportionately in the morning than in the afternoon.

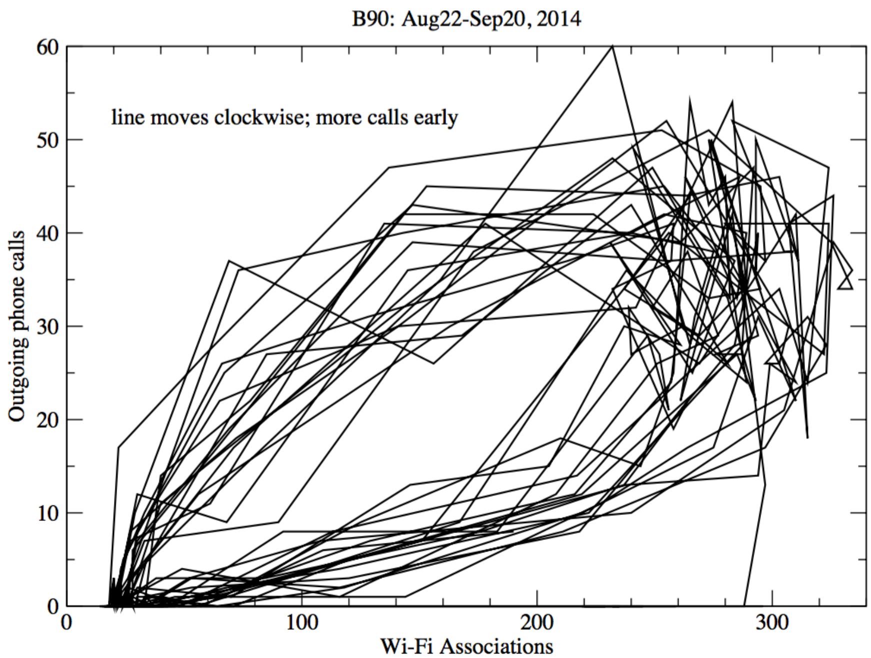


Figure C-6. Outgoing wired phone calls vs. Wi-Fi associations for Building 90

Figure C-7 shows the ratio of phone calls to Wi-Fi associations for the same time period. The ratio drops dramatically over the course of the day. The average ratio value could be used to adjust phone data to provide a much better estimate of occupancy, but this exercise points out just how unreliable phone data are in comparison.

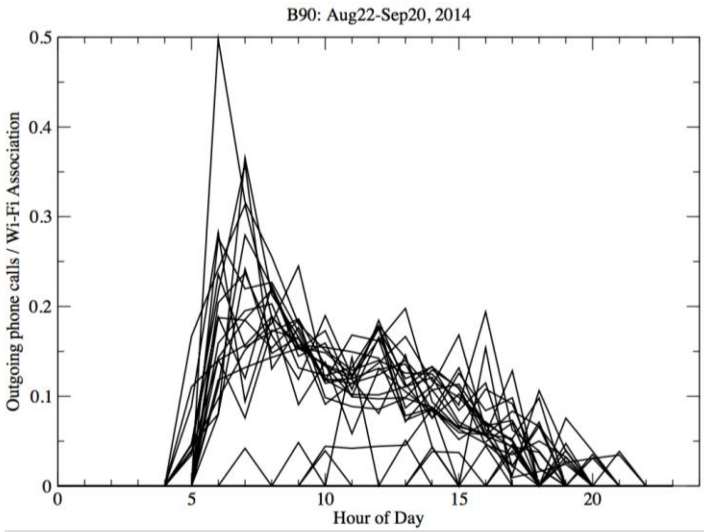


Figure C-7: Ratio of outgoing wired phone calls to WiFi associations for LBNL Building 90

Ideally, the relationship between each virtual sensing variable and the actual building occupancy would be known. To determine these relationships it is necessary to have "ground truth" data—the actual occupancy as a function of time of the buildings (or portions of buildings) that we are studying. We are exploring ways to obtain actual occupant counts for some buildings at some times, for direct comparison to our virtual sources. Figure C-8 shows image captures from a camera set up at LBNL to record construction of the FlexLab. While not a goal of this camera, it does show most of the main building entry. While there are several other doors so this is not sufficient to estimate occupancy by itself, it provides another potential example of virtual sensing.

Figure C-8: Images captured from FLEXLAB cameras near LBNL Building 90

Appendix D: Statistical model documentation and code documentation.

This appendix discusses the statistical model that is used; provides explanations for key choices made in defining the model; and gives documentation on the software implementation of the model.

Overview

The statistical model generates a baseline prediction that is a linear combination of separate predictions, each of which uses ordinary linear regression. Each linear regression prediction generates a predicted baseline. Input variables may include outdoor air temperature and/or one or more additional variables in addition to load as a function of time.

Outdoor air temperature data are handled differently from additional variables: within each of several operating mode categories (described below) the model assumes a piecewise-linear relationship between load and outdoor air temperature, with possibly different slopes in several user-specified temperature ranges. For example, there may be one slope for temperatures below 50F, another for temperatures between 50 and 60F, another from 60F to 75F, and still another for temperatures above 75F.

Additional explanatory variables are also assumed to have a piecewise-linear relationship with load, but with only two different slopes: one below a user-specified quartile or quintile [?] of the variable and one above. For instance, the number of active Wi-Fi connections can be used as an explanatory variable – which makes sense because it is likely to be highly correlated with the number of occupants in the building – and different slopes may be estimated for the number of connections being less than or greater than its 20th percentile.

Each linear regression prediction assigns different statistical weights to the individual interval data points. One regression gives relatively high weight to data near the beginning of the dataset. One gives relatively high weigh to data near the end of the dataset. Depending on the length of the dataset and on the choice of model parameters, additional regressions may be generated, each giving high weight to a different part of the data. The final baseline prediction at a given time is the weighted average of the individual regression predictions; see below for details.

We first discuss the statistical model in more detail, and then move onto the software implementation.

Details

1. The model is fit to “interval” data that are reported at regular time intervals, typically 15 minutes or 1 hour.

2. “Time of week” indicator variables are used to capture the recurring weekly pattern. This may be implemented by constructing a matrix in which each row represents a data point and each column represents a different time of the week. For instance, for hourly data, the matrix will have one row per data point, and 7 x 24 = 168 columns. If the nth data point was

collected on a Sunday at 1:00, the (n,1) element of the time of week matrix will contain a 1, and all other elements on that row will be zero.

3. If outdoor air temperature data are provided, the data are used in conjunction with load data to divide times of the week into categories:
    a. Times when the electric load is least sensitive to outdoor air temperature. During these times we assume the building is in “unoccupied mode.” This determination is made as follows. First, the outdoor air temperature vector is processed to create a matrix for fitting a piecewise-linear relationship between temperature and load, with different slopes for temperatures below 50F, between 50F and 60F, and above 60F. This is done by creating a three-column temperature matrix. The first column contains max(T, 50), where T is the outdoor temperature in degrees Fahrenheit. The second column contains max(0,min(10, T-50)); that is, it contains zero for temperatures below 50F, 10 for temperatures above 60F, and T-50 for temperatures between 50 and 60F. The third column contains min(0,T-60). So: a temperature of 45F will be encoded in the matrix as a row containing (45, 0, 0); a temperature of 55 F will be encoded as (50, 5, 0); and a temperature of 65 F will be encoded as (50, 10, 5). Linear regression is used to predict load from the temperature matrix (with no intercept term), and the difference between load and the predicted load is calculated for each point. Times of the week when the difference is negative at least 60% of the time are assigned to “unoccupied mode;” other times of the week are assigned to one of the two “occupied modes” as discussed below.
    b. Times when the electric load is more sensitive to outdoor air temperature are assumed to correspond to “occupied mode.” Applying the rule described above will sometimes create gaps: for example, Tuesday at 6:00, 7:00, and 9:00 will be assigned to “occupied” mode, but 8:00 will not. Such circumstances are probably rare in real-world building scheduling, so if a short “unoccupied” gap appears between occupied periods, we redefine the gap period as “occupied.” Then the times of “occupied load” are further divided into two categories:
        i. “Startup”, which is the first portion of occupied mode during the day, defined as the first several occupied hours in each calendar day.
        ii. The rest of the period of occupied mode is the “non-startup occupied period.”

4. If outdoor air temperature is provided, we create three separate “temperature matrices,” one for each operating mode defined above. Each matrix has a row for each time in the dataset, and multiple columns. If the nth time period is in, say, the “startup” category, then the nth column will contain zero for the “unoccupied” and “non-startup occupied” matrices, and will contain non-zero values only for the “startup” matrix. Values in the matrix are filled in with the same scheme defined above for determining the occupied vs unoccupied modes, except that more temperature ranges are used. (By default, they are < 40 F, 40 -55 F, 55 – 65 F, 65 – 75 F, 75 – 90 F, and > 90 F). If the default temperature bins would lead to a bin containing fewer than 10 data points, the bin boundary between the problematic bin and the bin below it is removed. For instance, if there are only a few data points above 90 F, then the bins for 75 – 90 F and > 90 F will be combined into one bin for > 75 F. When used as predictive variables in a linear regression model, these matrices fit piecewise-linear relationships between outdoor air temperature and electric load, with a different relationship in each of the three time categories.

5. If additional predictive variables are provided, the following procedure is followed:
    a. Interpolate if necessary to put them on the same time intervals as the load data.

b. If only on predictive variable is provided, and it is missing some data points, then impute missing data by using predictions from a “time of week” model (equivalent to imputing the missing data on, say, a Tuesday at 14:00 to be equal to the average of all of the non-missing data from Tuesdays at 14:00).
c. If more than one variable is provided, then (if necessary) impute missing values of the first variable from a linear model that includes both the time of week and the second variable as predictive variables. Then (if necessary) impute missing values of the second variable, and of subsequent variables, if any, from a linear model based on the first variable.
d. Create two matrices based on the predictive variables. For each matrix, the number of rows is equal to the number of data points, and the number of columns is the number of variables. For simplicity, suppose there is just a single predictive variable, z. Let z(n) be the value of z for the nth data point. For the first matrix, the nth row of the first (and only) column contains min(z(n), s) where the value of s is a user-specified quintile [?} of the z vector. For example, if the user chooses a quantile of 0.2 – the 20^{th} percentile – then the s is the 20^{th} percentile of the z values. For the second matrix, the nth row of the first (and only) column contains min(0,z(n)-s). If there is more than one predictive variable, a column is assigned to each variable and the procedure described above is applied to each column. When used in a linear regression model, these matrices fit a piecewise-linear relationship between the predictive variable and the electric load. The anticipated “use case” for these variables is to use data that are linearly related to building occupancy. The reason for allowing a different slope for data above and below a specified quantile is to accommodate the possibility that the first people to arrive in a building have a larger (or at least different) effect per person compared to later arrivals. For instance, whether there is one person in an office or five people, the office lights are likely to be on.

6. Finally, the outputs of the data processing steps above are put to use: the model predicts the baseline load from a linear regression of the observed load on the time of week indicator variables, temperature matrices (if provided), and other predictive variable matrices (if provided). Several or many regressions are performed, using a different set of statistical weights for the data points in each case, in order to allow the final predictions to adjust for changes in building energy behavior. The rationale and approach to this adjustment are discussed immediately below, in a section that is taken (with minor modification) from Piette et al., “Automated Measurement and Signaling Systems for the Transactional Network,” Lawrence Berkeley National Laboratory Report LBNL-6611E, 2013.

Statistical weights

The underlying statistical model accounts for weekly periodicity in load, for changes in load that are correlated with changes in outdoor air temperature, and for changes in load that are correlated with changes in building occupancy (as measured or estimated by predictive variables that are provided). But in most buildings there sources of load that are not accounted for by the input variables. For example, changes in nighttime lighting might lead to an increase or decrease in the load at night, so that the pattern of electricity consumption is different after the change was made than it was before. To adjust for this sort of change in behavior, in order to predict the load shape on a given day, we give more statistical weight to days that are nearby in time, whether before or after the given day. This is achieved by fitting the regression model using statistical weights that fall off as a function of time in both directions from a central day. A central time point is selected as discussed below, and the time difference between that point and every other data point is determined (in days, which may be fractional). The statistical weight, w, given to a point d days from the central time point is:

$$w(d) = \frac{D^2}{(D^2 + d^2)}$$

where D is a user-selected parameter defined by the weighting days argument.

Figure D-1 shows how this weighting function varies for different values of D.

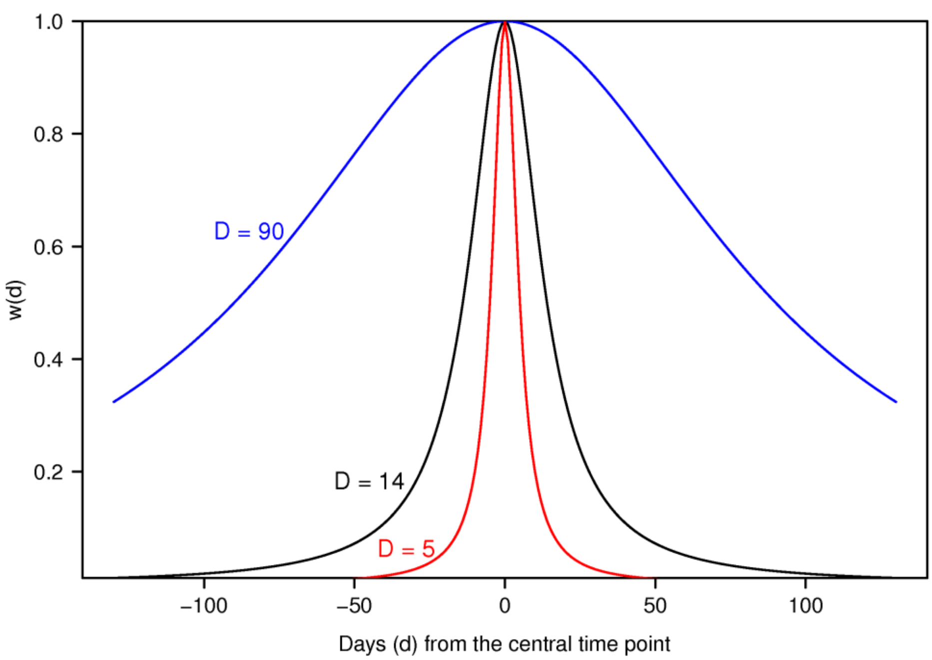


Figure D-1. Weighting function for different choices of D, the metric by which "short term" is measured.

The parameter D can be thought of as a "sensitivity" parameter that determines how closely the baseline model tries to match short-term fluctuations in the load data, versus capturing long-term trends. Setting a large value for D (such as 90 days) implies that data from three months ago are almost as informative about tomorrow's energy consumption as data from one week ago; setting a small value (such as 5 days) implies that data from two or three weeks ago are almost useless in predicting tomorrow's energy consumption. Empirically D=14 days is a good choice when predicting the short term load variation for several buildings we have studied, so we set it as the default value while allowing the user to change it if needed. Buildings that vary greatly from week to week would be better modeled with a smaller value of D, while buildings that are extremely consistent would be better modeled with a larger value of D.

To train the predictive model over a given time period, the weighted regression procedure is repeated for several different "central time points." Specifically, a set of "central time points" about D days apart is selected, spanning the time range of the data and the regression model is fit multiple times (e.g., using D=14, there would be 28 regression models generated in one complete year of training data), using each of these in turn as the “central time point.” Each of these models is used to make a prediction for each of the requested output times, resulting in a set of predictions for each output time: one prediction for each regression model. For a given output time, some of these predictions are from models in which the central time point was far from the output time, and some are from when the central time point was close to the output time. The predictions are weighted, using the same w(d) function above, to give more statistical weight to the predictions from “nearby” central time points.

The process for combining the individual regression predictions to generate the final prediction is illustrated in Figure D-2. The upper panel of the figure shows the final baseline prediction in blue. At any given time, the final prediction is the weighted sum of several different predictions, three of which are shown in the lower panels of the plot. For example, consider a point 11 days after the end of the training period. Since the first regression model has a central time point 0 days before the end of the training period, it is the most strongly weighted model at the point being predicted (see the red line on the second panel of the figure). The second regression model has a central time point 14 days earlier, so it has a lower weight (third panel). The third regression model has a central time point even farther in the past, and thus an even lower weight (final panel).

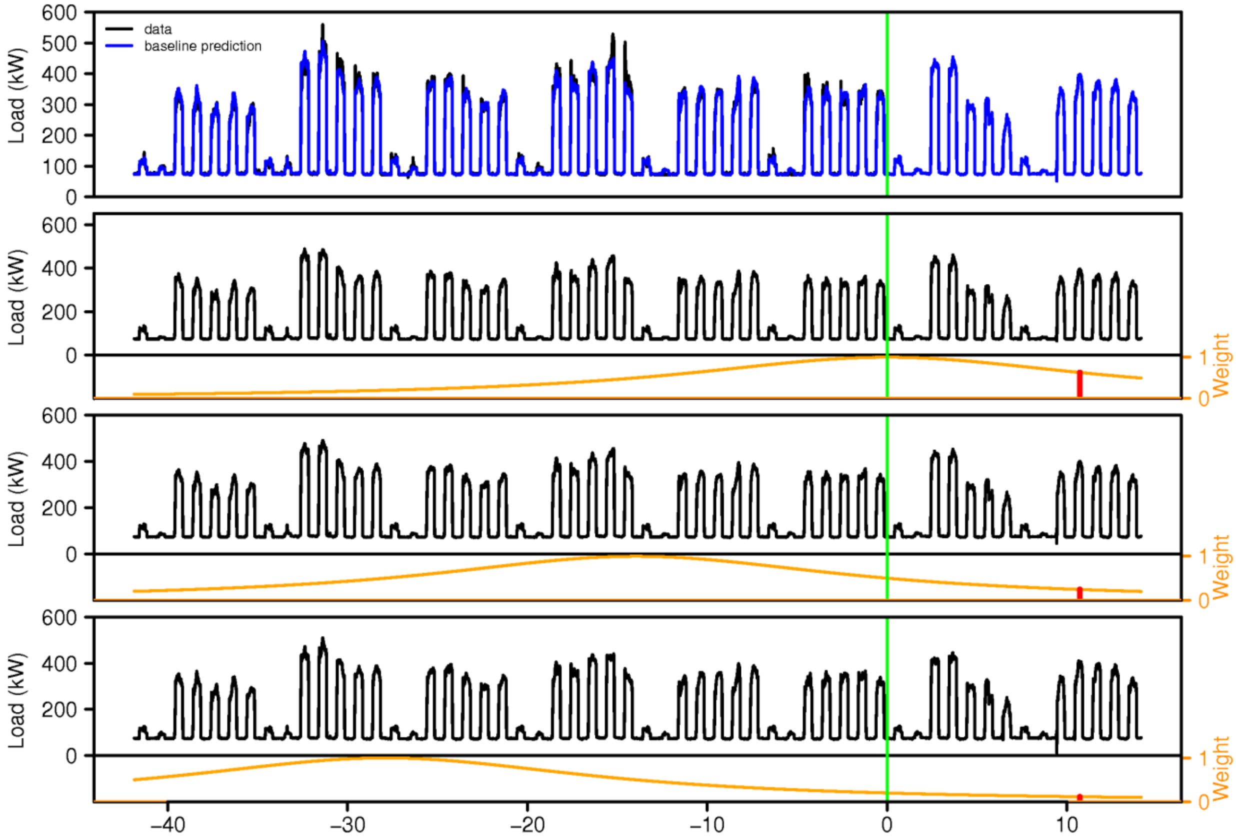


Figure D-2. Illustration of different weighting functions for statistical model.

Top panel:
Data (black) go from the left side of the plot up to the green line. The baseline prediction (blue) goes all the way across the plot;. To the right of the blue line the baseline prediction is a forecast, i.e. we have no data from the green line forward.

Second, third, and fourth panels show (1) linear regression predictions with central time points 0, 14, and 28 days before the end of the data, respectively, and (2) the weight function used for each prediction.

The weighting function w(d) has the effect that a prediction for a time less than D days after the end of the training data will be based mostly on the data from near the end of the training period, but a prediction for a time more than D days after the end of the training period will be based on a more equal weighting of the training period. As an example, consider using the parameter value D = 14 days with data from all of 2013 to predict the baseline from January 1, 2014 to July 1, 2014. The prediction on January 1, 2014 is the weighted sum of regression predictions that are fit to the 2013 training data using different central time points, as previously discussed. Since one of these central time points (December 31) is just one day away from the start of the time for which a baseline will be generated (January 1), that regression has a weight of over 0.99 at the start of the baseline. A previous regression, with a central time point about 14 days earlier, has a weight under 0.5 on January 1. A regression with a central time point an additional 14 days earlier has a weight under 0.2, and the regression centered in mid-November has a weight under 0.1, and so on back through time. In this case, training data prior to

November have weights so low that they are essentially negligible. Therefore, the prediction for January 1, 2014 is based almost entirely on data from December 2013, with data from November playing a minor role and data from the rest of 2013 having a nearly negligible effect.

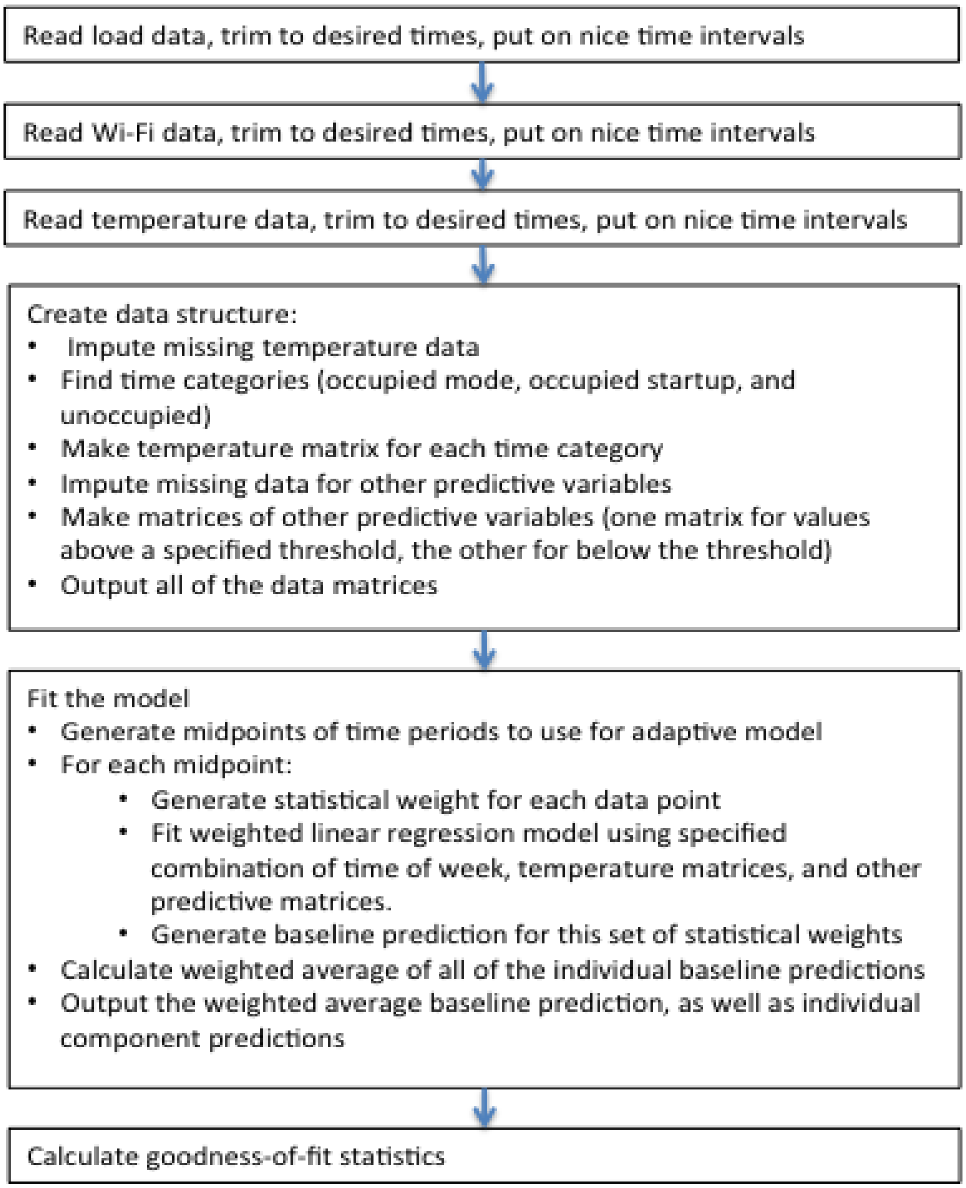
Now consider the prediction for June 30, 2014. This is d=180 days after the end of the training data. In making a prediction for June 30, the regression that has a central time point on December 31 is given a weight of 0.006. The regression with a central time point 14 days earlier has a weight of 0.005. The regression with a central time point another 14 days earlier has a weight of 0.004. Even the regression with a central time point a full year ago, at the end of June, 2013, has a weight of over 0.001, which is still 17% as much weight as the regression with the most recent central time point. Even the regression with the most distant time point, all the way back on January 1, 2013, is assigned 10% as much statistical weight as the regression with the most recent central time point. Thus, in contrast to the baseline prediction for early January 2014, which is based almost entirely on the previous month or two of training data, the baseline prediction for June 2014 ends up being an average of regression predictions that take into account the full year of training data, although still weighting the last half of the year more heavily than the first half.

We believe, based on limited tests of the model for several sites, that in most cases the optimal value of D will probably be of the order of 10 to 20 days both for quantifying demand response effectiveness and for making long-term predictions suitable for M&V applications. (Using smaller values of D cause the baseline prediction to be influenced strongly by anomalies or changes in building load shape that only last a few days or a week, whereas much larger values for D prevent the predictions from adapting to long-term changes in load patterns.

Software Implementation

The model is implemented in the R language. It uses only the standard libraries.

The program flow is straightforward, and is shown in the following diagram. Immediately following, we list each function in the package, and give the first several lines of comment code, which includes a brief description of the inputs, outputs, and functionality.



getTime = function(timeInfo,verbose=1,format=NULL)

given a vector of timestamps, return a vector of POSIXlt times

Input timestamps can be in any of the following formats:

1. Year-month-day Hours:Minutes

1. Year-month-day Hours:Minutes:Seconds

2. Seconds since 1970-01-01 00:00:00

3. Milliseconds since 1970-01-01 00:00:00 (as generated by sMAP for example)

trimDat = function(start, timevec, yvec, end=NULL, nDays = NULL)

Given a start time, a vector of times, and a data vector,

trim the time and data vectors to include just a specified

time interval. The interval may be specified by the total number of days or

by the end time. (If both are specified, end time is used).

makeTempMatrix = function(tempF,Tknots = c(40, 55, 65,75,90),

checkBins=T, verbose=0)

Input: vector of temperatures (degrees F)

Output: A matrix that breaks each temperature into bins, suitable for feeding

into a linear regression so as to get a piecewise-continuous model. For example,

with the default bins, a temperature of 58 will yield the following row of the

matrix: 40 15 10 3 0

this means "40 degrees up to 40F, plus 15 degrees to get to 55F, plus 10 degrees

to get to 65 F, plus 3 degrees to get to 68F, plus 0 degrees in the bin above 80F"

fillGaps = function(xvec,minGap=3)

input xvec is a vector of T and F, representing whether time period is an

occupied mode or not, e.g. c(F,F,F,F,F,T,F,F,T,T,T,F,T,T,F,F,F,F,F)

If there are a few F sprinkled in among T, we want to replace them with T: they

are probably occupied periods that the algorithm didn't flag as such.

minGap specifies the minimum gap in NUMBER OF INTERVALS. E.g. minGap=3 could

```
        # mean 3 hours (if data are hourly) or 45 minutes (for 15-minute data).


findOccUnocc = function(intervalOfWeek,loadVec,TempF,verbose=1)
        # Figure out which times of week a building is in one of two modes
        #  (called 'occupied' or 'unoccupied'). This is NOT based on whether
        # occupants are present: rather, in "occupied mode" the building is load is more
        # sensitive to outdoor air temperature than in "unoccupied mode."
        # Define 'occupied' and 'unoccupied' based on a regression
        # of load on outdoor temperature: times of week that the regression usually
        # underpredicts the load are categorized as'occupied mode', the rest are
        #'unoccupied mode' This is not foolproof but usually works well.

createDataStructure = function(tLoad,yLoad,tTemp=NULL,yTemp=NULL,
        tPredictors=NULL,xPredictors=NULL,xPredThresh = NULL, verbose=0)
        # Create a data structure containing time, load, temperature, and predictive variables
        # The load data are king: put the temperature data and other predictors on the same time intervals as the load data
        #
        # * Impute temperature and other predictive variables if necessary.
        # * Create temperature matrices to fit a piecewise-linear dependence on temperature
        #       - fit separate temperature dependence for occupied and unoccupied modes, and startup period.
        # *     Optionally, create separate matrices of other predictive variables to allow     different behavior when the variable is above vs
        # below a specified   percentile (specified by xPredThresh).
        # For instance, if a predictor variable is the number of active WiFi connections, you could allow a
        # different relationship between load and number of connections
        # when the number is below the 20th percentile than when it is above the 20th percentile.


findTimeCategories =function(tLoad,load,temp=NULL,intervalMinutes,
```

intervalOfWeek=NULL, startMinutes=120,verbose=0)

Find periods of the week that the building is in "occupied mode" and "unoccupied mode"

and split the occupied mode into "startup" and "rest of day"

Note that these modes do NOT depend on actual occupancy: they are based on when data

suggest the building is heated or air conditioned to a greater or lesser degree.

fitModel = function(timeVec, yVec, predFrame, timescaleDays = 14,

verbose=0)

Fit a linear model to predict yVec as a linear combination of predFrame.

Inputs:

timeVec: one timestamp per data point (may be numeric,

string in Y-m-d H:M format, or a POSIX time).

This is used to make final predictions that adapt with time

if the behavior of the system changes with time.

yVec: the variable to be predicted. Numeric. May contain NA.

predFrame: A data frame or vector of predictive variables. May contain "time-of-week"

variables as a column of type "factor."

GoodnessOfFit = function(time1, loadVec, time2, baselinePred,

verbose=1)

Calculate various metrics of how closely a baseline prediction matches actual load. Inputs are time vector and load vector, and time vector and baseline vector.

In the future this should be modified to accept vectors that span different time intervals, and simply choose the common interval.

For now, it simply assumes that if the vectors are the same length, they're directly comparable.

This does _not_ adjust for degrees of freedom, nor does it perform cross-validation; it merely calculates things

like RMSE; so it will be too optimistic if a baseline is fit with

a small timescale (so that it is responsive to week-to-week changes in the load).

So to get a good estimate of model fit, either

1. Use cross-validation (predict weeks or months that are not in the input data), OR

```
	# 2. When fitting the model, use timescaleDays greater than
	# about 28.
	# These should give similar results when predicting a few hours or even one week.

quickPlot = function(dataStruct,title=NULL,filename=NULL)
	# Input is the standard data structure
	# This plots load, Wi-Fi connections, and outdoor temperature vs time, only for non-imputed data.
	# This function lacks generality, e.g. it requires all load, Wi-Fi, and temperature
	# specifically: axis labels are hard-coded and it can't handle additional occupancy variables etc.
```