SunCast: Fine-grained Prediction of Natural Sunlight Levels for Improved Daylight Harvesting

Jiakang Lu and Kamin Whitehouse
Department of Computer Science, University of Virginia
Charlottesville, VA, USA
{jklu,whitehouse}@cs.virginia.edu

ABSTRACT

Daylight harvesting is the use of natural sunlight to reduce the need for artificial lighting in buildings. The key challenge of daylight harvesting is to provide stable indoor lighting levels even though natural sunlight is not a stable light source. In this paper, we present a new technique called SunCast that improves lighting stability by predicting changes in future sunlight levels. The system has two parts: 1) it learns predictable sunlight patterns due to trees, nearby buildings, or other environmental factors, and 2) it controls the window transparency based on a quadratic optimization over predicted sunlight levels. To evaluate the system, we record daylight levels at 39 different windows for up to 12 weeks at a time, and apply our control algorithm on the data traces. Our results indicate that SunCast can reduce glare by over 59% over a baseline approach with only a marginal increase in artificial lighting energy.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-Based Systems**]: Real-time and Embedded Systems

General Terms

Design, Experimentation, Performance

Keywords

Fine-grained Prediction, Sunlight, Daylight Harvesting, Wireless Sensor Networks

1. INTRODUCTION

Artificial lighting is the single largest energy consumer in commercial buildings, accounting for 26% of their total energy usage [1]. *Daylight harvesting* is the approach of using natural sunlight inside a building in order to reduce the electricity demand of artificial lighting. This approach holds particular promise for commercial buildings because they are primarily occupied during daylight hours. The key challenge is to provide stable levels of illumination (typically 500 ± 250 lux) even though natural sunlight is not a stable light source. An office should have enough light to read and work but not so much that it causes glare and discomfort, despite the fact that sunlight levels can change from 100 lux to 1000 lux or more in a matter of minutes due to passing shadows from clouds, trees, and nearby buildings. An emerging approach is to use electrochromic glass, also called smart glass [2], or motorized window blinds [3] to automatically adjust the transparency of a window. When the natural light source is too bright, the window transparency is decreased. When it is too dim, the window transparency is increased and supplemental artificial lighting may be used. Daylight harvesting has been demonstrated to reduce lighting energy by up to 40% in offices that have significant amounts of daylight [4, 5]. In addition, natural light is more pleasant and comfortable than artificial light and has been shown to increase employee productivity [6].

Despite the potential benefits, current daylight harvesting installations have achieved limited effectiveness. A recent study shows that 50% of existing photo-controlled daylight harvesting systems are disabled by the users and the other 50% operate at 50% of their intended performance [7]. One reason is that natural lighting levels can change very quickly but window transparency can only be changed relatively slowly. Rapid changes to window transparency cause confusion and annoyance to building occupants [8], and some windowing systems also have a physical limit on the rate of transparency change [9]. Any difference between the maximum window change speed and the rate of change in natural daylight either introduces glare (people disable the system) or causes energy waste (poor performance).

In this paper, we address the problem of minimizing both glare and energy usage, given that window transparency is subject to a *maximum switching speed*: the maximum instantaneous rate at which the window transparency can be changed. A daylight harvesting system has two forms of lighting actuation that offer different points in the energy/speed trade off: window transparency changes slowly but consumes little to no energy, whereas electric lighting can change quickly but consumes significant energy. Importantly, electric lighting offers only one-directional actuation: it can increase illumination, but cannot reduce illumination. Therefore, bright sunlight can only be addressed by a reduction in window transparency but, being slow, this cannot prevent a temporary glare spike if natural lighting levels increase suddenly. To address this problem, a daylight har-

IPSNŠ12, April 16–20, 2012, Beijing, China.

vesting system must *predict* glare before it happens so that it can reduce window transparency in advance, using electric lighting to compensate as necessary until the natural lighting levels increase. By predicting rapid increases in natural lighting, the system can provide constant lighting levels by converting temporary glare spikes into negative spikes that can be addressed through additional energy consumption. However, these daylight predictions must be fairly precise in terms of both the timing and magnitude of daylight changes: a rise in sunlight levels that is later or smaller than predicted will result in energy waste, whereas those that are earlier to larger than predicted will result in a glare spike. Thus, a key challenge of daylight harvesting is to adjust window transparency in anticipation of future changes in sunlight levels.

We present a new technique called *SunCast* that improves daylight harvesting performance by using fine-grained prediction of natural sunlight levels. SunCast is an on-line system that makes a new prediction and issues a new control command at every moment in time. It has two parts: a sunlight prediction algorithm and a control algorithm. To predict sunlight values, it first defines the *similarity* between sunlight values observed on previous days and those observed up until the current time on the current day. Then, it defines the distribution of future sunlight levels to be a weighted combination of historical sunlight levels at the same times of day, weighted by the previous days' similarity values. For example, by 10:00 AM one day, SunCast observes sunlight patterns typical of a sunny day. Of all historical data traces that exhibited similar patterns, some remained sunny while others became cloudy. Of those days that remained sunny, all days exhibited a trough from 11 to 11:30 AM due to a shadow from a nearby tree or building. SunCast combines all of these historical traces to produce a distribution of predicted sunlight values at every time in the future. Thus, instead of making an explicit model of a node's environment, SunCast uses a purely data-driven approach to create empirical distributions over both predictable and unpredictable features of sunlight time series. Weather predictions, day of year, or other explicit information about the environment can be integrated into SunCast by including it in the function used to match with historical traces.

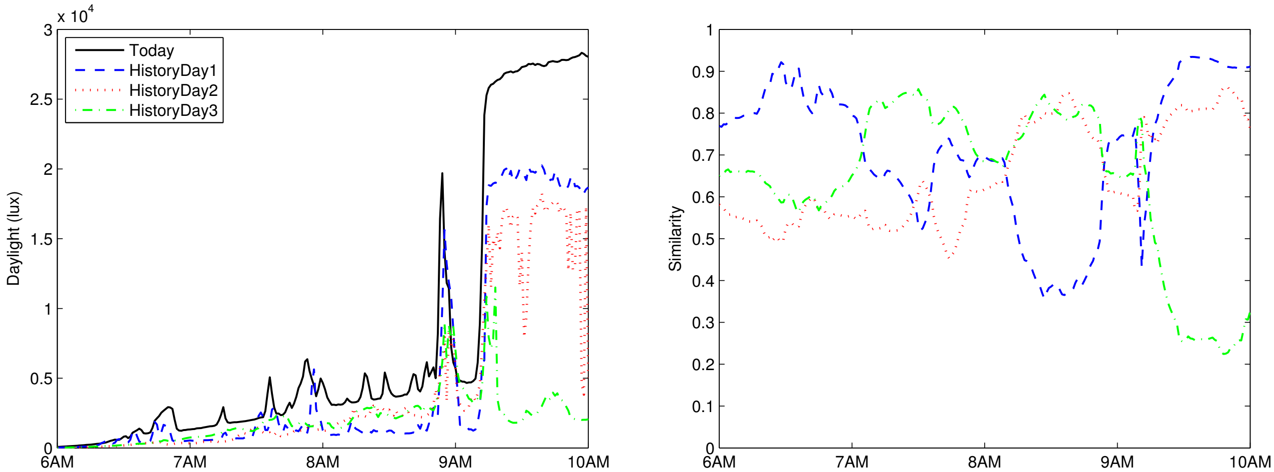
In contrast to prior techniques that predict average sunlight levels over a time period, SunCast predicts the actual sunlight values for every minute in a future time window, which allows the daylight harvesting system to set a specific window transparency level for each minute. Furthermore, SunCast predicts a distribution of sunlight levels, instead of predicting just a point estimate. This allows the daylight harvesting system to calculate the expected glare and energy usage for any given transparency level at any point in time, weighted by the probability distribution over the predicted light values. These distributions allow the system to identify predictable sunlight patterns such as sunrise or shadows from trees and nearby buildings, and to distinguish these from unpredictable patterns such as cloud movement.

To control the window transparency, SunCast uses a form of predictive control. Given a set of sunlight predictions, it uses quadratic optimization to choose a complete sequence of future window transparencies to minimize expected glare and energy, subject to switching speed constraints. It issues the first transparency value in that sequence as a control signal to the window. Every minute, a new light value is observed, the distributions and optimized values are updated, and a new control signal is issued. We evaluate this approach by deploying light sensors in 39 different locations for up to 12 weeks at a time, and applying the control algorithm on the data traces. Our results indicate that SunCast can reduce glare by over 59% over a baseline approach with only a marginal energy penalty. We conclude that SunCast helps solar energy harvesting technologies exploit predictable, large-scale, short-duration fluctuations in solar energy levels to substantially reduce glare and improve the comfort levels produced by existing energy harvesting systems.

2. BACKGROUND AND RELATED WORK

Solar radiation accounts for most of the renewable energy on Earth, with total solar irradiance measuring roughly 1.3 kW/m^2 [10]. Artificial lighting constitutes a large fraction of energy usage in commercial buildings, despite the fact that they are occupied primarily during daylight hours. The reasons for this energy usage are plentiful, and daylight harvesting cannot address all of these reasons. For example, lights are often left on at night for security reasons, and many buildings have a wide footprint so light from windows cannot reach the center of the building. However, daylight harvesting systems have gradually gained popularity in modern buildings and have been shown to have the potential for up to 40% energy savings [4, 11, 12]. Many new buildings are being designed with natural lighting in mind, and building codes in some countries as well as some LEED certification levels require all rooms to have natural lighting. The Lawrence Berkeley National Laboratory (LBNL) recently deployed a well-known daylight harvesting system in the new New York Times Headquarters Building [13] along with a field study that enhances the understanding of daylighting controls. Other daylight harvesting systems have been developed to minimize energy consumption, balance diverse user lighting preferences, and increases facilities managers' satisfaction [14]. In such systems, window shading technologies such as electrochromic windows [2] and motorized blinds [3] are widely used. However, the switching speed of electrochromic windows sometimes range from several minutes to up to two hours, depending on window size and outdoor temperature [9]. Even mechanical blinds have a maximum switching speed because rapid changes to the blind position has been shown to cause confusion to the user [8]. Furthermore, changing window transparency consumes energy. Limits on maximum switching speed can cause lighting errors and reduce user comfort, which ultimately leads to energy waste if users disable the system or configure it to be less aggressive in order to reduce glare [7].

The Illuminating Engineering Society of North America (IESNA) recommends 500 lux as the standard task illuminance for office workers performing regular tasks [15]. However, visual comfort depends on task requirements and individual user preferences. The upper limit on lighting depends on glare requirements, and is different for reading paper documents versus reading a back lit computer screen. The lower limit is usually discussed in terms of detectable vs. acceptable illuminance: 10-15% is generally within the undetectable range for most people assuming base levels of 500 lux [16], and up to 40% may be acceptable depending on how slowly the light is dimmed [17].



(a) Daylight

(b) Similarity

Figure 1: SunCast uses a *similarity metric* to identify historical data traces with patterns most similar to the current day. In this example, the most similar day changes over time.

Many sunlight prediction techniques have previously been developed, primarily in the context of solar energy harvesting with solar panels, and each techniques makes predictions based on different information and over a different period of time. Weather forecasts today are based on vast sensing infrastructure and advanced computer simulations [18, 19]. Sharma et al. explore the use of weather forecasts to improve energy harvesting prediction [20]. Forecasts can help predict cloudiness levels in the sky, but do not predict the effect of shadows and reflections at a particular location on the ground, which depends on the proximity to nearby buildings, the presence of leaves on trees, time of day, and seasonal changes in the azimuth of the sun. These factors affect solar energy levels as much haze, clouds, and precipitation. Today, websites provide hourly predictions of cloudiness levels, but even more fine-grained information is needed for control of window transparency. Classical time series analysis would suggest using auto-regression techniques [21], but any such model would change rapidly throughout the day and would depend on many external, unobserved variables.

Recently, several new approaches have been developed to predict the solar energy levels at a single point, most of which have focused on solar-powered sensing [22]. Some of these techniques choose a fixed sensor sampling rate based on long-term expected sunlight levels [23, 24, 25], while others make near-term predictions, e.g. 3-72 hours in advance [26, 27]. Other techniques use an EWMA over previous days [28, 29] or statistical correlations based on weather predictions [30, **?**]. However, solar energy harvesting is very different from daylight harvesting because solar energy can be stored in a capacitor or battery, whereas sunlight cannot be stored. Therefore, fine-grained prediction is not as essential for solar energy harvesting: the storage unit acts as a buffer and delays the impact of sunlight changes, giving the system more time to adapt by, e.g. changing the sampling rate. Unless storage is extremely limited, therefore, solar-powered sensing applications can suffice with predictions of *average* sunlight levels. In contrast, daylight harvesting systems have no buffer to delay a glare spike, and must therefore use fine-grained prediction to accomodate rapid changes in sunlight levels.

Wireless sensor networks (WSNs) have previously been used for light sensing and actuation to achieve cost effectiveness, energy efficiency, and user comfort. For example, Singhvi et al. proposed and demonstrated a lighting control system with wireless sensors and a combination of incandescent desk lamps and wall lamps actuated by the X10 system [31]. In addition to office lighting applications, Park et al. designed and implemented Illuminator, an intelligent lighting control system for entertainment and media production [32]. High fidelity wireless light sensors were developed and implemented to form a sensor network for collecting stage lighting information [33]. The SunCast daylight harvesting system also senses and controls light values, but the goal different: to achieve stable task lighting despite unstable natural sunlight levels.

3. PREDICTING SUNLIGHT VALUES

SunCast uses a three-stage process to generate fine-grained, continuous distributions of predicted sunlight values. First, the system calculates the *similarity* between the real-time data stream and historical data traces (Section 3.1). Second, it uses a regression analysis to *map* the trends in the historical traces to more closely match patterns of the current day (Section 3.2). Third, the system combines the weighted historical traces to predict the *distribution* of sunlight in the near future (Section 3.3).

3.1 Finding Similar Days in History

SunCast calculates the similarity of the current sunlight levels with all historical data traces previously observed. Calculating the similarity has two steps. First, we calculate the squared error between the real-time data stream and the historical data stream, for a time window of n readings in the recent past. The current data stream in the sliding window between t_1 and t_n is defined as $DS = \{x_{t_1}, x_{t_2}, ..., x_{t_n}\}$, and

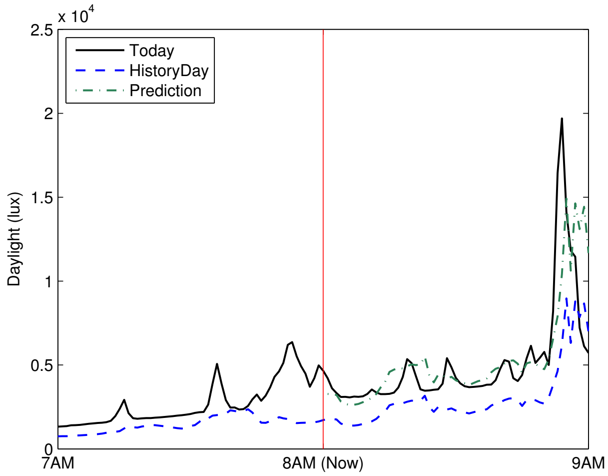


Figure 2: SunCast uses regression to improve the mapping between today (solid) and a historical day (dashed) to improve future the predictions based on that historical day (dashed-dot).

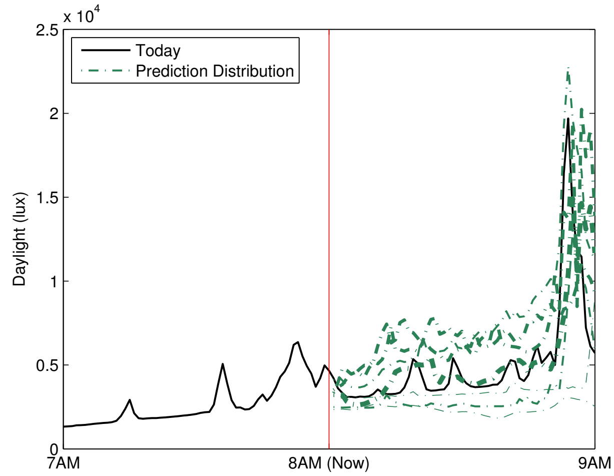


Figure 3: SunCast uses a weighted combination of historical data traces to create a distribution of predicted light values, depicted here by the weighted dashed lines.

the data stream of historical trace j is defined as $DS_j^{'} = \{x_{t_1,j}^{'}, x_{t_2,j}^{'}, ..., x_{t_n,j}^{'}\}$. The difference d between these two days is calculated as:

$$d(DS, DS_j^{'}) = \sum_{i=1}^{n} (x_{t_i} - x_{t_i,j}^{'})^2 \tag{1}$$

Second, we define a relative ranking among all the h historical traces by normalizing the difference values. Thus, the *similarity* s_j of historical trace j among h history days is defined as:

$$s_j = 1 - \frac{d(DS, DS_j^{'})}{\sum_{k=1}^{h} d(DS, DS_k^{'})} \tag{2}$$

These normalized values are used as *weights* to find the traces most similar to the current data, while still taking the entire historical data set into consideration.

Figure 1 illustrates the similarity metric for an example day with three historical traces using a 15-minute sliding window during the period from 6:00 AM to 10:00 AM. The example day has clear weather and 3 historical days that are hazy, partially cloudy, and sunny but becoming overcast, respectively. This particular sensor has direct sunlight around 8:50 AM, followed by a shadow due to a tree, and direct sunlight again around 9:10 AM. Early in the morning, History Day 3 is most similar, but once History Day 3 becomes overcast during the periods of direct sunlight, the hazy day becomes most similar. This example illustrates the benefits of a relative ordering rather than an absolute metric of similarity.

3.2 Mapping to Current Conditions

In the example above, the most similar historical trace was from a hazy day. This trace contained trends and patterns that are pertinent for predicting today's sunlight values, but the values are offset by a constant factor due to the level of haze. Similar effects are also produced by seasonal changes in sunlight intensity or other factors, and SunCast uses linear regression analysis to map the patterns in a historical trace to the conditions of the current day. The regression is formulated as $Y = b_j + a_j X_j$, where Y is the current data stream $\{y_{t_1}, y_{t_2}, ..., y_{t_n}\}$ and X_j is the jth historical trace $\{x_{t_1,j}^{'}, x_{t_2,j}^{'}, ..., x_{t_n,j}^{'}\}$. After solving for a_j and b_j, we use the linear regression model to predict the future sunlight values for today, based on the historical traces patterns: for a prediction length l, the predicted data based on history day j is

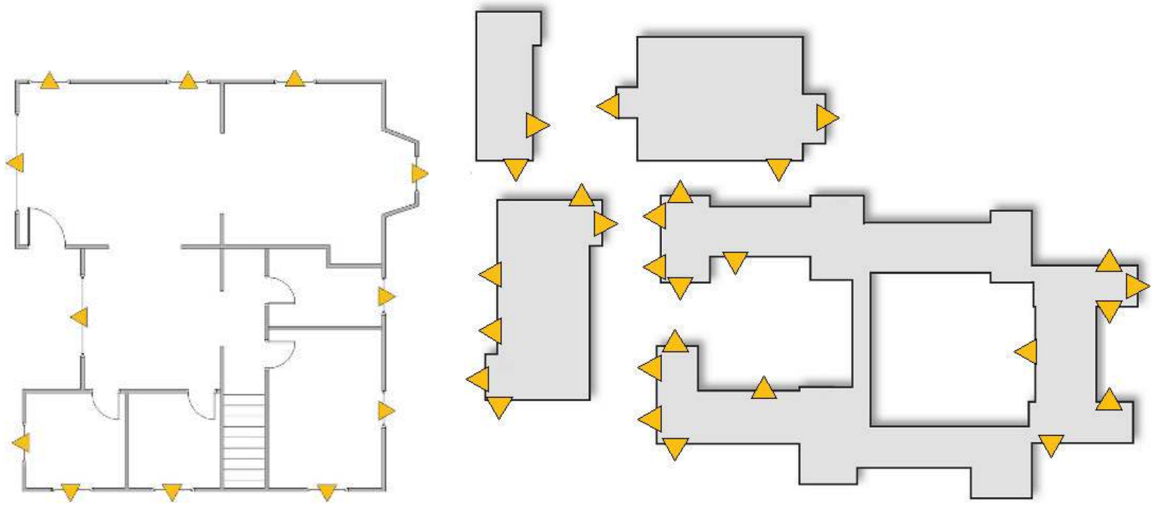
$$Y_j^* = b_j + a_j X_j^* \tag{3}$$

where Y_j^* is the predicted data $\{y_{t_{n+1},j}^*, y_{t_{n+2},j}^*, ..., y_{t_{n+l},j}^*\}$ and X_j^* is the historical trace $\{x_{t_{n+1},j}^{'}, x_{t_{n+2},j}^{'}, ..., x_{t_{n+l},j}^{'}\}$.

Figure 2 illustrates how this mapping process works. In the example, the current clock time is 8:00 AM and our system applies regression between the current data and a historical trace over the prior one-hour time window between 7:00 AM and 8:00 AM. Then, the model learned is applied to the next time one-hour time window between 8:00 AM and 9:00 AM to predict the future values today based on the historical trace. As the figure illustrates, the regression analysis preserves patterns in the historical trace while correcting for constant differences in the slope and bias between the two days.

3.3 Creating a Prediction Distribution

After applying regression analysis to all h historical traces, we apply the regression model to the future time window to produce h predictions of length l. These are combined into a h-by-l matrix. The similarity values can be multiplied against the prediction distribution to produce the *prediction distribution* $\hat{\mathbf{x}}$, as shown in Equation 4, where $\hat{x}_{t_{n+i},h} \in \hat{\mathbf{x}}$ is the weighted sunlight prediction for time $n+i$ based on historical day h.



(a) Residential House Deployment

(b) Campus-wide Deployment

Figure 4: To evaluate SunCast, we deployed light sensors at 39 locations for up to 12 weeks at a time.

$$\hat{\mathbf{x}} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_h \end{bmatrix}^T \begin{bmatrix} x^*_{t_{n+1},1} & x^*_{t_{n+2},1} & \cdots & x^*_{t_{n+l},1} \\ x^*_{t_{n+1},2} & x^*_{t_{n+2},2} & \cdots & x^*_{t_{n+l},2} \\ \vdots & \vdots & \ddots & \vdots \\ x^*_{t_{n+1},h} & x^*_{t_{n+2},h} & \cdots & x^*_{t_{n+l},h} \end{bmatrix} \quad (4)$$

Figure 3 illustrates an example of prediction distribution using the same example day above and ten historical traces. Each dashed-dotted line represents the expected sunlight based on the prediction of one historical trace, and the thickness of the line indicates the similarity between the current day and that history day. As the figure shows, SunCast produces a fairly wide distribution for the time between 8:00 AM and 8:50 AM, due to varying levels of haze that might be encountered. However, given that the data between 7:00 AM and 8:00 AM is fairly indicative of a clear day, the most heavily weighted predictions all have a peak in sunlight around 8:50 AM when direct sunlight first hits this node and before the shadow of a tree. Two predictions of a cloudy day (thin lines at the bottom) remain in the prediction distribution, but both have very low weights.

4. SETTING WINDOW TRANSPARENCY

In this section, we present a mathematical formulation of the daylight harvesting problem to illustrate how the SunCast prediction distributions can be used for on-line window control. We define window transparency wt to be the percentage of incoming daylight that penetrates the window: the window is fully closed at 0% transparency and fully open at 100%. The *setpoint* is the desirable lighting level for task illumination: too much harvested light will cause *glare*, while too little will increase *energy* consumption of artificial lighting. The window switching speed *wSpeed* is the maximum percent change in window transparency allowed per minute. A daylight harvesting system does not need to predict far into the future because values in the far future do not affect current control parameters. We define a maximum prediction window len to be

$$len = \frac{max(100\% - wt, wt - 0\%)}{wSpeed} \quad (5)$$

This window size ensures that the system predicts far enough that it is always able to respond to predicted values; larger values of *wSpeed* lead to smaller prediction windows. Then, the system finds a series of window transparency values that minimize the expected lighting error for the entire prediction window with k historical traces: $predDist_{k,len}$. The objective function is

$$minimize \sum_{j=1}^{k} \sum_{i=n+1}^{n+len} |wt_{t_i} \times \hat{x}_{t_i,j} - setpoint| \quad (6)$$

subject to limits on both window transparency and switching speed

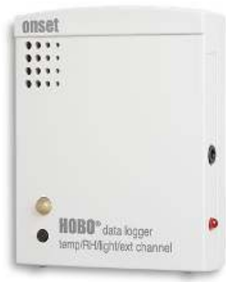
$$0\% \leqslant wt_{t_i} \leqslant 100\% \quad (7)$$

$$|wt_{t_{i+1}} - wt_{t_i}| \leqslant wSpeed \quad (8)$$

Once the optimization function is solved, the system updates the current window transparency to the first value from the solution derived: $wt^*_{t_{n+1}}$. All other transparency values from the solution are discarded, and were only calculated to ensure that target transparency values for the future could still be achieved given switching speed constraints. This entire process is repeated every time step when a new light reading is measured.

4.1 Balancing Prediction and Reaction

The algorithm described above is a pure prediction algorithm, which is ideal for preparing in advance for predictable rapid changes in sunlight, such as sunrise, sunset, or a shadow. However, during periods of stable sunlight, such as mid-day, predictions from historical traces will actually hinder performance because current conditions are a better predictor of future values than any historical trace. In such cases, better performance is achieved by a *reactive* algorithm that sets the window transparency based on the



(a) Light Sensor

(b) Open View

(c) Blocking Tree

(d) Blocking Building

Figure 5: The light sensors we deployed were subject to a wide range of environmental influencers that caused shadows and reflections throughout the day.

current sunlight values, without making any predictions. In order to accommodate this condition, SunCast introduces a *hybrid* scheme that switches smoothly between prediction and reaction: it uses a purely reactive scheme on each historical trace j and, if the the lighting error is below a threshold β, then all values in the row j of $\hat{\mathbf{x}}$ are replaced with the current light reading. The basic rationale is that if a historical trace is stable enough that pure reaction performs sufficiently well, then the trace is not providing any useful prediction information. In fact, it can even be harmful to performance because window transparency will be determined by small peaks and troughs from that day that are not likely to re-occur today. Therefore, the trace is only used if it indicates a rapid change in sunlight that cannot be accommodated to the switching speed limitations. The user can tune the balance between prediction and reaction by changing the parameter β.

4.2 Balancing Energy and Glare

Daylight harvesting systems must balance energy usage and glare when responding to predictions about future sunlight. Being too aggressive about energy conservation will introduce glare if there are rapid peaks in sunlight, but being too conservative will cause energy usage if sunlight levels are not as high as expected. In SunCast, we allow the user to customize the balance between energy and glare by introducing a new control variable called *daylight weight*, which tunes the maximum percentage of lighting that can be provided by natural daylight. If the daylight weight is 0%, then the system uses purely electric lighting without any concern about glare. If the daylight weight is 100%, then the system maximize the opportunity to harvest natural sunlight, but risks suffering from more glare. One alternative to this scheme is to define a different penalty function for glare or

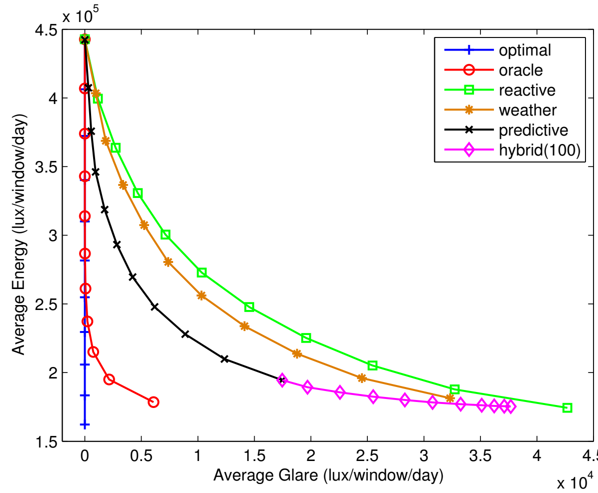


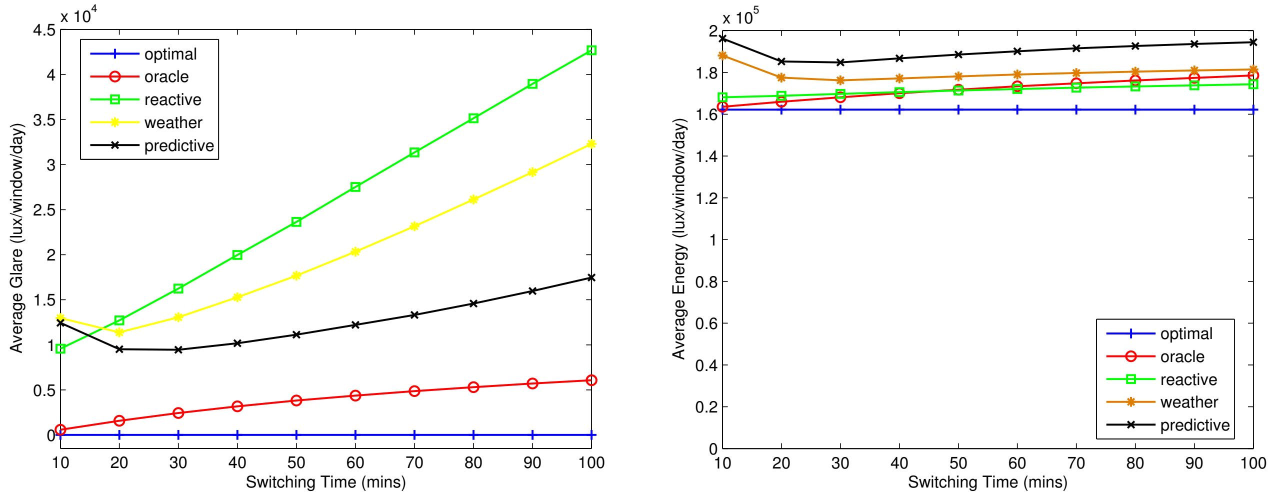
Figure 6: The two algorithms based on SunCast, *prediction* & *hybrid(100)*, offer a more desirable trade-off between glare and energy usage than the *reactive* scheme, and begin to approach the performance of *oracle* and *optimal* schemes.

energy usage, which would allow the system to achieve a more subtle trade off between energy and glare. However, such a scheme could no longer be formulated as a quadratic optimization, and we believe that users will find it easier to set a single linear knob rather than to create a customized penalty function.

5. EXPERIMENTAL SETUP

Daylight intensity and distribution are deeply dependent on geographic location, building orientation, sky conditions and nearby surroundings. Therefore, in order to evaluate SunCast in a realistic environment, we deployed two testbeds in different buildings: one in a residential house and another of deployment on university campus, as shown in Figures 4(a) and 4(b). The testbeds consist of 12 and 27 U012-12 Hobo data loggers designed by Onset, as shown in Figure 5(a). These nodes can monitor the environment with the built-in light, temperature, and humidity sensors. At each window, a node facing the outside measures the incoming daylight. Some nodes are deployed at windows with open views while others are located behind a blocking tree outside. Several examples of deployment are shown in Figure 5.

We collected the light sensor data from the testbed at a sampling rate of once per minute, and the duration of the sensor deployment lasted for 4 weeks at the house and 12 weeks on campus. During the deployment period, we observed a wide variety of weather patterns, including sunny, cloudy, rain, fog and snow storm. All the data were stored in the data loggers and are manually read out from the nodes every week. The system presented in this paper does not require communication among nodes, but these data loggers could be replaced with wireless sensors for convenience.



(a) Glare

(b) Energy

Figure 7: As the switching time (minimum time to go from completely transparent to opaque) increases, the predictive scheme increasingly outperforms reactive in terms of glare and is a constant factor worse in terms of energy usage. For very short switching times, prediction does not help.

6. EVALUATION

In this section, we evaluate how SunCast predictions affect the performance of daylight harvesting. We evaluate the system on both campus and house testbeds. Due to space limitations, we present only the results of the campus testbed. The results from the residential testbed produced nearly identical trends.

6.1 Baseline and Optimal Algorithms

As a baseline for comparison, we use a purely *reactive* scheme that uses closed-loop feedback control to set the window transparency: it periodically measures the current daylight and sets window transparency to come as close to the target setpoint as possible, subject to the switching speed constraints.

We introduce another baseline scheme called *weather*, which uses the same optimization formulated in Section 4, except that it operates on the subset of history days that have the same cloudiness level as the current day. We classify the history days with the daily cloudiness levels based on the weather reports from local airports. This scheme provides insight on the impact of selecting history days on the system performance.

We upper bound the benefits of prediction using another scheme that we call *oracle*, which uses the same optimization formulated in Section 4, except that it operates on the actual future light values instead of predicted values. This scheme provides the best performance possible with the control algorithm used in our analysis, assuming perfect daylight prediction.

Finally, we upper bound daylight harvesting performance using an *optimal* algorithm that always uses the window transparency that minimizes both energy and glare. This scheme is not subject to switching speed constraints, and provides the theoretical upper bound on energy and glare for any control scheme. If daylight levels are high enough, it will produce no glare and no energy usage. Electric lighting will only be used when daylight levels are below the target setpoint.

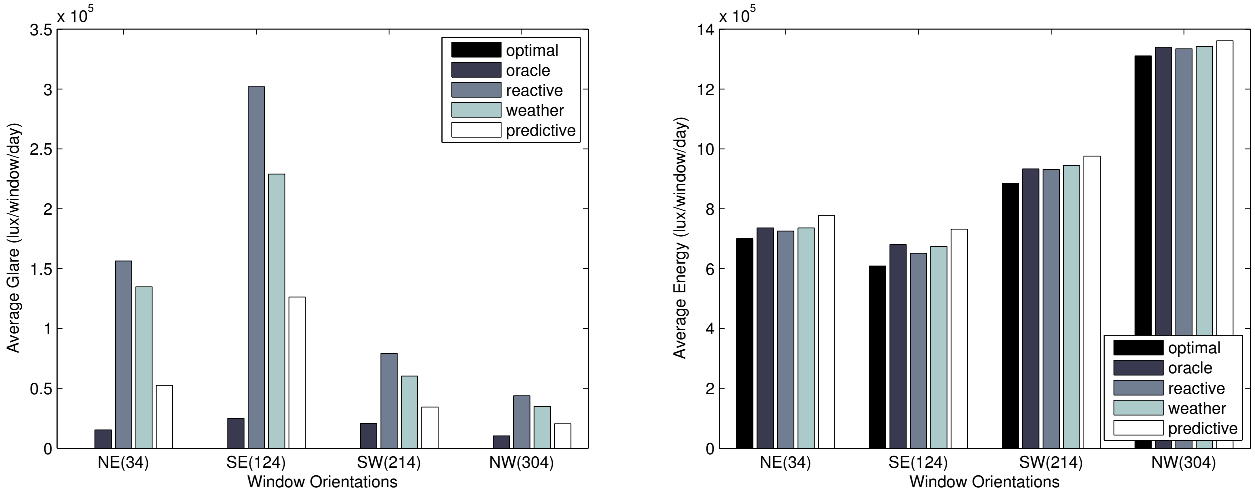
6.2 Evaluation Metrics

We evaluate the performance of the daylight harvesting system in terms of two evaluation metrics: energy and glare. *Energy* is defined as the amount of artificial lighting used by the scheme to maintain the setpoint at the window, measured in lux per window per day. We did not evaluate the energy in kilowatt hours, because that depends on the type of light bulb assumed. The values can easily be converted by assuming a particular type of light bulb. *Glare* is defined as the amount of the harvested light above the target setpoint, also measured in lux per daylight harvesting window per day.

6.3 Experimental Results

We evaluate our system against the baseline and optimal algorithms in a trace-based simulator that replays the empirical data traces from the campus testbed and executes the algorithm described in Section 4. The simulator allows the control scheme to adjust window transparency and measures both glare and electric lighting usage. In our experiments, we use 2,000 lux as the setpoint at the window, which is chosen to produce lighting levels in the interior spaces close to the industry standard of 500 lux. The control loops repeat every 1 minute, which is the sampling rate with which the light data was collected. We test a range of daylight weights from 0% to 100% for all the control schemes, and investigate the effect of balancing between prediction and reaction at the daylight weight of 100%. In the hybrid scheme, we set error thresholds varying from 0% to 100% of the setpoint at the window.

The experimental results of daylight harvesting are shown in Figure 6, where each line represents a different control scheme and the points on the line indicate the average energy and glare results as the daylight weight. All points on



(a) Glare (b) Energy

Figure 8: The south-east facing windows suffer the most glare because the window is completely transparent right before sunrise. The predictive scheme reduces this glare substantially. Energy usage is dominated by dusk and dawn, and is therefore not sensitive to orientation.

the *hybrid(100)* line use 100% daylight weight, and instead the points represent the energy/glare performance for varying values of the β threshold that switches between reaction and prediction. If the daylight weight and the β threshold are integrated into a single user-controlled knob, the predictive and hybrid(100) schemes form a single curve that represents the range of SunCast performance. This curve provides the user with a much more desirable balance between glare and energy usage than the reactive scheme, and approximately halves the distance between *reactive* and *oracle* performance. For total glare levels of about 10,000 lux per day, the predictive scheme consumes about 20% less energy than the reactive scheme. Similarly, at artificial lighting levels of 200 kilolux per day, the predictive scheme produces more than 45% less glare than the reactive scheme. In contrast, the approach based on weather classification reduces only 3.7% energy and 18.6% glare over the reactive scheme.

7. ANALYSIS

In this section, we analyze the degree to which fine-grained prediction contributes to achieving stable task lighting under different scenarios. We also discuss the impact of window switching speeds, window orientations, and cloudiness levels on the performance of daylight harvesting.

7.1 Sensitivity to Switching Speed

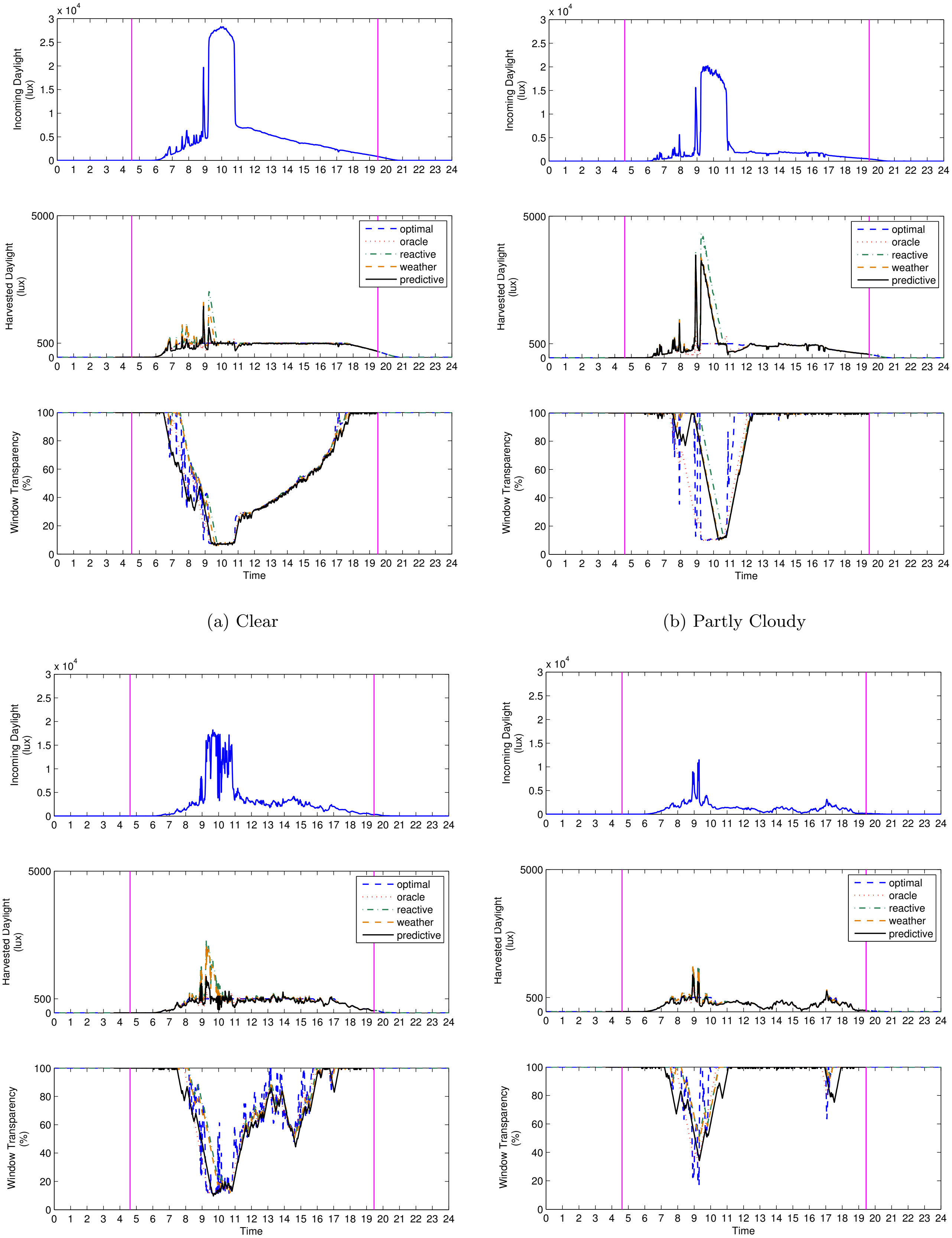
To investigate the impact of switching speed on the system performance, we run the same control schemes with a range of switching times varying from 10 minutes to 100 minutes: these are the minimum times required to make a full transition from transparent to opaque. These values represent an estimated range of comfortable speed changes, as well as the physical limits of some electrochromic windows [9]. For a fair comparison, all of these experiments use 100% daylight weight. Figure 7 shows the energy and glare levels of all algorithms as the switching time is varied. Figure 7(b) shows that the predictive scheme always wastes slightly more energy than the reactive scheme, for a given switching time, but Figure 7(a) shows that the predictive scheme outperforms reactive in terms of glare. Predictive performs best when the minimum switching time is long: at the switching time of 100 minutes, it achieves 59% less glare than the reactive scheme. As the minimum switching time gets smaller, prediction into the future is no longer beneficial and even hinders performance. At the same time, the *reactive* scheme approaches the *oracle* scheme, and at switching times of less than about 15 minutes, reactive begins to outperform predictive.

7.2 Sensitivity to Window Orientation

Figure 8 shows the results of energy and glare broken down by the direction that the sensors are facing on the campus testbed. These results indicate that windows facing the southeast have the highest level of glare. This is because the window is in the fully transparent state at dawn and is suddenly subject to bright, direct sunlight after sunrise. The predictive scheme anticipates the sunrise and reduces glare by over 50%. The northwest windows consumes the largest amount of energy because there the least direct sunlight from that direction. However, energy usage is dominated by dawn and dusk and therefore does not change substantially with window direction. This analysis does not illustrate the effects of predicting rapid changes due to trees, windows, or other predictable shadows, since these factors are different for each window and are averaged out over all windows facing each cardinal direction.

7.3 Sensitivity to Cloudiness Level

Figure 9 illustrates the system performance of the control schemes on the four example days that represent typical cloudiness levels: clear, partly cloudy, most cloudy, and overcast. The diagrams show detailed traces of: 1) sunlight levels during the day; 2) harvested daylight of control schemes; and 3) window transparency adjusted by control schemes. For all control schemes, the daylight weight is



(a) Clear

(b) Partly Cloudy

(c) Mostly Cloudy

(d) Overcast

Figure 9: The predictive scheme based on SunCast is most advantageous over reactive on sunny and overcast days because the sunlight patterns are most predictable. On partially cloudy days, unpredictability causes both glare and energy waste.

100% and the window switching speed is 1%/min. The results show that the predictive scheme approaches the optimal window control on a clear or overcast day. This is because sunlight levels are relatively stable on a clear or overcast day, so sunlight patterns are more predictable. Cloudy days are less predictable. For example, the predictive scheme produces high glare between 9:00 AM and 10:00 AM on a partly cloudy day. This is caused by the sudden sunlight increase around 9:00 AM, which is not predicted by the historical traces: in the early morning, the sunlight levels resembled a cloudy day, but shortly after sunrise the levels approached that of a hazy day. As the sunlight variability becomes smaller, the predictive scheme has better performance on the mostly cloudy day.

7.4 Predictability Analysis

We analyze the times at which SunCast has the largest effect on *task lighting stability*: the variance of indoor lighting levels, as compared with the task lighting setpoint. Lower variance levels indicate that task lighting is more stable. For a fair comparison, all experiments use 100% daylight weight and the window switching speed of 1%/min.

Figure 10 illustrates the system performance of the control schemes on four example windows. Each window has a different predictable feature: morning light, a building shadow, a tree shadow, and blocked view. The diagrams show detailed traces of improvements in the task lighting stability due to both the predictive and the weather-based control schemes. The results show that prediction creates a large improvement in task lighting stability during predictable periods of lighting variability. At other times when the lighting values are not as predictable, the SunCast control algorithm achieves the same task lighting stability as a simple reactive scheme. This analysis demonstrates that the probability distributions created by SunCast help to accurately differentiate predictable and unpredictable patterns, allowing the system to exploit predictable patterns without paying a penalty during unpredictable periods.

8. LIMITATIONS AND FUTURE WORK

The results of this study demonstrate that a data-driven approach can effectively predict natural daylight levels in a way that can improve daylight harvesting effectiveness. However, Section 7.4 illustrates that this approach is limited to sunrise, sunset, trees, nearby buildings, and other relatively predictable environmental factors. Many rapid daylight changes such as those caused by passing clouds are still unpredictable. Our system is designed to revert to a reactive approach during unpredictable periods, as explained in Section 4.1, but in current work we are exploring ways to merge data traces from multiple light sensors deployed throughout a building or a group of buildings to predict cloud boundaries. This *group estimation* approach can also be used to help improve the distribution over future light levels, even for predictable changes. For example, an east-facing window may be best able to predict the intensity of the glare spike at a south-facing window, as the sun rounds the corner of a building. An important challenge of this approach is that it requires communication and coordination between a group of geographically distributed nodes. Therefore, the advantage in terms of daylight harvesting will need to be balanced with the communication and energy demands of the sensor devices.

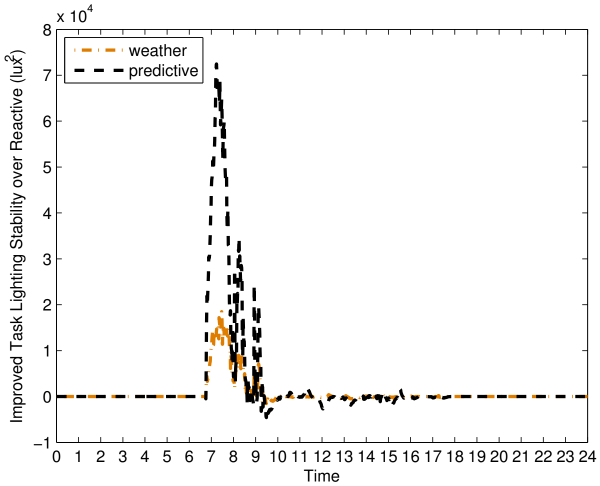
In addition to daylight harvesting, SunCast can also be used for other applications that would benefit from sunlight prediction. We are currently developing new control algorithms for solar-powered sensing systems that employ photovoltaics to harvest solar energy for perpetually-powered sensing. These systems are typically coupled with rechargeable energy storage such as batteries or fuel cells that preclude the need for fine-grained sunlight prediction. However, storage capacity may be limited by cost, size, or weight. For example, a 9-cubic millimeter solar-powered sensing system recently developed uses a 12μAh battery [34]. Many existing approaches to solar-powered sensing choose a fixed sampling rate that minimizes the chance of fully depleting the energy supply, based on long-term predictions of solar energy levels. However, these approaches lose the ability to exploit the small peaks and troughs typical of natural daylight. For example, opportunities for energy harvesting are lost when sunlight levels rise suddenly but the battery is already fully charged. Similarly, data can be lost if the battery is depleted in order to harvest more energy, but sunlight levels unexpectedly drop.

9. CONCLUSIONS

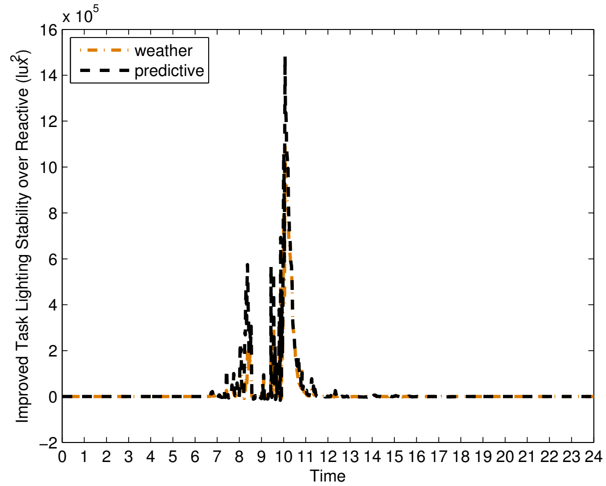
In this paper, we present a new sunlight prediction framework called *SunCast* that produces a distribution of predicted sunlight values in the near future. SunCast is an improvement over existing sunlight prediction schemes in two ways: 1) continuous predictions over time enable exploitation of predicable, large, and short-duration peaks and troughs in sunlight levels, and 2) a distribution of predictions at each time point enables managing the risks and rewards, weighted by the probability that there will be too much or too little sunlight. We present a predictive control scheme based on quadratic optimization that performs daylight harvesting based on these predictions, and evaluate using data traces collected from 39 light sensors deployed in different windows. Our results demonstrate that SunCast can produce substantial performance improvements for daylight harvesting, reducing glare by 59% with only a marginal increase in electric lighting usage. The reduction in glare should will improve the total energy savings from daylight harvesting systems because fewer people will disable the system due to unacceptable lighting comfort. The SunCast prediction and control algorithms can be incorporated into existing daylight harvesting algorithms simply by changing the control algorithm, adding storage for historical data traces, and configuring the parameters: *wSpeed*, daylight ratio, and β. In addition to daylight harvesting application explored in this paper, SunCast can be applied to other problems that would benefit from fine-grained, short-term sunlight prediction. Furthermore, the empirical data-driven techniques could possibly be extended to predict of any data stream that exhibits daily or periodic trends that results from a large number of factors, such as highway traffic patterns, city pollution levels, or office building occupancy.

Acknowledgements

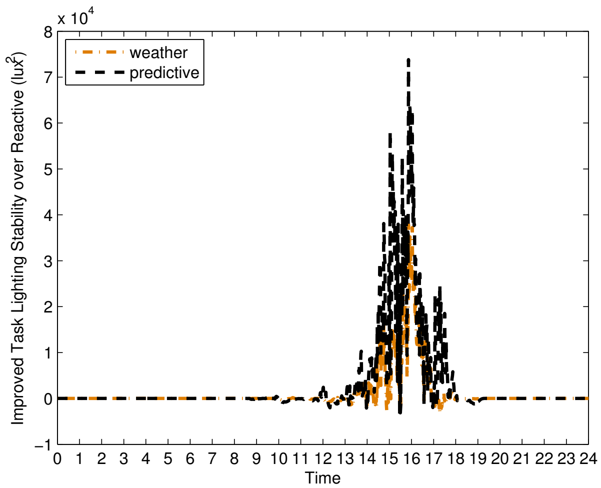
Special thanks to our reviewers for helpful feedback, and to Tarek Abdelzaher for helping to clearly articulate the contributions of this paper. This work is based on work supported by the National Science Foundation under Grants No. 1038271 and 0845761.



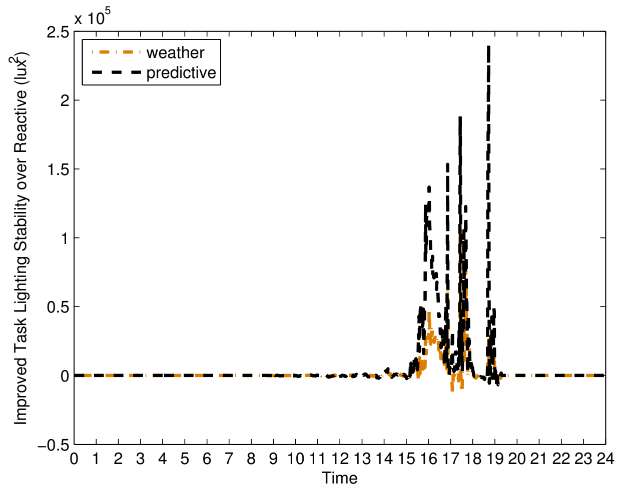
(a) Northeast View (Sunrise)



(b) Southeast View with Large Tree



(c) Southwest View with Nearby Building and Tree



(d) Northwest View with Multiple Trees

Figure 10: SunCast improves lighting stability over the reactive scheme at different times of day for each window, depending on the environmental factors that cause glare. During unpredictable periods, it performs no worse than reactive. The weather based scheme is much less effective at dawn.

10. REFERENCES

[1] U.S. Department of Energy. Energy Efficiency Trends in Residential and Commercial Buildings, 2008.

[2] National Renewable Energy Laboratory. Electrochromic Windows Basics. `http://www.nrel.gov/buildings/electrochromic_basics.html`.

[3] S. Selkowitz and E. Lee. *Integrating Automated Shading and Smart Glazings with Daylight Controls.* Lawrence Berkeley National Laboratory, 2004.

[4] J.D. Jennings, F.M. Rubinstein, D. DiBartolomeo, and S.L. Blanc. Comparison of Control Options in Private Offices in an Advanced Lighting Controls Testbed. *Journal-Illuminating Engineering Society*, 29(2):39–60, 2000.

[5] Y.J. Wen, J. Granderson, and A.M. Agogino. Towards Embedded Wireless-Networked Intelligent Daylighting Systems for Commercial Buildings. In *SUTC '06: Proceedings of IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, volume 1, pages 6–pp., 2006.

[6] M.G. Figueiro, M.S. Rea, A.C. Rea, and R.G. Stevens. Daylight and Productivity: A Field Study. In *ACEEE Summer Study 2002.*

[7] D. Rude. Why Do Daylight Harvesting Projects Succeed or Fail? *Construction Specifier*, 59(9):108, 2006.

[8] S. Stevens. Intelligent Façades: Occupant Control and Satisfaction. *International Journal of Sustainable Energy*, 21(2):147–160, 2001.

[9] E.S. Lee, S.E. Selkowitz, R.D. Clear, D.L. DiBartolomeo, J.H. Klems, L.L. Fernandes, G.J. Ward, V. Inkarojrit, and M. Yazdanian. Advancement of Electrochromic Windows. 2006.

[10] R.C. Willson and A.V. Mordvinov. Secular Total Solar

Irradiance Trend During Solar Cycles 21–23. *Geophys. Res. Lett*, 30(5):1199, 2003.

[11] D.H.W. Li and J.C. Lam. Evaluation of Lighting Performance in Office Buildings with Daylighting Controls. *Energy and Buildings*, 33(8):793–803, 2001.

[12] D.W.T. To, L.K. Sing, T.M. Chung, and C.S. Leung. Potential Energy Saving for a Side-lit Room Using Daylight-linked Fluorescent Lamp Installations. *Lighting Research & Technology*, 34(2):121, 2002.

[13] E.S. Lee, S.E. Selkowitz, G.D. Hughes, R.D. Clear, G. Ward, J. Mardaljevic, J. Lai, M.N. Inanici, and V. Inkarojrit. Daylighting the New York Times Headquarters Building. *Lawrence Berkeley National Laboratory, final report LBNL-57602*, 2005.

[14] J.A. Granderson. *Human-centered sensor-based Bayesian control: Increased energy efficiency and user satisfaction in commercial lighting*. PhD thesis, University of California, Berkeley, 2007.

[15] M.S. Rae. The IESNA Lighting Handbook–Reference and Application, Illuminating Society of North America, 2000.

[16] Y. Akashi and J. Neches. Detectability and Acceptability of Illuminance Reduction for Load Shedding. *Journal of the Illuminating Engineering Society of North America*, 33(1):3–13, 2004.

[17] G.R. Newsham, S. Mancini, and R.G. Marchand. Detection and Acceptance of Demand Responsive Lighting in Offices with and without Daylight. *Leukos*, 4(3):130–156, 2010.

[18] B. Plale, D. Gannon, J. Brotzge, K. Droegemeier, J. Kurose, D. McLaughlin, R. Wilhelmson, S. Graves, M. Ramamurthy, R.D. Clark, S. Yalda, D.A. Reed, E. Joseph, and V. Chandrasekar. CASA and LEAD: Adaptive Cyberinfrastructure for Real-time multiscale Weather Forecasting. *Computer*, 39(11):56–64, 2006.

[19] T. Gneiting and A.E. Raftery. Weather Forecasting with Ensemble Methods. *Science*, 310(5746):248, 2005.

[20] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy. Cloudy Computing: Leveraging Weather Forecasts in Energy Harvesting Sensor Systems. In *SECON '10: Proceedings of the 7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks*, pages 1–9. IEEE, 2010.

[21] W. Ji, C. Chan, J. Loh, F. Choo, and LH Chen. Solar radiation prediction using statistical approaches. In *IEEE*.

[22] C. Bergonzini, D. Brunelli, and L. Benini. Algorithms for Harvested Energy Prediction in Batteryless Wireless Sensor Networks. In *IWASI '09: Proceedings of the 3rd International Workshop on Advances in Sensors and Interfaces*, pages 144–149. IEEE, 2009.

[23] D. Noh, L. Wang, Y. Yang, H. Le, and T. Abdelzaher. Minimum Variance Energy Allocation for a Solar-Powered Sensor System. In *DCOSS '09: Proceedings of the 5th IEEE International Conference on Distributed Computing in Sensor Systems*, pages 44–57. Springer, 2009.

[24] K.W. Fan, Z. Zheng, and P. Sinha. Steady and Fair Rate Allocation for Rechargeable Sensors in Perpetual Sensor Networks. In *SenSys '08: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, pages 239–252. ACM, 2008.

[25] Lili Wang, Yong Yang, Dong Kun Noh, H.K. Le, Jie Liu, T.F. Abdelzaher, and M. Ward. AdaptSens: An Adaptive Data Collection and Storage Service for Solar-Powered Sensor Networks. In *RTSS '09: Proceedings of 30th IEEE Real-Time Systems Symposium*, pages 303 –312, 2009.

[26] C. Moser, D. Brunelli, L. Thiele, and L. Benini. Real-time Scheduling for Energy Harvesting Sensor Nodes. *Real-Time Systems*, 37(3):233–260, 2007.

[27] C. Moser. *Power Management in Energy Harvesting Embedded Systems*. PhD thesis, ETH Zurich, 2009.

[28] A. Kansal, J. Hsu, S. Zahedi, and M.B. Srivastava. Power Management in Energy Harvesting Sensor Networks. *ACM Transactions on Embedded Computing Systems (TECS)*, 6(4):32–es, 2007.

[29] C.M. Vigorito, D. Ganesan, and A.G. Barto. Adaptive Control of Duty Cycling in Energy-harvesting Wireless Sensor Networks. In *SECON '07: Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*.

[30] H. Suehrcke and PG McCormick. A Performance Prediction Method for Solar Energy Systems. *Solar Energy*, 48(3):169–175, 1992.

[31] V. Singhvi, A. Krause, C. Guestrin, J.H. Garrett Jr, and H.S. Matthews. Intelligent Light Control Using Sensor Networks. In *SenSys '05: Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, pages 218–229. ACM, 2005.

[32] H. Park, J. Burke, and M.B. Srivastava. Design and Implementation of a Wireless Sensor Network for Intelligent Light Control. In *IPSN '07: Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, pages 370–379. ACM, 2007.

[33] H. Park, J. Friedman, P. Gutierrez, V. Samanta, J. Burke, and M.B. Srivastava. Illumimote: Multimodal and High-Fidelity Light Sensor Module for Wireless Sensor Networks. *IEEE Sensors Journal*, 7(7):996–1003, 2007.

[34] G. Chen, M. Fojtik, D. Kim, D. Fick, J. Park, M. Seok, M.T. Chen, Z. Foo, D. Sylvester, and D. Blaauw. Millimeter-scale Nearly Perpetual Sensor System with Stacked Battery and Solar Cells. In *ISSCC '10: 2010 IEEE International Solid-State Circuits Conference Digest of Technical Papers*, pages 288–289. IEEE, 2010.