 doi:10.20944/preprints201808.0120.v1

Article

Multi-Objective Control of Air Conditioning Improves Cost, Comfort and System Energy Balance

Andrew Izawa[1] and Matthias Fripp[1]*

[1] Department of Electrical Engineering, University of Hawaii at Manoa, Honolulu, Hawaii, 96822
* Correspondence: mfripp@hawaii.edu; Tel.: +1-808-956-3795

Abstract: A new model predictive control (MPC) algorithm is used to select optimal air conditioning setpoints for a commercial office building, considering variable electricity prices, weather, occupancy and lighting. This algorithm, Cost-Comfort Particle Swarm Optimization (CCPSO), is the first to combine a realistic, smooth representation of occupants' willingness to pay for thermal comfort with a bottom-up, non-linear model of the building and air conditioning system under control. We find that using a quadratic preference function for temperature can yield solutions that are both more comfortable and lower-cost than previous work that used a "brick wall" preference function with no preference for further cooling within an allowed temperature band and infinite aversion to going outside the allowed band. Using historical pricing data for a summer month in Chicago, CCPSO provided a 3% reduction in costs vs. a "brick-wall" MPC approach with similar comfort and 13% reduction in costs vs. a standard night setback strategy. CCPSO also reduced peak-hours demand by 3% vs. the "brick-wall" strategy and 15% vs. standard night-setback. At the same time, the CCPSO strategy increased off-peak energy consumption by 15% vs. the "brick-wall" strategy. This may be valuable for power systems integrating large amounts of renewable power, which can otherwise become uneconomic due to saturation of demand during off-peak hours.

Keywords: HVAC model predictive control, demand response, EnergyPlus, particle swarm optimization (PSO), renewable energy, smart grids

MSC: 49M37, 65K05, 90-04, 90B35, 90B50, 90C29, 90C56, 90C90, 91B08, 91B10, 91B26, 91B42, 91B74

1. Introduction

In recent years, networked computation has become nearly ubiquitous, and power systems have begun relying more heavily on renewable power and market-based settling of supply and demand. These trends create both a need and an opportunity for dynamic pricing and demand response to help balance the power system. In a smart grid environment, price-responsive customers and devices can reschedule electricity loads from the times when electricity supply is scarce and production costs are high to times when supply is abundant and costs are low, thereby reducing bills and also improving the supply-demand balance for the power system as a whole [1–10].

Air conditioning systems in buildings provide a rich opportunity for demand response. Buildings account for 40% of all electricity consumed in the United States, with half that figure attributed to heating, ventilation and air conditioning (HVAC) [11,12]. There is an extensive literature on adjusting the timing of air conditioner operation in commercial buildings to reduce demand during high-cost times and increase it in low-cost times [13–23].

In most cases, price-responsive control algorithms seek to perform this shift while maintaining temperatures within a narrow band that preserves occupant comfort. This strategy uses the building's thermal mass as a form of thermal storage that has the same effect as an electric battery, without encroaching on the comfort of the occupants [17–23]. Among these algorithms, one of the most advanced, introduced by Corbin, Henze, and May-Ostendorp [24]. Their algorithm uses the Particle Swarm Algorithm [25] with a detailed thermal simulation in EnergyPlus [26] to perform model

 doi:10.20944/preprints201808.0120.v1

predictive control of a commercial office building HVAC system. Although the authors don't mention it, Corbin et al.'s algorithm represents a sort of "gold standard" for building control—it is able to directly optimize HVAC operation throughout the day to minimize costs under time-varying pricing, using state-of-the-art EnergyPlus building software to directly account for the complex, non-linear behavior of the coupled building and HVAC system. These factors are important in building energy demand, and include changing efficiency based on chiller and fan load and indoor/outdoor temperatures and humidity, switching modes between economizer and non-economizer operation, inefficient ranges due to reheat, and possible inability to reach setpoints under difficult conditions.

However, there is one significant shortcoming in the algorithms discussed above—they use "brick wall" representation of people's preferences for comfort, assuming the occupants are indifferent within a narrow range around an ideal temperature, but will not tolerate temperatures outside this range. This is equivalent to using a square well to represent user preferences, with infinite willingness to pay to move into the allowed band from outside, and zero willingness to pay to move closer to the ideal temperature once temperatures are within the allowed band. This produces solutions that generally peg the temperature at the lower edge of the allowed band during pre-cooling times, and the upper edge of the allowed band during expensive times. This likely does not match true occupant preferences, and foregoes opportunities to make occupants more comfortable when prices are low, and also to sacrifice some comfort for greater savings (moving slightly outside the standard comfort band) when prices are high.

A few previous studies have used more realistic functions to represent customer preferences [27–33]. These add a component to the cost calculation representing users' willingness to pay to move back to the ideal temperature. This component takes a few shapes: linear in absolute deviation from ideal (a triangular well)[29,32], or may have a "bathtub" shape (curved bottom with plateaus on either side)[28,30] or a quadratic curve with a minimum at the ideal temperature [27,33]. Unlike Corbin et al. [24] and the other brick-wall algorithms, these techniques are able to choose settings that increase user comfort when low costs justify it, or relax temperature settings further when customers would prefer that due to high cooling costs. This has the potential to produce solutions that give a better balance of cost vs. comfort, depending on each customer's preferences. They could also potentially provide more flexiblity to the power system, since they will give deeper demand reductions in times of scarcity and greater increases in times of abundance. However, all of the algorithms reported in this paragraph use simplified linear or quadratic models of the building's response to setpoints during their optimization, so they cannot model the complex, nonlinear behavior of the building. Consequently, although they can closely follow user preferences, they cannot optimize operation of the building itself as precisely as Corbin et al.'s PSO method [24].

In this study, we introduce a new model predictive control algorithm for commercial building air conditioners—cost-comfort particle swarm optimization (CCPSO)—that combines the best of both of these approaches: precise optimization via a detailed, nonlinear building model and a direct consideration of user preferences for savings vs. comfort. We test CCPSO against the "brick wall" PSO approach and standard night-setback strategies, and find that it can produce solutions that are superior in *both* cost and comfort. This approach also allows customers to choose a wider range of solutions than previously reported strategies, possibly sacrificing comfort to get lower costs than the established approaches, or vice versa. We also explicitly treat this as a multi-objective problem—building operators seek to minimize both cost and discomfort—and show the Pareto frontier of cost-comfort combinations that are available to customers with different preferences for cooling vs. cost-savings (also see [29]). Finally, we investigate for the first time the effect that a control strategy closely tuned to user preferences and building behavior could have on power demand during peak and off-peak hours. We find that incorporating more realistic user preferences could improve the system's ability to help balance time-varying supply and demand, an attribute that will be especially important in future, high-renewable power systems.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

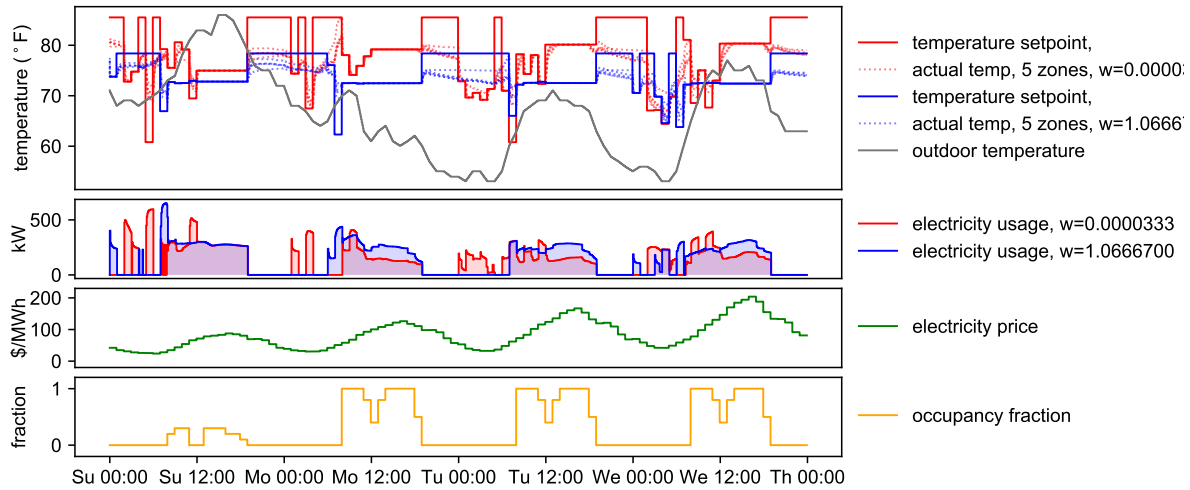

Figure 1. System operation with CCPSO on Wednesday 8/8/2012-Saturday 8/11/2012.

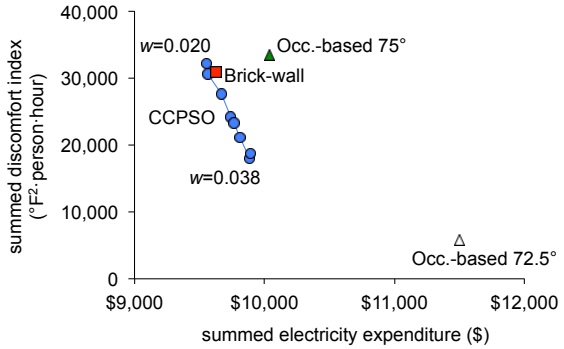

Figure 2. Pareto frontier for cost and comfort when optimizing $\mathbf{t}_{\text{set}}$ over August 2012. Blue points show optima from CCPSO for w values from 0.020 to 0.038. Other points show results from "brick-wall" algorithm with hard temperature boundaries and night setback strategy with fixed temperatures during occupied hours. CCPSO finds solutions that simultaneously reduce both discomfort and cost relative to other methods.

2. Results and Discussion

As discussed in section 3, CCPSO uses a penalty factor w to indicate preferences for cooling vs. cost-reductions. CCPSO applies this penalty, denoted in units of $\$/(°\text{F})^2/\text{person/hour}$, to the discomfort index each hour, in units of $(°\text{F})^2/\text{person/hour}$, and adds the resulting penalty to the objective function (cost function) that it seeks to minimize when choosing setpoints for each day ($t_{\text{set},d,h}$). The discomfort index is calculated based on the difference between the current temperature in each building zone and the ideal temperature for occupant comfort (t_{ideal}). Consequently, higher values of w drive the algorithm toward more comfortable solutions (closer to t_{ideal}), and lower values drive it toward less expensive solutions.

Fig. 1 shows four days of behavior for the Chicago office building model in which we tested CCPSO, for two different penalty factors w. CCPSO consistently takes advantage of low electricity prices to pre-chill the building early in the morning. Then temperatures are allowed to coast (using high $t_{\text{set},d,h}$) until occupants arrive at 8 am. During occupied hours, the controller maintains temperatures close to t_{ideal}. When CCPSO is used with a low penalty factor ($w = 0.020$), it chooses higher temperatures in the afternoon compared to CCPSO with a higher penalty factor ($w = 0.038$). This allows energy consumption to be reduced during the afternoon or during the pre-chilling period, or both. This effect is stronger on days with high power prices (Wed-Thu).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

Table 1. Cost of electricity and discomfort for customers with a range of discomfort penalty factors w, during August 2012.

Penalty factor w	**0.020**	**0.030**	**0.038**
Electricity cost ($)			
Night-setback 72.5°	11,502	11,502	11,502
Night-setback 75°	10,040	10,040	10,040
Brick-wall w/o term	9,642	9,642	9,642
Brick-wall w/ term	9,624	9,624	9,624
CCPSO	9,554	9,766	9,891
Discomfort cost ($)			
Night-setback 72.5°	118	174	222
Night-setback 75°	682	1,003	1,283
Brick-wall w/o term	619	911	1,166
Brick-wall w/ term	629	925	1,184
CCPSO	655	698	718
Total cost ($)			
Night setback 72.5°	11,620	11,675	11,724
Night setback 75°	10,722	11,042	11,323
Brick-wall w/o term	10,261	10,553	10,808
Brick-wall w/ term	10,253	10,549	10,808
CCPSO	10,209	10,464	10,609

Fig. 2 plots electricity expenditure and discomfort for values of the penalty factor w between 0.020 and 0.038 when using the CCPSO algorithm. Building managers can select any location along this Pareto frontier, trading off higher costs vs. higher comfort. The night-setback strategy with t_{base}=75°F is far from the optimal frontier, at (10,500, 32,000); all the CCPSO points shown in this figure have better comfort *and* cost than this strategy. Using t_{ideal} (72.5°F) produces an outcome with much higher electricity costs but also greater comfort (we would expect CCPSO to converge to this point as the penalty factor w is raised arbitrarily, but we didn't investigate this due to limited computing time).

The red marker in Fig. 2 shows the performance of the brick-wall strategy with a termination period like CCPSO (see section 3.1). This falls slightly above the frontier found with the CCPSO algorithm. Solutions along the shaded section of the CCPSO line dominate that found via the brick-wall strategy, producing both lower cost and lower discomfort. Solutions further to the right produce greater comfort than the brick-wall strategy, but at a higher cost; solutions to the left have lower comfort and lower cost. Although not shown here, we found that using a termination period with the brick-wall strategy had a negligible effect in this case study, shifting the solution slightly toward lower comfort and lower cost. The termination period might have greater importance if the planning period were shortened, so that effects from the first, binding day of the planning period continue to be felt during the termination period.

Table 1 compares the cost of operating according to the CCPSO algorithm, the brick-wall algorithms or the night-setback strategies, including both direct electricity costs and the financial value of discomfort. Results are tabulated for several different penalty factors w, corresponding to customers with varying preferences for comfort vs. financial savings. The night-setback strategy with t_{base}=72.5°F is the most comfortable, but also the most expensive. The related strategy with t_{base}=75°F provides financial savings but some extra discomfort. Both the brick-wall and CCPSO algorithms provide significant savings in electricity cost relative to the night-setback strategies, and the brick-wall algorithms provide similar comfort to the night-setback strategy with t_{base}=75°F.

The key difference between CCPSO and the brick-wall strategies is that CCPSO can make appropriate tradeoffs between cost and comfort, depending on customer preferences. For customers with the lowest willingness to pay for comfort (w=0.020), CCPSO identifies a strategy with slightly higher discomfort than the brick-wall strategies, but reduces electricity costs enough to compensate, resulting in net savings of 0.4%. For customers with stronger comfort preferences, CCPSO chooses

 doi:10.20944/preprints201808.0120.v1

Table 2. Quantity and timing of electricity used under several control strategies with similar comfort levels. CCPSO is most effective at reducing loads during system peaks and increasing them off-peak.

	Night-setback 75°F	Brick-wall w/term	CCPSO (w=0.023)
Avg. HVAC electrical load (kW)	154	164	166
Avg. load during 5% highest-price hours (kW)	318	279	271
Avg. load during 5% lowest-price hours (kW)	0	178	205
Avg. price for all power used ($/MWh)	$88	$79	$77

setpoints that produce greater comfort than the brick-wall strategies, but also higher electricity costs. The comfort improvements dominate, and for users with the strongest comfort preferences in Table 1 (w=0.038), CCPSO produces net savings of about 2% relative to the brick-wall strategies.

In addition to improving cost and comfort for the customer, CCPSO is intended to provide a price response that helps keep the power system balanced. Table 2 compares the size of price response from several HVAC control strategies that provide similar comfort. All three strategies use a similar total amount of electricity; however, CCPSO shifts consumption toward lower-price times, resulting in a lower cost per kWh used. Specifically, during hours with the top 5% highest prices, when the power supply is most constrained, CCPSO reduces demand by 15% compared to the night-setback strategy and 3% compared to the brick-wall algorithm. During times when power is most abundant, when electricity prices are in the bottom 5%, the CCPSO strategy increases demand by 15% relative to the brick-wall algorithm (the night-setback strategy uses no power during the lowest-priced hours). Increasing consumption during low-cost times may be especially important for enabling power systems to use more renewable power, as one of the key factors limiting adoption of wind and solar power is saturation of demand during sunny, windy and low-load times, which reduces the economic value of renewable power or even causes it to be discarded unused [34–39].

3. Methods

In this section, we provide give a detailed description of the CCPSO model and inputs. The complete input data, code and results used for this study are available in the Supplementary Information.

3.1. Optimization Problem

CCPSO selects temperature setpoints for a building HVAC system with the goal of maximizing net benefit to the building occupants—the value of comfort minus electricity costs. Setpoints are selected and evaluated for a sequence of future timesteps, in order to take advantage of opportunities for pre-cooling the building when electricity price are low and/or "coasting" when prices are high. At any timestep, the building fabric may be warmer or colder than the long-run average. This thermal energy—or lack of it—constitutes work deferred or stored by the HVAC system, which could result in higher or lower costs in later timesteps. It is not possible to directly assign a financial value to the energy stored in the building fabric, because it depends on electricity prices, occupancy and weather in later timesteps, as well as the strategy used to respond to those conditions. So CCPSO uses two methods to assess this value indirectly.

First, although we expect CCPSO to be run once per day or more often, the planning period extends more than 24 hours into the future. This forces the model to consider—and optimize—how energy in the building mass at the end of the first day will affect costs on later days.

Second, the planning period is followed by a standardized "termination period," with pre-specified setpoints, to assign a financial value to thermal energy in the building fabric at the end of the planning period. Without these extra days, the algorithm would be biased toward leaving the building underconditioned at the end of the planning period, since that would reduce costs during

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

Nomenclature

Optimization Model

δ_s	duration of timestep s (hours)
$t_{\text{act},d,s,z}(\mathbf{t}_{\text{set}})$	temperature in zone z during time step s on day d, if setpoints $\mathbf{t}_{\text{set}}$ are used
t_{ideal}	most-preferred temperature for building occupants (assumed to be 72.5°F)
$\mathbf{t}_{\text{set}}$	vector of n temperature setpoints for HVAC system for all time blocks $h \in H$ on days 1 to n_{p} (decision variables)
$b_{d,s,z}(\mathbf{t}_{\text{set}})$	dollar-denominated benefit in zone z during timestep s of day d if the system uses setpoints $\mathbf{t}_{\text{set}}$
$c_{d,s}(\mathbf{t}_{\text{set}})$	cost of purchasing electricity during timestep s of day d if the system uses setpoints $\mathbf{t}_{\text{set}}$
D	sequence of future days used for performance evaluation (1, 2, …, $n_{\text{p}} + n_{\text{t}}$)
$e_{d,s}(\mathbf{t}_{\text{set}})$	amount of electricity used (kWh) during timestep s of day d, if using setpoints $\mathbf{t}_{\text{set}}$
H	sequence of time blocks within each day, for which setpoints are chosen (00:00-01:00, 01:00-02:00, …, 11:00-12:00, 12:00-19:00, 19:00-24:00)
n	number of time steps in planning period ($n_{\text{p}} \times \lvert H \rvert$)
n_{p}	length of planning period (days): number of future days for which optimal setpoints are chosen (7 for this work)
n_{t}	length of termination period (days): additional days used for end-state evaluation (7 for this work)
$o_{d,s,z}$	building occupancy in zone z during time step s on day d (# people)
$r_{d,s}$	electricity price during timestep s of day d (\$/kWh)
S	sequence of time steps within each day, used for building simulation (00:00-00:05, 00:05-00:10, …, 23:55-24:00)
w	amount the building owner is willing to pay to avoid discomfort ($\$/(°\text{F})^2/\text{person/hour}$)
Z	set of building zones (N, S, E, W, Core)

Particle Swarm Procedure

$\mathbf{t}_p(k)$	temperature setpoints for particle p during iteration k
$\mathbf{b}_g(k)$	best-performing particle in full population during iteration k
$\mathbf{b}_p(k)$	best-performing version of particle p up through iteration k
$\mathbf{R}_1(k)$, $\mathbf{R}_2(k)$	diagonal matrices of random numbers, uniformly distributed on [0,1], regenerated each iteration
$\mathbf{v}_p(k)$	velocity of particle p during iteration k
a	acceleration constant (set to 1)
$c_0(k)$	inertial weight during iteration k
c_1	cognitive attraction weight (0.7)
c_2	social attraction weight (1.2)
k	iteration counter
$k_{\max}$	maximum allowed number of iterations (200)
$p \in P$	particle index
P	set of particles in the swarm $\{1, \ldots, 45\}$
$v_{\max}, v_{\min}$	upper and lower bounds on inertia (0.4, 0.9)

the planning period with no apparent negative impact on later days. The termination period also returns the building model to a standard state before evaluating other setpoints.

Previous work [24] has included the planning period but not the termination period. As will be discussed below, for our example building we found that cooling decisions on one day have a minimal effect on the state of the building one week later, so the termination period may not strongly influence decisions about the first day of operation. However, the termination period is also needed as part of the thermal modeling process (discussed further below), and performing cost and comfort assesment during this period may increase accuracy somewhat.

In concrete terms, the problem to be solved each day by CCPSO is given by

$$\max_{\mathbf{t}_{\text{set}} \in \mathbb{R}^n} \sum_{\substack{d,s,z \in \\ D \times S \times Z}} b_{d,s,z}(\mathbf{t}_{\text{set}}) - \sum_{d,s \in D \times S} c_{d,s}(\mathbf{t}_{\text{set}}) \tag{1}$$

To solve this problem, optimal setpoints $\mathbf{t}_{\text{set}}$ must be found for the planning period, timeblocks H on days 1 to n_{p}. The function $b_{d,s,z}(\mathbf{t}_{\text{set}})$ represents the benefit of the thermal comfort in the corresponding time and location if the system uses setpoints $\mathbf{t}_{\text{set}}$, i.e., the amount the building owner

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

would be willing to pay for this amount of cooling. The function $c_{d,s}(\mathbf{t}_{\text{set}})$ represents the cost of purchasing electricity to run the HVAC system with these setpoints. Below, we discuss the computation of these functions in more detail.

In order to reduce the dimensionality of $\mathbf{t}_{\text{set}}$ and achieve faster solutions, problem (1) includes two simplifications. The time blocks $h \in H$ are hourly from midnight to noon, then one block for 12:00-19:00 and one more for 19:00-24:00. This provides high resolution during pre-cooling and high-occupancy periods and a sparser representation during coasting and low-occupancy periods. In addition, rather than choosing different temperature setpoints for each zone, we choose a single setpoint for each time block and apply it to all zones. This representation reduces the problem size while still allowing detailed modeling of pre-chilling strategies. This creates a $|H| \cdot n_{\text{p}} = 98$ dimensional $\mathbf{t}_{\text{set}}$ similar to [24].

3.1.1. Economic demand for cooling

We are not aware of any literature describing building managers' demand for cooling services, so we introduce a simple, plausible form. First, we make the following assumptions:

1. if cooling were free, managers would cool to the temperature that makes the most occupants thermally neutral, t_{ideal} (assumed to be 72.5°F in this paper);
2. the amount of cooling that would be selected declines as the cost of cooling rises;
3. the relationship expressed in 2 is linear; and
4. the perceived benefit of cooling is proportional to building occupancy, measured in person·hours.

Assumption 1 reflects rational behavior; assumption 2 expresses the economic "law of demand"; assumptions 3 and 4 are hypothesized for the purpose of this paper. Below, we test a range of proportionality constants for assumption 3; future work could use field research to estimate both the form and coefficients for the demand function to improve on assumptions 3 and 4.

With these assumptions, we obtain a demand function for cooling during a given timestep:

$$p = 2wo\delta[c_{\text{ideal}} - c] \tag{2}$$

where p is the marginal cost of cooling the building ($/°F), 2 is introduced to help with integration later, w is a linear proportionality constant ($/(°F)^2/person·hour), c is the amount of cooling applied (°F), c_{ideal} is the amount of cooling needed to reach t_{ideal}, o is the building occupancy (# people) and δ is the duration of the timestep (hours). The standard interpretation of this demand function is that p is the amount that the manager would be willing to pay for one more unit of cooling, when already obtaining c units of cooling. So the total value that the manager places on c units of cooling can be calculated by integrating the demand function:

$$\begin{aligned} b(c) &= \int_0^c p(x)dx = \int_0^c 2wo\delta \cdot (c_{\text{ideal}} - x)dx \\ &= 2wo\delta \cdot (2cc_{\text{ideal}} - c^2) \\ &= -wo\delta \cdot (c_{\text{ideal}}^2 - 2cc_{\text{ideal}} + c^2) + wo\delta c_{\text{ideal}}^2 \\ &= wo\delta c_{\text{ideal}}^2 - wo\delta \cdot (c_{\text{ideal}} - c)^2 \\ &= wo\delta c_{\text{ideal}}^2 - wo\delta \cdot [(t_0 - t_{\text{ideal}}) - (t_0 - t_{\text{act}})]^2 \\ &= wo\delta c_{\text{ideal}}^2 - wo\delta \cdot (t_{\text{act}} - t_{\text{ideal}})^2 \end{aligned} \tag{3}$$

Here, we have used $c = t_0 - t_{\text{act}}$, where t_0 is the building temperature with no cooling and t_{act} is the actual temperature. For optimization purposes, we may ignore $wo\delta c_{\text{ideal}}^2$, the benefit of cooling to t_{ideal}, which is constant. This leaves a discomfort index that can be included in the optimization:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

$$o\delta \cdot (t_{\text{act}} - t_{\text{ideal}})^2 \tag{4}$$

With this discomfort index and a change of sign, (1) becomes

$$\begin{aligned} \min_{\mathbf{t}_{\text{set}} \in \mathbb{R}^n} & \sum_{d,s \in D \times S} c_{d,s}(\mathbf{t}_{\text{set}}) \\ & -w \sum_{d,s,z \in D \times S \times Z} o_{d,s,z} \delta_s \cdot (t_{\text{act},d,s,z}(\mathbf{t}_{\text{set}})) - t_{\text{ideal}})^2 \end{aligned} \tag{5}$$

Note that (5) can be interpreted as a multi-objective optimization problem minimizing discomfort and cost, where w is a weight or penalty charge used to reconcile the two objectives. By varying the value of w it is possible to choose an operating strategy anywhere on the Pareto frontier for these two objectives—trading off between minimum cost and minimum discomfort. The algorithm described here allows users to choose any point along this frontier.

3.1.2. Electricity cost

If forecasts of future electricity prices, building occupancy and weather conditions are available, then the cost of electricity can be calculated as $c_{d,s}(\mathbf{t}_{\text{set}}) = r_{d,s} e_{d,s}(\mathbf{t}_{\text{set}})$, where $r_{d,s}$ is the electricity price during timestep s of day d, and $e_{d,s}(\mathbf{t}_{\text{set}})$ is the amount of electricity used by the HVAC system during that timestep, in response to the setpoints $\mathbf{t}_{\text{set}}$. This gives the objective function:

$$\min_{\mathbf{t}_{\text{set}} \in \mathbb{R}^n} \sum_{d,s \in D \times S} r_{d,s} e_{d,s}(\mathbf{t}_{\text{set}}) - w \sum_{d,s,z \in D \times S \times Z} o_{d,s,z} \delta_s \cdot (t_{\text{act},d,s,z}(\mathbf{t}_{\text{set}})) - t_{\text{ideal}})^2 \tag{6}$$

Section 3.2 shows how we choose values of $\mathbf{t}_{\text{set}}$ to solve this optimization problem, Section 3.3 discusses how we compute $e_{d,s,z}$ and $t_{\text{act},d,s,z}$ from $\mathbf{t}_{\text{set}}$ and Section 3.4 discusses the specific data we used for this work. Section 2 presents our results.

3.2. Particle Swarm Optimization

Problem (6) is inherently non-convex. For example, turning off the cooling system and "coasting" during one time block (by assigning a high value to $t_{\text{set},d,h}$ for that period) may be a locally optimal strategy, but not globally optimal. Further, the most natural method for calculating cost and comfort due to the choice of $\mathbf{t}_{\text{set}}$—building simulation, as discussed in Section 3.3—does not yield derivatives for the objective function. Consequently, this problem must be solved via a derivative-free, global optimization method. For this work, we use particle swarm optimization (PSO) to do this.

As originally described by Kennedy, Eberhart and Shi [25], PSO is a meta-heuristic that progressively refines a population (swarm) of candidate solutions (particles) per iteration (generation); it is suited well for global optimization problems. Inspired by flocking movements of fish and birds, it makes few assumptions about the search space and does not require the problem to be differentiable.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

We use a PSO variant [40] that utilizes a logarithmically-decreasing inertial weight c_0 [41] inspired by findings in [42], alongside the usual cognitive and social attraction weights, c_1 and c_2:

$$\begin{aligned}\mathbf{v}_p(k+1) =& c_0(k)\mathbf{v}_p(k) \\ &+ c_1\mathbf{R}_1(k)[\mathbf{b}_p(k) - \mathbf{t}_p(k)] \\ &+ c_2\mathbf{R}_2(k)[\mathbf{b}_g(k) - \mathbf{t}_p(k)]\end{aligned} \tag{7}$$

$$\mathbf{t}_p(k+1) = \mathbf{t}_p(k) + \mathbf{v}_p(k+1) \tag{8}$$

$$c_0(k) = v_{\max} + (v_{\min} - v_{\max})\log_{10}\left(a + \frac{10k}{k_{\max}}\right) \tag{9}$$

Elements of this equation are described in the Nomenclature table, along with the specific values used for this work. The vector-valued quantities, $\mathbf{t}_p$, $\mathbf{v}_p$, $\mathbf{b}_p$ and $\mathbf{b}_g$ have one component for each time block on each day of the planning period.

Positions of all particles are updated once per generation k, calculated by adding fractions of individual previous velocities, and cognitive and social components. The cognitive component denotes a particle's memory of its personal best, $\mathbf{b}_p$, against its current position $\mathbf{t}_p$. Similarly, the social component denotes communication of the entire swarm, with comparisons against the swarm's current best position, $\mathbf{b}_g(k)$. Particles are evaluated by calculating the objective function of (6) with $\mathbf{t}_{\text{set}} = \mathbf{t}_p$, using EnergyPlus software as discussed in Section 3.3. Using inertia $c_0\mathbf{v}_p$, particles maintain some of their previous trajectory, which is noted to improve exploration properties [43]. Upon completion, $\mathbf{t}_{\text{set}}$ is set equal to the best solution in the final iteration k: $\mathbf{t}_{\text{set}} = \mathbf{b}_g(k)$

To accelerate convergence, the search space for $\mathbf{t}_{\text{set}}$ is bounded between 60°F and 90°F. If a particle hits the bounds of the search space, it is reflected back into the search space, as if the space was mirrored over the boundary.

To form the initial population, 44 particles are dispersed in the search space using a uniform, random distribution. An additional particle is initialized using a heuristic approach given by:

$$t_{45,\text{d,h}}(0) = \begin{cases} 66°\text{F}, h = \text{03:00-04:00 or 07:00-08:00} \\ 66°\text{F}, r_{d,h} < 50 \\ 90°\text{F}, r_{d,h} \geq 150 \\ 75°\text{F}, \text{otherwise} \end{cases}, \tag{10}$$

where $r_{d,h}$ is the mean value of $r_{d,s}$ for time steps that fall in block h on day d.

Attraction weights are selected to satisfy the following, ensuring stability [44]:

$$0 < (c_1 + c_2) < 4 \tag{11}$$

$$(c_1 + c_2)/2 - 1 < c_0 < 1. \tag{12}$$

Within these constraints, the coefficients are fine-tuned for faster convergence based on interactive testing.

Iteration continues until either $k = k_{\max}$, computation time reaches two hours, or the objective function improves by less than $15 over the course of 15 successive iterations.

3.3. Simulation Environment

3.3.1. EnergyPlus Software

To use the PSO algorithm discussed in Section 3.2, it is necessary to repeatedly calculate the functions $e_{d,s,z}(\mathbf{t}_p)$ and $t_{\text{act},d,s,z}(\mathbf{t}_p)$, which show the electricity usage and temperature if setpoints $\mathbf{t}_p$ are used for the building's HVAC system. To perform these computations, we use EnergyPlus software. EnergyPlus is a whole-building energy simulation engine that is widely used in academic and industry

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

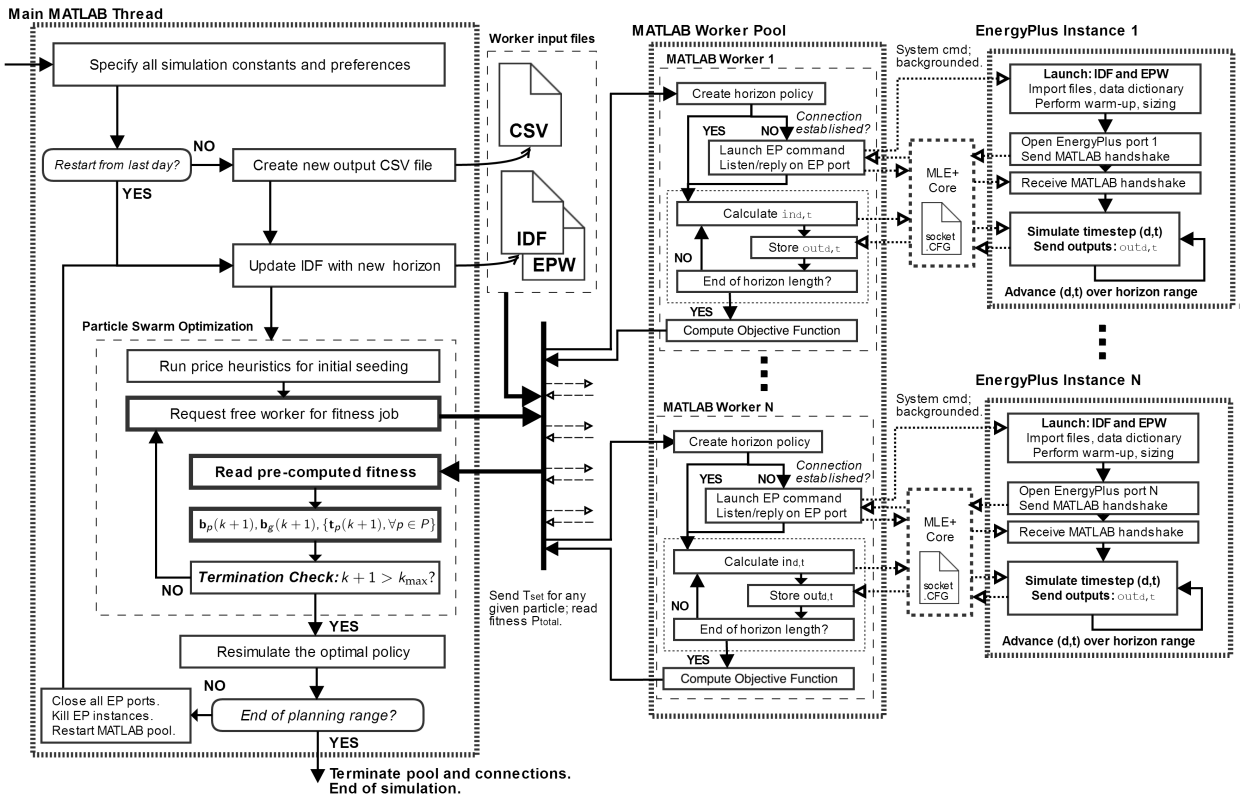

Figure 3. An overview of the optimizer architecture, which utilizes a co-simulation between MATLAB and EnergyPlus.

research to help design energy-efficient buildings. It can provide highly detailed simulations of the operation and effects of an HVAC system, incorporating factors such as building shape, materials and construction, internal and external gains, and the size of HVAC components.

During a simulation, EnergyPlus performs the following steps:

1. Read a data dictionary (provided with the software) which describes classes of components and their input fields.
2. Read a user-supplied input data file (IDF) which defines simulation parameters, building structure, component specifications, schedules, and desired outputs.
3. Optionally, read summer and winter design day (DDY) files and use these data to autosize components (used to adjust a generic building model to a particular climate zone).
4. Read a user-provided EnergyPlus weather (EPW) file which defines TMY3 weather.
5. Conduct a warm-up simulation, where the building is simulated repeatedly using the first day of weather and a single schedule for equipment settings; the warm-up period ends when the building model reaches steady state.
6. Simulate days and timesteps needed for study. For our work, this includes the following steps:
   (a) Simulate conditioning period (recent history of weather and setpoints)
   (b) Simulate planning period (predicted weather, using setpoints $\mathbf{t}_p$)
   (c) Simulate termination period (standardized conditions and setpoints)
7. Write results to output files.

For our work, the most important step is 6. In particular, steps 6b and 6c are used to calculate the objective function of our main optimization problem (6) for each particle during each iteration of PSO.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

3.3.2. Accelerated Computation

Repeatedly completing all the steps shown in the previous section would introduce significant computational overhead, since only steps 6b and 6c are needed to calculate $e_{d,s,z}$ and $t_{\text{act},d,s,z}$. Here we discuss several techniques we used to eliminate this overhead and accelerate the computation.

Co-Simulation

EnergyPlus supports a co-simulation environment through a Ptolemy II external interface, which is usually used to communicate with Building Controls Virtual Test Bed (BCVTB) middleware. The MLE+ Matlab toolbox uses this interface to provide a co-simulation environment controlled from Matlab [12]. Our model code, written in Matlab, uses MLE+ to communicate directly with an EnergyPlus simulation. This architecture is shown in Fig. 3. As noted in Fig. 3, the core of MLE+ handles the BCVTB packet protocol and establishes socket communication between EnergyPlus and a Matlab worker.

With this architecture, EnergyPlus performs steps 1–5 once, and then our Matlab code begins communicating with EnergyPlus once it reaches step 6. Then Matlab exchanges data with EnergyPlus (e.g., temperature setpoints and simulation results) for *each timestep*, rather than conducting the whole simulation in a single pass. With this architecture, it is not necessary for EnergyPlus to reread an IDF in order to incorporate changes to setpoints; formal output files are not required either. Thus, this approach only requires EnergyPlus to restart and run steps 1–5 when initializing the optimization for a new date range. This is shown in the outermost loop in Fig. 3, where the IDF template is written.

Use the Same Period for Termination and Conditioning

In order to obtain consistent and accurate results, it is necessary to simulate several days of the building's recent history (weather and HVAC setpoints) before each simulation of the planning and termination periods. This ensures that the state of the virtual building matches the current state of the real building before the new setpoints are evaluated. Simulating this "conditioning period" requires the same amount of computation as the planning or termination periods, since each period is one week long in our model. In principle, repeatedly computing the conditioning period could extend solution times by 50 percent. However, we reduce this overhead in our work by using the recent history (the previous 7 days) as the termination period. That is, we use historical weather, setpoints and electricity prices for the 7 previous days to define both the conditioning and termination periods (steps 6a and 6c). Consequently, immediately after the termination period, the virtual building is in the same state it would be after the conditioning period, so the co-simulation can return directly to the planning period. This technique uses the previous week to represent conditions that may prevail between 7 and 14 days in the future. Over the long term, this should be a nearly unbiased estimate. In addition, the termination period does not require high accuracy since it is only used to avoid bias at the end of the planning period, which has only a small effect on costs.

Checkpoint and Restore

It would be efficient to complete steps 1–6a once, then save the building's state, and then restore the building to this state during each iteration before simulating the planning and termination periods (steps 6b and 6c). However, EnergyPlus is not able to save the current state of a building and then resume simulation from that point. Consequently, we experimented with using Distributed MultiThreaded Checkpointing (DMTCP) software for this purpose [45]. DMTCP is usually used to take snapshots of long computations in a cluster environment to provide non-native fault tolerance. DMTCP can checkpoint processes on both sides of a communication socket, but not just one side (like EnergyPlus). So to use DMTCP, we created a Python-based wrapper for EnergyPlus, which normally forwards packets transparently between MATLAB and EnergyPlus. This wrapper watches for special packets from Matlab requesting creation and restarting of checkpoints. At this point, the wrapper script

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

closes the Matlab socket, checkpoints or restores itself and EnergyPlus, then opens a new connection with Matlab. This process usually restores EnergyPlus successfully, but it fails once every few thousand iterations. Consequently we did not use this technique in the model runs presented in this paper. Checkpoint-and-restore could nevertheless be a useful solution for similar models with extensive warm-up periods, especially if users want to use different periods for conditioning and termination.

3.4. Simulation Process

3.4.1. Simulation Parameters

Building Model

We tested our algorithm using the benchmark large Chicago office building included with the EnergyPlus installation. This building has exterior walls constructed with eight inches of concrete for the walls and interior floors, possessing a window-to-wall ratio of 38%. The orientation of the building is south, with an aspect ratio of 1.5. The glazing U-value is 3.24 $W/m^2/K$; the solar heat gain coefficient is 0.39. These properties indicate a building with high thermal mass, and follow ASHRAE 90.1-2004.

This particular building is modeled as 12 stories tall. Excluding the basement, each floor possesses five occupied zones. These zones are 2.74m tall, modeled with the following attributes:

Zones (Z):	(North, South, East, West, Core)
Zone volume:	$(860, 860, 554, 554, 6949)$ m^3
Zone interior mass:	$(626, 626, 404, 404, 5064)$ m^2
Occupancy (max):	1 person/18.58 m^2, $\forall z \in Z$
Lighting (max):	10.76 W/m^2, $\forall z \in Z$
Equipment (max):	4.71 W/m^2, $\forall z \in Z$
Infiltration:	0.000302 $m^3/s/m^2$, perimeter only

For all intermediate stories, a single five-zone floor is modeled, with a multiplier factor of 10. In total, there are 15 zones and 46,320 m^2 of conditioned space. Each floor is conditioned using a variable air volume (VAV) system with two large water-cooled chillers. A night cycle manager operates in 30 minute cycles and uses outdoor air to provide free cooling when appropriate. Primary heating and reheating coils are disabled, so that the air conditioning system can be set to "coast" by selecting a high $t_{\text{set},d,h}$; in this case, t_{act} floats until thermal equilibrium is reached. The fractional schedule in Fig. 1 shows occupancy relative to max, throughout the building. Lighting and equipment are assumed to follow the same fractional schedule as occupancy.

Length of Planning and Termination Periods

As discussed in Section 3.3, each iteration of our algorithm concludes by modeling building performance during a multi-day termination period. This period is used both to evaluate lingering effects of the setpoints chosen during the plannning period, and also to return the simulated building to a state that matches the current state of the real building. For both of these purposes, the termination period must be long enough to return the building to a standard state, which is not affected by any setpoints that were used before the termination period.

In order to choose the termination period length n_{t}, we conducted a test of the effect of thermal mass on deep cooling operation. For this test, we define a night-setback strategy as follows:

$$t_{\text{set},d,h} = t_{\text{occ},d,h}(t_{\text{base}}) \equiv \begin{cases} t_{\text{base}}, & o_{d,h,z} > 0 \\ 90°\text{F}, & o_{d,h,z} = 0 \end{cases} \tag{13}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

with

$$t_{\text{base}} = 75°\text{F} \tag{14}$$

The test is shown in Fig. 4 (also see [24]). After reaching steady state using the setpoints given by (13) and (14), an extreme chilling schedule (60°F) is used continuously until the end of Day 7. Power consumption decreases for several days after the building returns to normal operation; i.e., a normal level of comfort is maintained with less power input. By Day 14, the power consumption and zone temperatures return to within 1% of the steady state values. In this sense, the building "forgets" the extreme perturbation one week earlier. We conclude that the building's thermal state will be determined almost entirely by the weather and setpoints from the most recent 7 days, so we set $n_{\text{t}} = 7$ days. We also set the planning period n_{p} to 7 days, to ensure that this period includes all the days that could be affected by the binding decisions made for day 1.

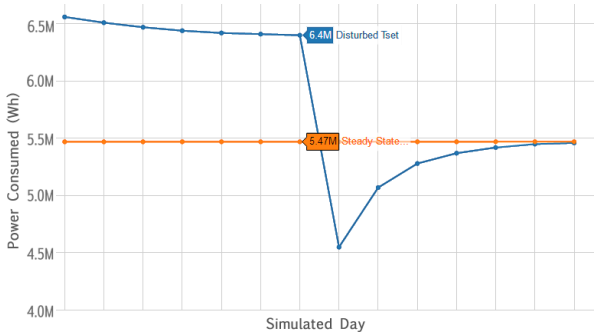

Figure 4. In the blue, an extreme schedule of $t_{\text{set},d,h} = 60°\text{F}$ for every timestep in the planning horizon (all $d \in D$ and $s \in S$). The default policy is to operate at 75°F for all occupied hours, which has its steady state power draw in orange.

Discomfort Penalty

As discussed in section 3.1, the w parameter can be set by the user to a suitable value to indicate their particular preference for low cost vs. high comfort. Each value of this parameter will drive the algorithm toward a different point on the Pareto frontier of cost vs. comfort. By testing several different values, it is possible to plot part of this frontier.

We are not aware of any literature on building managers' willingness to pay for comfort for building occupants, so we anchor our range by estimating the implicit value of w for a user who adopts the default strategy shown by (13) and (14). First, we assume that the user considers $t_{\text{base}} = 75°\text{F}$ to represent an optimal tradeoff between cost and comfort, among all strategies like (13). If (6) is

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

constrained to use the night-setback strategy given by (13), then at an optimal t_{base} we have (treating functions of $\mathbf{t}_{\text{set}}$ as functions of t_{base}):

$$\frac{d}{dt_{\text{base}}} E[c_{\text{elec}}(t_{\text{base}}) + wq(t_{\text{base}})] = 0, \text{ so} \tag{15}$$

$$\frac{dE[c_{\text{elec}}(t_{\text{base}})]}{dt_{\text{base}}} + w\frac{dE[q(t_{\text{base}})]}{dt_{\text{base}}} = 0, \text{ and} \tag{16}$$

$$w = -\frac{dE[c_{\text{elec}}(t_{\text{base}})]/dt_{\text{base}}}{dE[q(t_{\text{base}})]/dt_{\text{base}}} \tag{17}$$

$$\approx -\frac{\Delta_{t_{\text{base}}} E[c_{\text{elec}}(t_{\text{base}})]/\Delta t_{\text{base}}}{\Delta_{t_{\text{base}}} E[q(t_{\text{base}})]/\Delta t_{\text{base}}} = -\frac{\Delta_{t_{\text{base}}} E[c_{\text{elec}}(t_{\text{base}})]}{\Delta_{t_{\text{base}}} E[q(t_{\text{base}})]} \tag{18}$$

$$\approx -\frac{\begin{bmatrix} \sum_{d\in D}\sum_{s\in S}\sum_{z\in Z} r_{d,s} e_{d,s,z}(t_{\text{base}} + \delta_t) \\ -\sum_{d\in D}\sum_{s\in S}\sum_{z\in Z} r_{d,s} e_{d,s,z}(t_{\text{base}} - \delta_t) \end{bmatrix}}{\begin{bmatrix} \sum_{d\in D}\sum_{s\in S}\sum_{z\in Z} q_{d,s,z}(t_{\text{base}} + \delta_t) \\ -\sum_{d\in D}\sum_{s\in S}\sum_{z\in Z} q_{d,s,z}(t_{\text{base}} - \delta_t) \end{bmatrix}} \tag{19}$$

In equation (18) we approximate the derivatives by finite difference quotients for a small change in t_{base}. Then equation (19) estimates the change in the electricity-cost and discomfort functions in response to slightly higher and lower values of t_{base}. We evaluated (19) by simulating the entire study period (August 1–31), using $t_{\text{base}} = 75°\text{F}$ and $\delta_t = 0.5°\text{F}$. (We assumed $t_{\text{base}} = 75°\text{F}$ prior to August 1.) This produced an estimate of $w \approx 0.026$. This suggests a penalty on the order of \$0.01 for each (person $\cdot$ hour $\cdot \Delta°\text{F}^2$) from t_{ideal}, similar to [46]. We then identified several different preferences in a nearby range for further study:

$$W \equiv \left\{ \begin{array}{l} 0.020, 0.023, \\ 0.025, 0.028, \\ 0.030, 0.033, \\ 0.035, 0.038 \end{array} \right\} \frac{\$}{\text{person} \cdot \text{hour} \cdot °\text{F}^2} \tag{20}$$

3.4.2. Evaluation Process

We simulated use of our algorithm on a rolling basis over the period of August 1–31, 2012.

At the start of each simulated day, we retrieved power prices and weather for a 7-day planning period beginning on that day. We used historical weather data for Chicago from NOAA [47] and locational marginal power prices from the PJM Interconnection for corresponding days in 2007 (Fig. 5). Prices from 2007 were used to match [24], to which we compare our results. We also retrieved historical conditions for use in the conditioning and termination periods. For the historical period prior to August 1, 2012, we assume the building was operated with the night-setback policy given by (13) and (14). For this test, we assumed perfect foresight of weather and prices.

We then used the PSO algorithm discussed in Section 3.2 to choose optimal setpoints $\mathbf{t}_{\text{set}}$ for the coming planning period. Finally, we recorded the weather, setpoints, costs and discomfort from the *first day* of the planning period for later reference. The recorded values from the first day of each optimization represent the "actual" operation of the simulated system—they are used as the historical conditions for the termination/conditioning period, and are reviewed later to evaluate the performance of the system over the whole study period.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 6 August 2018 doi:10.20944/preprints201808.0120.v1

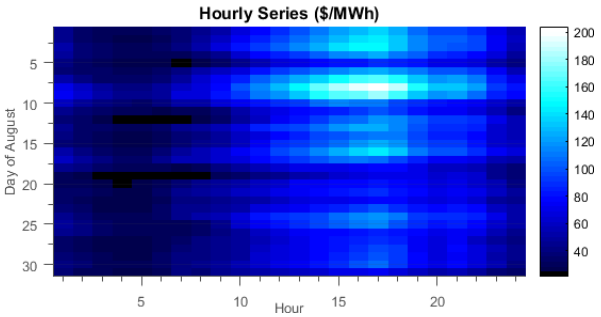

Figure 5. The hourly price series used for August 2012.

3.4.3. Benchmarks

Brick-Wall Algorithm

Ref. [24] uses an algorithm similar to ours, but uses "brick-wall" constraints on PMV. This algorithm requires that setpoints fall strictly within the range 71.6°F $\leq t_{\text{set},d,h} \leq$ 75°F during occupied hours, but doesn't assign any other cost to discomfort. Ref. [24] presents two versions of the brick-wall algorithm, one with a termination period, and one without. Their algorithm is otherwise identical to ours, aside from the performance enhancements noted in Section 3.3.2. To create benchmarks for our algorithm, we used each of the brick-wall algorithms to choose setpoints for August 1–31, then evaluated the total cost and discomfort that would occur over this period, using the discomfort equation (4).

Night-Setback Policy

In order to benchmark our algorithm against a simpler strategy, we also evaluated electricity cost and comfort when using the night-setback strategy given by (13), with two different values for t_{base}: 75°F (conservative) and 72.5° (comfort-seeking).

3.4.4. Computional Performance

Simulations were computed as separate single-node jobs, one for each value of w. Each month-long simulation used 15 Matlab workers, communicating with 15 copies of EnergyPlus via MLE+. The jobs ran on the University of Hawaii's Cray CS3000; each compute node utilized two 10-core Intel processors and 110GB of RAM on CentOS Linux. On average, a single planned day required 1.25 hours to compute.

4. Conclusions

This work introduces a new model-predictive control algorithm, CCPSO, which seeks to optimize air conditioner temperature setpoints throughout the day, responding to time-varying electricity prices, in order to minimize cost while also maintaining a satisfactory level of comfort. CCPSO differs from previous work by combining an explicit, continuous model of user preferences for comfort vs. cost savings with a detailed simulation of building thermal behavior via EnergyPlus software. The dual-objective (cost and comfort) CCPSO algorithm can be tuned to suit a variety of users, achieving the best combination of cost and comfort based on each building's occupants' preferences.

Compared to a standard night-setback strategy that achieves similar comfort (75°C during the day, off during the night), CCPSO reduced power costs by 15% during a month of cooling in Chicago. Compared to previous work using a PSO control algorithm with "brick-wall" constraints (no preference for further cooling within an allowed temperature band, and infinite aversion to going outside the allowed band), CCPSO reduced power costs by about 3%. Although these savings are modest, they

come at no additional cost and may be significant for a large building—$300 per month in this case. With the CCPSO algorithm, savings can also be increased or decreased—with a corresponding tradeoff in comfort—depending on users' comfort preferences.

Of particular interest for high-renewable power systems, we found that CCPSO is more responsive to dynamic prices than the previous approaches, providing a 3-15% reduction in load during system peaks and $\geq$15% increase in load during off-peak times. This response could improve utilization of efficient baseload plants in current-day power systems, and could help integrate renewable power in future power systems, where high and low prices may correspond to scarcity and abundance of renewable power. In this sense, CCPSO offers a "best of both worlds" approach, enabling building managers to find the best balance of cost and comfort, and at the same time improving the supply-demand balance in the rest of the power system, reducing costs for all users.

In future work, we plan to study opportunities for improved performance via enhanced building thermal memory, develop reduced-form models to reduce simulation time, incorporate uncertain forecasts, consider demand charges, and test the CCPSO algorithm in practice in large buildings.

Author Contributions: Conceptualization, M.F. and A.I.; Methodology, M.F. and A.I.; Software, M.F. and A.I.; Validation, A.I.; Formal Analysis, M.F. and A.I.; Investigation, A.I.; Resources, M.F.; Data Curation, M.F. and A.I.; Writing—Original Draft Preparation, A.I.; Writing—Review & Editing, M.F.; Visualization, M.F. and A.I.; Supervision, M.F.; Project Administration, M.F.; Funding Acquisition, M.F.

Funding: This research was funded by U.S. National Science Foundation award number 1310634.

Conflicts of Interest: The authors declare no conflict of interest. The funding sponsor had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

1. Kwon, S.; Ntaimo, L.; Gautam, N. Optimal Day-Ahead Power Procurement With Renewable Energy and Demand Response. *IEEE Transactions on Power Systems* **2017**, *32*, 3924–3933. doi:10.1109/TPWRS.2016.2643624.
2. Asensio, M.; Muñoz-Delgado, G.; Contreras, J. Bi-Level Approach to Distribution Network and Renewable Energy Expansion Planning Considering Demand Response. *IEEE Transactions on Power Systems* **2017**, *32*, 4298–4309. doi:10.1109/TPWRS.2017.2672798.
3. Park, L.; Jang, Y.; Cho, S.; Kim, J. Residential Demand Response for Renewable Energy Resources in Smart Grid Systems. *IEEE Transactions on Industrial Informatics* **2017**, *13*, 3165–3173. doi:10.1109/TII.2017.2704282.
4. Nikoobakht, A.; Aghaei, J.; Shafie-khah, M.; Catalão, J.P.S. Assessing Increased Flexibility of Energy Storage and Demand Response to Accommodate a High Penetration of Renewable Energy Sources. *IEEE Transactions on Sustainable Energy* **2018**, pp. 1–1. doi:10.1109/TSTE.2018.2843161.
5. do Prado, J.C.; Qiao, W. A Stochastic Decision-Making Model for an Electricity Retailer with Intermittent Renewable Energy and Short-Term Demand Response. *IEEE Transactions on Smart Grid* **2018**, pp. 1–1. doi:10.1109/TSG.2018.2805326.
6. Nezamoddini, N.; Wang, Y. Real-Time Electricity Pricing for Industrial Customers: Survey and Case Studies in the United States. *Applied Energy* **2017**, *195*, 1023–1037. doi:10.1016/j.apenergy.2017.03.102.
7. Andersen, L.M.; Hansen, L.G.; Jensen, C.L.; Wolak, F.A. Using Real-Time Pricing and Information Provision to Shift Intra-Day Electricity Consumption: Evidence from Denmark. p. 34.
8. Roldán Fernández, J.M.; Payán, M.B.; Santos, J.M.R.; García, A.L.T. The Voluntary Price for the Small Consumer: Real-Time Pricing in Spain. *Energy Policy* **2017**, *102*, 41–51. doi:10.1016/j.enpol.2016.11.040.
9. Talari, S.; Shafie-khah, M.; Osório, G.J.; Aghaei, J.; Catalão, J.a.P.S. Stochastic Modelling of Renewable Energy Sources from Operators' Point-of-View: A Survey. *Renewable and Sustainable Energy Reviews* **2018**, *81*, 1953–1965. doi:10.1016/j.rser.2017.06.006.
10. Elma, O.; Taşcıkaraoğlu, A.; Tahir İnce, A.; Selamoğulları, U.S. Implementation of a Dynamic Energy Management System Using Real Time Pricing and Local Renewable Energy Generation Forecasts. *Energy* **2017**, *134*, 206–220. doi:10.1016/j.energy.2017.06.011.

11. Shaikh, P.H.; others. A review on optimized control systems for building energy and comfort management of smart sustainable buildings. *Renewable and Sustainable Energy Reviews* **2014**, *34*, 409–429.
12. Bernal, W.; Behl, M.; Nghiem, T.X.; Mangharam, R. MLE+: A Tool for Integrated Design and Deployment of Energy Efficient Building Controls. *4th ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings, (BuildSys '12), Toronto, Canada.* **2012**.
13. Yoon, J.H. Demand Side Load Control in Residential Buildings with HVAC Controller for Demand Response. Thesis, 2015. doi:10.15781/T2862BH45.
14. Yoon, J.H.; Bladick, R.; Novoselac, A. Demand Response for Residential Buildings Based on Dynamic Price of Electricity. *Energy and Buildings* **2014**, *80*, 531–541. doi:10.1016/j.enbuild.2014.05.002.
15. Hammerstrom, D.J.; Ambrosio, R.; Carlon, T.A.; DeSteese, J.G.; Horst, G.R.; Kajfasz, R.; Kiesling, L.L.; Michie, P.; Pratt, R.G.; Yao, M. Pacific Northwest GridWise™ Testbed Demonstration Projects; Part I. Olympic Peninsula Project. Technical Report PNNL-17167, Pacific Northwest National Laboratory (PNNL), Richland, Washington, 2007.
16. Erickson, V.L.; Cerpa, A.E. Occupancy Based Demand Response HVAC Control Strategy. Proceedings of the 2Nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building; ACM: New York, NY, USA, 2010; BuildSys '10, pp. 7–12. doi:10.1145/1878431.1878434.
17. Corbin, C.D.; Henze, G.P.; May-Ostendorp, P. A Model Predictive Control Optimization Environment for Real-Time Commercial Building Application. *Journal of Building Performance Simulation* **2013**, *6*, 159–174. doi:10.1080/19401493.2011.648343.
18. Ilic, M.; Black, J.; Watz, J. Potential Benefits of Implementing Load Control. IEEE Power Engineering Society Winter Meeting, 2002, 2002, Vol. 1, pp. 177–182 vol.1. doi:10.1109/PESW.2002.984981.
19. Papaefthymiou, G.; Hasche, B.; Nabe, C. Potential of Heat Pumps for Demand Side Management and Wind Power Integration in the German Electricity Market. *IEEE Transactions on Sustainable Energy* **2012**, *3*, 636–642. doi:10.1109/TSTE.2012.2202132.
20. Tiptipakorn, S.; Lee, W.J. A Residential Consumer-Centered Load Control Strategy in Real-Time Electricity Pricing Environment. Power Symposium, 2007. NAPS '07. 39th North American, 2007, pp. 505–510. doi:10.1109/NAPS.2007.4402357.
21. Black, J.W.J.W. Integrating Demand into the U.S. Electric Power System : Technical, Economic, and Regulatory Frameworks for Responsive Load. Thesis, Massachusetts Institute of Technology, 2005.
22. Lee, Y.M.; Horesh, R.; Liberti, L. Optimal HVAC Control as Demand Response with On-Site Energy Storage and Generation System. *Energy Procedia* **2015**, *78*, 2106–2111. doi:10.1016/j.egypro.2015.11.253.
23. Ali, M.; Safdarian, A.; Lehtonen, M. Demand Response Potential of Residential HVAC Loads Considering Users Preferences. IEEE PES Innovative Smart Grid Technologies, Europe, 2014, pp. 1–6. doi:10.1109/ISGTEurope.2014.7028883.
24. Corbin, C.D.; Henze, G.P.; May-Ostendorp, P. A model predictive control optimization environment for real-time commercial building application. *Journal of Building Performance Simulation* **2013**, *6*, 159–174.
25. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. Proceedings of ICNN'95 - International Conference on Neural Networks; IEEE: Perth, Australia, 1995; Vol. 4, pp. 1942–1948 vol.4. doi:10.1109/ICNN.1995.488968.
26. Crawley, D.B.; Lawrie, L.K.; Winkelmann, F.C.; Buhl, W.F.; Huang, Y.J.; Pedersen, C.O.; Strand, R.K.; Liesen, R.J.; Fisher, D.E.; Witte, M.J.; Glazer, J. EnergyPlus: Creating a New-Generation Building Energy Simulation Program. *Energy and Buildings* **2001**, *33*, 319–331. doi:10.1016/S0378-7788(00)00114-6.
27. Leow, W.L.; Larson, R.C.; Kirtley, J.L. Occupancy-Moderated Zonal Space-Conditioning under a Demand-Driven Electricity Price. *Energy and Buildings* **2013**, *60*, 453–463. doi:10.1016/j.enbuild.2013.01.027.
28. West, S.R.; Ward, J.K.; Wall, J. Trial Results from a Model Predictive Control and Optimisation System for Commercial Building HVAC. *Energy and Buildings* **2014**, *72*, 271–279. doi:10.1016/j.enbuild.2013.12.037.
29. Hong, Y.Y.; Lin, J.K.; Wu, C.P.; Chuang, C.C. Multi-Objective Air-Conditioning Control Considering Fuzzy Parameters Using Immune Clonal Selection Programming. *IEEE Transactions on Smart Grid* **2012**, *3*, 1603–1610. doi:10.1109/TSG.2012.2210059.
30. Constantopoulos, P.; Schweppe, F.C.; Larson, R.C. ESTIA: A Real-Time Consumer Control Scheme for Space Conditioning Usage under Spot Electricity Pricing. *Computers & Operations Research* **1991**, *18*, 751–765. This preprint lacks graphs; they are in 1987 technical report and thesis., doi:10.1016/0305-0548(91)90013-H.

31. Livengood, D.; Larson, R. The Energy Box: Locally Automated Optimal Control of Residential Electricity Usage. *Service Science* **2009**, *1*, 1–16. doi:10.1287/serv.1.1.1.
32. Nguyen, D.T.; Le, L.B. Joint Optimization of Electric Vehicle and Home Energy Scheduling Considering User Comfort Preference. *IEEE Transactions on Smart Grid* **2014**, *5*, 188–199. doi:10.1109/TSG.2013.2274521.
33. Mady, A.E.D.; Provan, G.M.; Ryan, C.; Brown, K.N. Stochastic Model Predictive Controller for the Integration of Building Use and Temperature Regulation. Proceedings of the 25th AAAI Conference on Artificial Intelligence; Association for the Advancement of Artificial Inteligence: San Francisco, California, 2011.
34. Imelda.; Fripp, M.; Roberts, M.J. Variable Pricing and the Cost of Renewable Energy. Technical Report NBER Working Paper No. 24712, National Bureau of Economic Research, Cambridge, Massachusetts, 2018.
35. Fripp, M. Switch: A Planning Tool for Power Systems with Large Shares of Intermittent Renewable Energy. *Environmental Science & Technology* **2012**, *46*, 6371–6378. 11, doi:10.1021/es204645c.
36. Bird, L.; Cochran, J.; Wang, X. Wind and Solar Energy Curtailment: Experience and Practices in the United States. Technical Report NREL/TP-6A20-60983, National Renewable Energy Laboratory, Golden, Colorado, 2014.
37. Li, C.; Shi, H.; Cao, Y.; Wang, J.; Kuang, Y.; Tan, Y.; Wei, J. Comprehensive Review of Renewable Energy Curtailment and Avoidance: A Specific Example in China. *Renewable and Sustainable Energy Reviews* **2015**, *41*, 1067–1079. doi:10.1016/j.rser.2014.09.009.
38. Golden, R.; Paulos, B. Curtailment of Renewable Energy in California and Beyond. *The Electricity Journal* **2015**, *28*, 36–50. doi:10.1016/j.tej.2015.06.008.
39. Denholm, P.; O'Connell, M.; Brinkman, G.; Jorgenson, J. Overgeneration from Solar Energy in California. a Field Guide to the Duck Chart. Technical Report NREL/TP6A20-65023, National Renewable Energy Laboratory, Golden, Colorado, 2011.
40. Chen, S. Constrained Particle Swarm Optimization, ver 20130330. https://www.mathworks.com/matlabcentral/fileexchange/25986-constrained-particle-swarm-optimization, 2013.
41. Gao, Y.; An, X.; Liu, J. A Particle Swarm Optimization Algorithm with Logarithm Decreasing Inertia Weight and Chaos Mutation. 2008 International Conference on Computational Intelligence and Security, 2008, Vol. 1, pp. 61–65. doi:10.1109/CIS.2008.183.
42. Bansal, J.; others. Inertia weight strategies in particle swarm optimization. 2011 Third World Congress on Nature and Biologically Inspired Computing (NaBIC). IEEE, 2011, pp. 633–640.
43. Shi, Y.; Eberhart, R.C. Empirical study of particle swarm optimization. Proceedings of the 1999 Congress on Evolutionary Computation, 1999 (CEC 99). IEEE, 1999, Vol. 3.
44. Perez, R.; Behdinan, K. Particle swarm approach for structural design optimization. *Computers & Structures* **2007**, *85*, 1579–1588.
45. Ansel, J.; Arya, K.; Cooperman, G. DMTCP: Transparent Checkpointing for Cluster Computations and the Desktop. 23rd IEEE International Parallel and Distributed Processing Symposium; , 2009.
46. Pedersen, T.S.; others. Using heat pump energy storages in the power grid. Control Applications (CCA), 2011 IEEE International Conference on. IEEE, 2011, pp. 1106–1111.
47. EnergyPlus. EnergyPlus Weather Data by Location. https://energyplus.net/weather-location/north_and_central_america_wmo_region_4/USA/IL/USA_IL_Chicago-Midway.AP.725340_TMY3, 2015.