Title
HVAC system energy optimization using an adaptive hybrid metaheuristic

Journal
Energy and Buildings, 152

ISSN
03787788

Authors
Ghahramani, Ali
Karvigh, Simin Ahmadi
Becerik-Gerber, Burcin

Publication Date
2017-10-01

DOI
10.1016/j.enbuild.2017.07.053

HVAC SYSTEM ENERGY OPTIMIZATION USING AN ADAPTIVE HYBRID METAHEURISTIC

Ali Ghahramani [a], Simin Ahmadi Karvigh [b], Burcin Becerik-Gerber [c]

[a] PhD Student, Sonny Astani Dept. of Civil and Environmental Engineering, Viterbi School of Engineering, Univ. of Southern California, KAP 217, 3620 South Vermont Ave., Los Angeles, CA 90089-2531. E-mail: aghahram@usc.edu
[b] PhD Student, Sonny Astani Dept. of Civil and Environmental Engineering, Viterbi School of Engineering, Univ. of Southern California, KAP 217, 3620 South Vermont Ave., Los Angeles, CA 90089-2531. E-mail: ahmadika@usc.edu
[c] Associate Professor, Sonny Astani Dept. of Civil and Environmental Engineering, Viterbi School of Engineering, Univ. of Southern California, KAP 224C, 3620 South Vermont Ave., Los Angeles, CA 90089-2531. E-mail: becerik@usc.edu, Tel: +1 213 740 4383 (Corresponding Author)

ABSTRACT
Previous research efforts for optimizing energy usage of HVAC systems require either mathematical models of HVAC systems to be built or they require substantial historical operational data for learning the optimal operational settings. We introduce a model-free control policy that begins learning optimal settings with no prior historical data and optimizes HVAC operations. The control policy is an adaptive hybrid metaheuristic that uses real-time data, stored in building automation systems (e.g., gas/electricity consumption, weather, and occupancy), to find optimal setpoints at the building level and controls the setpoints accordingly. The algorithm consists of metaheuristic (k-nearest neighbor stochastic hill climbing), machine learning (regression decision tree), and self-tuning (recursive brute-force search) components. The control policy uses smart selection of daily setpoints as the control basis, making the control schema complementary to legacy building management systems. To evaluate, we used the DOE reference small office building in all U.S. climate zones and simulated different control policies using EnergyPlus. The proposed algorithm resulted in 31.17% energy savings compared to the baseline operations (22.5 °C and 3 K). The algorithm has a superior performance in all climate zones for the goodness of measure (i.e., normalized root mean square error) with a value of 0.047.

KEYWORDS:
HVAC system; Energy efficiency; Optimal control; Online learning; Setpoint optimization; Adaptive learning

1. INTRODUCTION

Buildings account for about 30% of the total energy consumption and greenhouse gas emissions in the world [1]. This share is expected to increase due to the rise in time spent in buildings [2]. The largest contributors to the energy consumption in buildings are Heating, Ventilation, and Air Conditioning (HVAC) systems (about 50% of the consumption in the developed countries [2]), which are responsible for providing thermal comfort and acceptable air quality. HVAC systems often use a single loop negative feedback control logic based on indoor air temperature to adjust the supply air flow and temperature entering a thermal zone [3]. HVAC controllers strive to keep the difference between a temperature setpoint and thermostat measurements below a threshold (i.e., deadband). HVAC systems provide minimum airflow rates to satisfy acceptable air quality (ASHRAE Standard 62.1 (Ventilation for Acceptable Indoor Air Quality) [4]), when the difference between the thermostat measurement and setpoint is smaller than the deadband. Although it has been shown adjusting HVAC system control

 https://doi.org/10.1016/j.enbuild.2017.07.053
https://escholarship.org/uc/item/1rk582x1

parameters (e.g., temperature setpoints) based on dynamic factors (e.g., weather variations) considerably reduces energy consumption [5], these control parameters often remain fixed due to the lack of methods for learning optimal parameters in practice. Control policies that allow for dynamic adjustment of temperature setpoints for improved energy efficiency can lead up to 37% energy savings depending on the climate, building size, and construction materials [5].

Control policies for improving HVAC energy efficiency can be categorized into two types: model-based, and data-driven. The model-based type of control policies require mathematical models to be built and they use building and HVAC system specifications to generate control rules. These control rules often impact the HVAC operations by modifying existing control loops or generating new ones. However, the mathematical models, used in this type of control policies, are often not readily available and might change over time during a building's lifecycle [6]. It should also be noted that model-based control policies often require modifications to control loops to form a feedback-based control logic, which makes them more difficult to be implemented in different types of HVAC systems. The data-driven type of policies impact HVAC systems operations by adjusting the control parameters (e.g., temperature setpoints and deadband) [5, 7-9]. These methods use historical operational data, in addition to the building data (e.g., occupancy) and environmental data (e.g., weather) to generate control rules. These policies can potentially work for any HVAC system type since they do not require a physical model and specifications of the HVAC system and also do not intervene existing HVAC system control logics. Thus, the review provided in this paper focuses on the control policies that are model-free and data-driven and are potentially capable of addressing occupants' dynamic thermal comfort requirements.

Data-driven control policies address the subjectivity of the model-based controllers by learning from historical operational data. Kusiak et al. [10] used a data-driven optimization method for reducing HVAC energy consumption in a typical office via adjusting control settings (i.e., supply air temperature and supply air static pressure). They tested eight supervised machine learning algorithms to model the nonlinear relationship among energy consumption, controlled settings, and uncontrolled variables. They found that the multiple-linear perceptron (MLP) ensemble provides the highest accuracy among the tested algorithms. Through applying a particle swarm optimization algorithm on the data driven models, they selected the control settings that minimize energy consumption and demonstrated 7% HVAC energy savings. Feldmeier and Paradiso [11] used a Proportional-Integral (PI) controller to adjust HVAC system setpoints, and demonstrated 24% energy savings compared to a baseline HVAC control system in a testbed building. In another study, Kusiak et al. [12] used multilayer perceptron algorithm with time-series to predict future performance of an HVAC system based on control settings and observed historical information. They then used an interior-point method for solving the optimization problem for optimal control parameters and realized 20.15% energy savings compared to the baseline. Brooks et al. [13] developed an occupancy-based controller for VAV HVAC systems. Experimental results demonstrated 29–80% energy savings in five rooms served by the VAV HVAC system. Because of the shared HVAC equipment, the rooms could not be independently conditioned. Ghahramani et al. [14] used historical performance of an HVAC system to create a zone airflow and setpoint function and used it to find optimal temperature setpoints through solving an optimization problem for energy, with comfort, indoor air quality, and system performance constraints taken into consideration. Compared to their earlier work focusing on an operational strategy for comfort solely, their proposed approach reduced energy use by about 12.08 % in three target zones while improving occupants' comfort.

Evolutionary algorithms are also vastly used to learn the optimal control parameters, using historical data. Kusiak in [15] used an evolutionary algorithm to find optimal control settings (i.e., supply air temperature and supply air static pressure) of an HVAC system based on a data-driven model built for system performance. The evolutionary algorithm used in this study (i.e., a modified SPEA-LS (Strength Pareto Evolutionary Algorithm with Local Search) can be categorized as a type of metaheuristics and the modeling technique used was multi-layer perceptron, which is a supervised machine learning technique.

The optimized setpoints, determined by solving the model, resulted in energy savings of 21.4%. Nishiguchi et al. [16] proposed a data-driven optimal control method for an HVAC system, using multivariate thin plate spline approximation for modeling the historical operational data and a basic gradient descend method, such as quasi-Newton's method to find the optimal control settings. In this method, model input variables are the variables influential to HVAC energy consumption, such as chilled water and cooling water temperatures and flow rate, and chilling load. Compared to the baseline operations, their method resulted in an average savings of approximately 9%. Fong et al. [17] developed a metaheuristic algorithm coupled with a simulation approach using evolutionary programming to adjust optimum operational settings for the chilled water and supply air temperature system in response to dynamic cooling loads and changing weather conditions throughout a year and proved their proposed approach could reduce energy consumption by about 7%.

In summary, all of the above-mentioned methods are implemented with fixed control parameters and models, which do not account for dynamic behavior (e.g., system performance decay or retrofit) of building systems. In addition, they require historical performance data in order to train the learning components, which might not be available or expensive to be collected. Obtaining the required historical data for optimal settings requires building systems to be operated by various operational settings under different indoor and outdoor conditions, which is a challenging task and might result in wasting energy and violating occupant comfort. In order to address these challenges, a data-driven optimal control policy, which simultaneously learns optimal control parameters from historical data and optimizes HVAC operations, is needed. .

In a previous study, we demonstrated that using daily selection of optimal temperature setpoints (a setpoint that minimizes energy consumption for each day) at the building level can potentially save between 7 to 37 % of energy depending on the climate, building size, and construction materials [5]. In this paper, we introduce a novel *adaptive hybrid metaheuristic algorithm* to find optimal setpoints at the building level in an online learning fashion (i.e., learning starts from the first day of HVAC operations). The proposed algorithm consists of a metaheuristic component (i.e., a k-nearest neighbor stochastic hill climbing), a machine learning component (i.e., a regression decision tree), and a self-tuning component (i.e., a recursive algorithm). Once the algorithm is implemented, the metaheuristic component begins searching the control parameter space (i.e., setpoints) for selecting more energy efficient setpoints through calculating a gradient based on similar historical data points. Meanwhile, the machine learning component fits a mathematical model to the neighborhood of similar historical data points and finds the local optimal in the neighborhood. In parallel to these two components, the self-tuning component recursively strives to improve the performance of the two other components through adjusting the hyper parameters. In order to evaluate the performance of our algorithm, we used the DOE small size office building reference model [18], which is an EnergyPlus software simulation file, and tested our algorithm in simulations of all 16 climate zones in the United States.

The paper is organized as follows. In Section 2, the adaptive hybrid metaheuristic algorithm, evaluation metrics, energy calculations, and the simulation models, as well as the procedures are presented. The results of implementing our methodology including the evaluation metrics are presented in section 3. Section 4 provides a discussion on how our algorithm can be used in a control policy to address occupants' thermal comfort requirements. The limitations and future steps of the study are presented in Section 5. Finally, section 6 provides a summary of the results and concludes the paper.

2. METHODOLOGY

Our *hybrid metaheuristic algorithm,* for learning building level optimal setpoints for achieving building level energy efficiency, consists of (1) a metaheuristic component (i.e., a k-nearest neighbor stochastic hill climbing optimization that performs a directed search for selecting energy efficient setpoints), (2) a machine learning component (i.e., a regression decision tree that finds the optimal setpoints given the

searched space), and (3) a self-tuning component (i.e., a recursive algorithm that searches for the optimal in the hyper parameters space of the metaheuristic and the machine learning components). The input to the algorithm is the energy consumption and dynamic independent variables, which can be used to predict the energy consumption such as outside air temperature and occupancy patterns. Based on the historical data, the metaheuristic component begins to search the control parameter space (i.e., setpoints) for selecting energy efficient setpoints by calculating a gradient based on the similar historical data. Once the historical data points in a neighborhood satisfy a threshold (described later in this section), the machine learning component fits a mathematical model to that neighborhood and finds the local optimal in the neighborhood. In parallel to these two components, the self-tuning component recursively optimizes the hyper parameters of the two other components to improve their performance. Figure 1 demonstrates the block diagram for the information flow of the proposed algorithm.

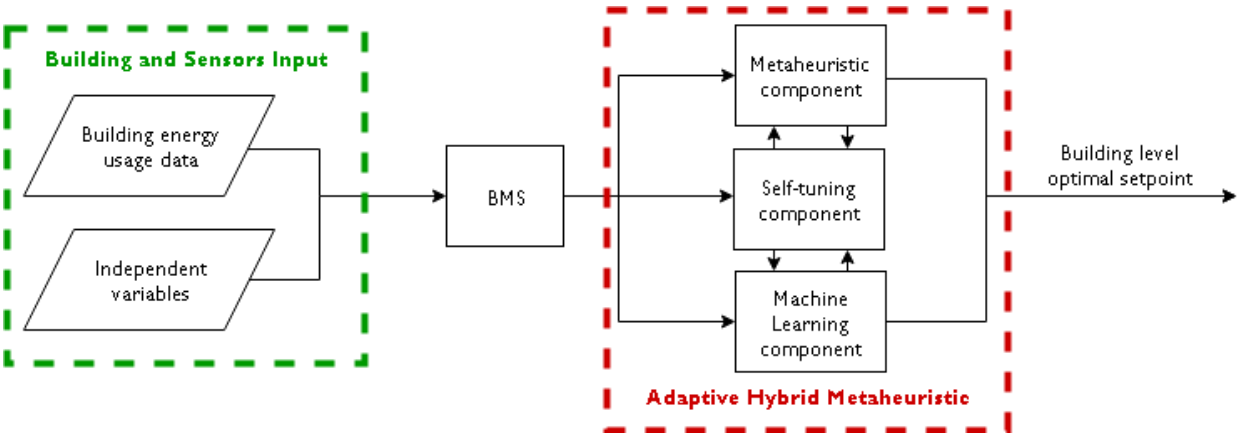


Figure 1. A block diagram for input streams and proposed algorithm components

Our metaheuristic is a modified stochastic hill climbing algorithm that selects energy efficient setpoints based on the observed energy consumption and independent variables. Stochastic hill climbing is a search algorithm in artificial intelligence domain, which is part of the family of local search [19]. It is an iterative algorithm that starts with an arbitrary solution to a problem, then attempts to find a better solution by incrementally changing the decision variable (i.e., temperature setpoint). If the change (having the absolute value of the step size) produces a better solution, an incremental change is made to the new solution. This process is repeated until no further improvements can be found. Generic stochastic hill climbing algorithm uses the single closest point to the point of interest as the reference for calculating the gradient. Due to the noise in the measurements from unforeseen conditions/unmeasured variables and the fact that the data is sparse (i.e., no value exists for a given combination of dimension values), we use k-nearest neighbor for reducing the effect of noise. The k-nearest neighbor algorithm utilizes Euclidean distance (L2-norm) for calculating the distances between data points and identifying the k closest neighbors. We used the majority rule for choosing the direction of the gradient. In other words, the gradient of the search direction is decided through a linear combination of the k-nearest neighbors (see Figure 2 for pseudocode of the metaheuristic component). The hyper parameters for the modified stochastic hill climbing are: hill climbing step size (s), and the number of neighbors (k). Hill climbing step size (s) is the magnitude of change in the decision variable (i.e., temperature setpoint). The number of neighbors (k) specifies the number of closest points to the point that a setpoint needs to be selected.

```
algorithm Hybrid metaheuristic is
    input:  Vector of data points (X), target functions (y)
            Acceptable range of decision variable (rv)
            Step size (s), Number of nearest neighbor (k), Machine learning distance (d)
            Point (x_j)
            In memory: decision variables (v), gradient directions (dv), estimated fitness (f)
    output: Decision variable (v_j) that minimizes y over rv for the point (x_j)
    neighbors ←—— k nearest neighbors of the x_j in X
    if max {distance of neighbors} < d
      dec ←—— ∑ (dv (neighbors with f > 0)) - ∑ (dv (neighbors with f < 0 ))
      propNeighbors ←—— neighbors with sign of dv(neighbor)* f(neighbor)) = sign(dec)
      base ←—— round(mean(v(propNeighbors)))
      if dec > 0, then dvj ←—— s
      elseif dec < 0, then dvj ←—— -s
      else dv_j ←—— randomly signed s
      end
      v_j ←—— bounded {base + dv_j} in rv
    else
      decTree ←—— decision tree on X and y
      vj ←—— decision variable in rv that minimizes decTree at x_j
    end
    Return v_j
(after implementing the v_j in the search at x_j)
    if measured y_j < mean (y(propNeighbors)), then f_j ←—— 1;
    else f_j ←—— -1;
    end
```

Figure 2. Metaheuristic (modified Stochastic Hill Climbing)

The machine learning technique that we use to enhance the learning process is the regression decision tree [20]. This algorithm is used to fit a model on the portion of the historical data that a decision needs to be made. By evaluating the model at different choices of the control parameters (i.e., temperature setpoints), we select the control parameter that minimizes the energy consumption. Regression decision tree (aka, regression tree) uses a tree-like graph or model, in which each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. This structure helps to map observations of a data point to the data point's target value (i.e., class label). In this study, we used a generic regression decision tree. To tune the internal parameters of the decision

tree, we used 10 fold cross validation. We chose not to present the details of the regression decision tree algorithm due to the space limitations and the fact that this algorithm is a well-established machine learning technique. The details on the internal structure of the algorithm can be found in [20]. The target value in our case is a continuous variable (i.e., energy consumption). Once a regression decision tree is fit on a close proximity of the point, in which a setpoint needs to be selected, the energy consumption for different setpoints are predicted using the regression decision tree. The setpoint that minimizes the energy consumption is then selected as the decision variable's value (i.e., setpoint) for the specific day. The component has a hyper parameter (i.e., the threshold mentioned earlier in this section) as the Euclidean distance (L2-norm) (d) for the proximity in which machine learning component should operate. All of the hyper parameters (i.e., hill climbing step size, number of neighbors, and machine learning operating distance) and the choice of setpoint for the first day of HVAC operation are the only arbitrary variables that that should be initialized as explained above. The choice of the setpoint for the first day has to be selected from an acceptable range, which is often set by the experts' knowledge. For example, if the outside temperature is relatively low (e.g., below 10 °C), the lowest setpoint in a range of thermally acceptable setpoints (e.g., 22°C) should be selected. If the setpoints are selected incorrectly, it might take a long time for the system to tune the parameters and reach an efficient performance.

The self-tuning component is responsible for tuning all of the hyper parameters a (Figure 3). As soon as the data on the BMS database begins to pile up, this component reclusively reviews the previous days' data and tests all different values of hyper parameters s, k, and d for the nearest neighbors of the prediction point (i.e., the optimal daily setpoint) and selects the hyper parameter combination that is expected to lead to the setpoint that minimizes the energy consumption. The minimization objective function is the L2 norm of the difference between the optimal setpoints calculated from the regression decision tree and the setpoints generated from different permutations of the hyper parameters. All of the hyper parameters indirectly impact each other. For example, the number of the neighbor's (k) to be considered for the target point uses L2 norm distance of points, which indirectly impact the distance (d) for the regression tree. Consequently, the recursive component exhaustively searches for the best combination of all of the factors for the prediction point. The pseudocode of the recursive algorithm can be found in Figure 3.

```
algorithm Self-tuning algorithm is
    input: Vector of data points (X), target functions (y)
           Acceptable range of decision variable (rv)
           Step size range(s), Number of nearest neighbor range (k), Machine learning distance range (d)
           Point (x_j)
           In memory: decision variables (v), gradient directions (dv), estimated fitness (f)
    output: Tuned hyper parameters (s_j, k_j, d_j) for the point x_j
    locTree ←— decision tree on the X and y
    for each k ∈ k
        neighbors ←— k nearest neighbors of the x_j in X
        for each neighbor ∈ neighbors
            vn ←— decision variable in rv that minimizes decTree at the neighbor
            for each d ∈ d
                for each s ∈ s
                    v_{k,n,d,s} ←— run hybrid metaheuristic algorithm for (X,y, rv, s, k, d, neighbor)
                    error_{k,n,d,s} ←— || v_{k,n,d,s} - v_n ||_2
                end
            end
        end
        error_{k,d,s} ←— mean (error_{k,n,d,s}) over n
    end
    Return k, d, and s that minimize the ERROR
```

Figure 3. Self-tuning process of the hybrid metaheuristic

Learning process starts from the first day of operations. Starting from a setpoint in the first day and observing the dependent variable (i.e., energy consumption), and independent dynamic variables (i.e., outside temperature), the algorithm selects a setpoint for the following day. In this study, we limit our investigations to only one dynamic factor (i.e., weather conditions represented as outside temperature) since outdoor temperature is an effective dynamic variable [21] and the DOE provides reference data for it. Other factors, such as occupancy variations, have not been added to the algorithm since there is no well-established occupancy pattern set for buildings in the literature. In addition, if the dynamic factor is a categorical variable, the algorithm can be applied to each category separately.

In summary, the modified stochastic hill climbing algorithm first finds k similar days based on the outdoor temperature using a k-nearest neighbor algorithm and sets a setpoint based on the stochastic gradient descent. Once enough data for outdoor temperatures and setpoints become available (i.e., max (distance of k nearest neighbors) < d), the machine learning component finds the optimal values in the

explored space. The self-tuning component searches all permutations of the hyper parameters at each day to find the set of hyper parameters that minimizes the energy consumption based on the historical model.

2.1. Evaluation Metrics and Process

To the best of our knowledge, there is no optimal control or optimization method, which provides the same functionality (i.e., adaptively searching, learning, and controlling based on the historical data in an online-learning manner) to compare with our algorithm for validation. Therefore, we compared the setpoints and energy consumption of the proposed control policy with the actual optimal values. The actual optimal setpoints and the associated energy consumptions were driven via a brute-force search using the simulations of all of the permutations of the setpoints for each day. In order to better understand the contributions of each component of the proposed algorithm to the learning performance, we calculated the setpoints and energy consumption of the proposed control policy at four levels: (1) generic metaheuristic (i.e., stochastic hill climbing), (2) k-nearest neighbor metaheuristic (i.e., k-nearest neighbor stochastic hill climbing), (3) hybrid metaheuristic (i.e., k-nearest neighbor stochastic hill climbing with machine learning), and (4) *adaptive hybrid metaheuristic* (i.e., the proposed algorithm, which can be described as self-tuning k-nearest neighbor stochastic hill climbing with machine learning). We carried out the setpoint comparison in terms of the goodness of fit of the setpoints at all four levels relative to the actual optimal setpoints over the year. The goodness of fit for the comparison of the setpoints was derived through calculating the normalized root mean square error between the setpoints of the control policies and the optimal setpoints that minimize energy consumption for each day over the year. Equation 1 describes the formula used for calculation of the goodness of fit.

$$\text{Goodness of fit } (\mathbf{setpoints}, \mathbf{optimalS}) = \frac{\|\mathbf{setpoints} - \mathbf{optimalS}\|_2}{\text{size } (\mathbf{optimalS}) * \text{mean}(\mathbf{optimalS})} \qquad \text{Equation 1}$$

where **setpoints** is the vector of the setpoints driven in the control policies, **optimalS** is the vector of actual optimal setpoints, and the size (**optimalS**) is number of elements in the **optimalS**.

The lower the values of the goodness of fit metric, the higher the performance of the algorithm in finding setpoints close to the actual optimal setpoints.

We also compared the energy savings for the four levels with the maximum saving driven by the brute-force search to assess the energy saving performance of each level. The energy consumption comparisons were performed by calculating the percentage of saving with respect to the energy consumption of the baseline settings (i.e., 22.5 °C setpoint). Accordingly, the savings would be compared to evaluate the performance of the four levels and the brute-force algorithm. Equation 2 demonstrates the formula for calculating the relative energy savings.

$$\text{Energy savings } (\mathbf{ens}, \mathbf{baselineE}) = \frac{\text{sum}(\mathbf{baselineE} - \mathbf{ens})}{\text{sum}(\mathbf{baselineE})} \qquad \text{Equation 2}$$

where **ens** represents the vector of energy consumptions associated with setpoints, **baselineE** is the vector of baseline setpoint setting energy consumptions.

In order to set up the energy simulation models, we first discretized temperature setpoints as the decision variable for the proposed algorithm since temperature setpoint is a continuous variable. A highly fragmented setpoint search space exponentially increases the computational cost by the order of parameters' space size. Consequently, we discretized the setpoints by assigning the granularity of 1 °C uniformly over the whole setpoint space. In addition, we selected decision variable range (i.e., setpoint range) to be 19.5 to 25.5 °C. Considering the fixed deadband of 3 K (default value for the DOE reference models), the resulting setpoints covered a wider range than the values used in different key studies in the

literature [18-21]. Office buildings account for the largest floor space (18%) and number (18%) of the commercial buildings in the United States [22]. 38% of the workers in commercial buildings work in office buildings [22]. Consequently, we focused on this type of building. We also set the simulation period to be one year as it covers the whole climatic variations for removing the bias. We then adjusted the temperature setpoints on the building energy model file (i.e., .idf file) and ran the simulation models via a programming language (i.e., MATLAB software). We searched the model's text file to locate the variable (e.g., temperature setpoint) and replace the desired values. We took the summation and average of the hourly energy consumption, and outside temperature output of the simulations, respectively, and represented them as single values for each day. We also excluded the first 28 days of the simulations due to the effects of simulation warm-up days [23]. Warm-up period is a period that EnergyPlus uses to tune and calibrate the internal model parameters.

To evaluate the applicability and study the performance of our *adaptive hybrid metaheuristic algorithm*, we used the small office building built after 2004, as the dynamic adjustments of the control parameter require a BMS, which is more likely to be available in the buildings built after this date. The small office is a 1 floor building with five thermal zones (5 thermostats controlling the temperatures). Total floor area is 511m^2. Aspect ratio is 1.5. Floor-to-floor height is 3.05 m. Glazing fraction is 0.21. Construction materials and equipment specifications were set based on ASHRAE standards. ASHRAE Standards 90.1-2004 (ASHRAE 2004a), and 62.1-2004 (ASHRAE 2004b).Roof construction is insulation entirely above deck. The attic roof with wood joist is built with roof insulation and 1.6 cm gypsum board. Wall construction is steel frame. Exterior walls are wood-frame walls (2X4 40sm OC) which have 2.5cm stucco and 1.6cm gypsum board with wall Insulation and 1.6cm gypsum board. Heating equipment is furnace and cooling equipment is PACU (Packaged Air Conditioning Unit). Air Distribution equipment is SZ CAV (Single-Zone Constant Air Volume). The occupancy used in the model is 18.6 m^2/person. Figure 4 demonstrates building geometry of the small office building model.

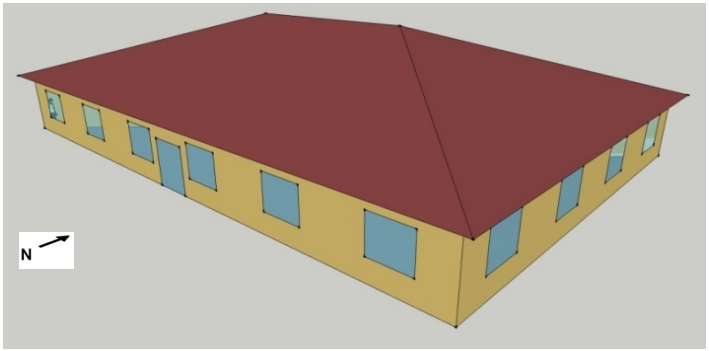


Figure 4. Building geometries of small office building simulation model

The cities studied in the paper were Miami, Florida (1A), Houston, Texas (2A), Phoenix, Arizona (2B), Atlanta, Georgia (3A), Los Angeles, California (3B), Las Vegas, Nevada (3B), San Francisco, California (3C), Baltimore, Maryland (4A), Albuquerque, New Mexico (4B), Seattle, Washington (4C), Chicago, Illinois (5A), Denver, Colorado (5B), Minneapolis, Minnesota (6A), Helena, Montana (6B), Duluth, Minnesota (7), Fairbanks, Alaska (8), covering all of the climatic zones in the U.S. The cities are most populated cities in each climate zone as presented in Figure 5. Climate 1 represents the hottest, and climate 16 represents the coldest.

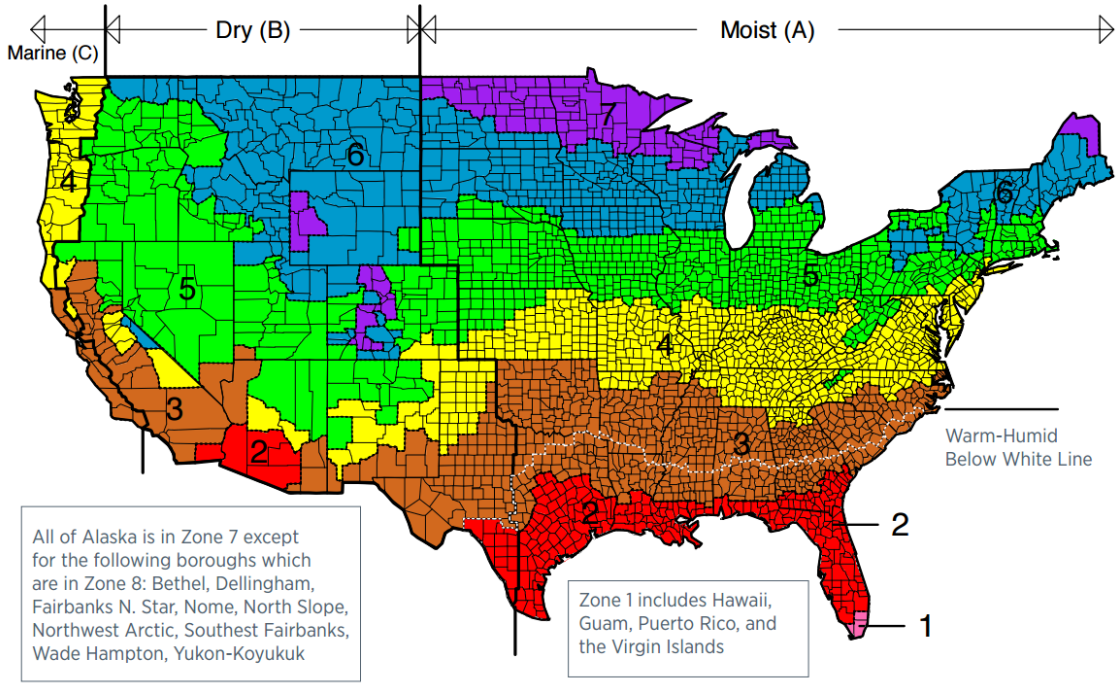


Figure 5. Climate zone classification ([24])

The occupancy schedules in reference building models were assigned in terms of the HVAC operations as weekdays (from 6:00 AM to 10:00 PM), Saturdays (from 9:00 AM to 5:00 PM), and Sundays and holidays (off). Since occupancy and heat loads have a direct impact on the energy consumption, we only used the weekdays to have a uniform occupancy impact. More information on office buildings features and specifications can be found in [18, 25]. We stored the simulation results as a CSV file and calculated the energy usage for each day by summing the default hourly simulation outputs. The simulation settings, internal, and external variables (e.g., setpoint, outside temperature, city, day of year) were also stored as a vector of features.

3. RESULTS AND DISCUSSION

First, we present the implementation steps of our algorithm, using the DOE reference small size office building, located in climate zone 5A (Chicago, Illinois) in order to better explain the algorithm. We selected the climate zone 5A due to the fact that the weather variations in this climate zone include both hot and cold conditions (daily average outside temperature varies between -11 to 30 °C in the reference climate files). Figure 6 presents the whole building energy consumption, including the electricity and gas used by the building systems (e.g., lighting systems and the HVAC system), as well as only by the HVAC system (the baseline control parameters of 22.5 °C for the setpoint and 3 K for the deadband) for the entire year, which include weekdays, Saturdays, Sundays and holidays).

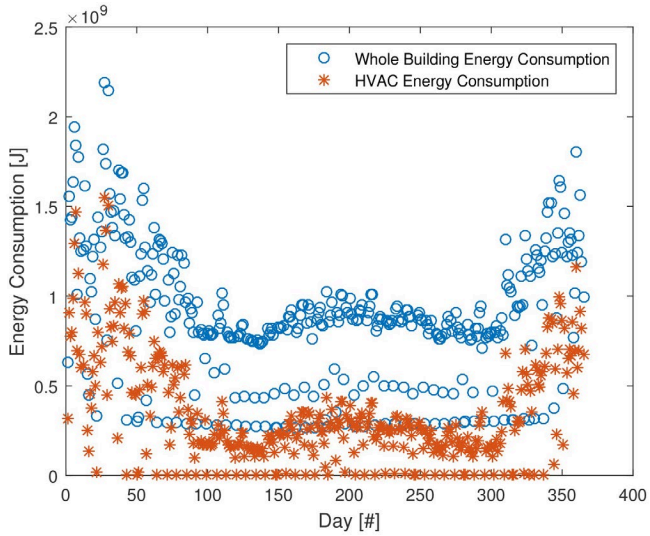
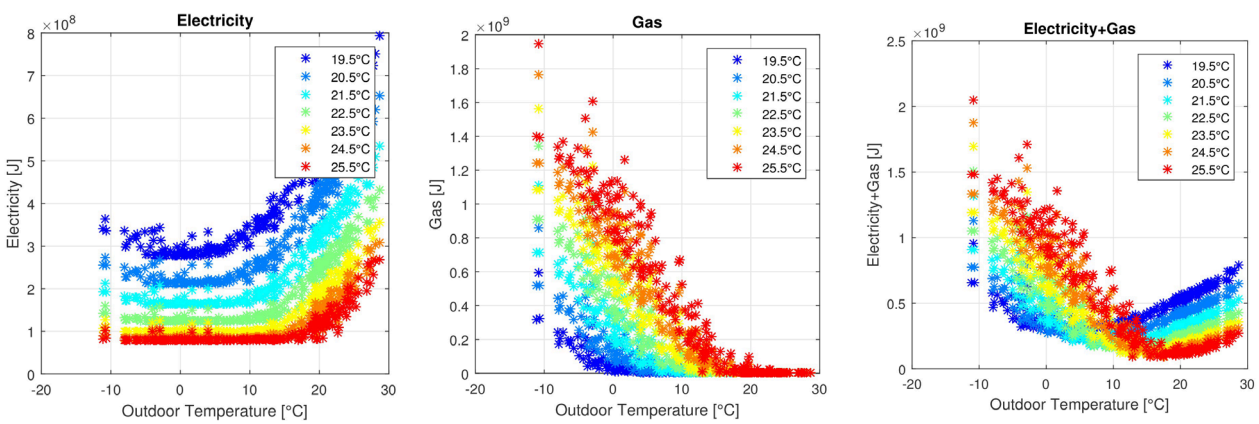


Figure 6. HVAC system and whole building energy consumption for the small building built after 2004 in Chicago, Illinois.

Figure 7 demonstrates the weekdays HVAC energy consumption (electricity, gas, and both) for the range of setpoints with respect to the average daily outdoor temperatures. As it can be seen in Figure 7c, in low outdoor temperature the highest setpoint consumed the lowest energy, while in the high outdoor temperature the lowest setpoint consumed the lowest energy. As the outdoor temperature increases, we observe a transition in the lowest energy consuming setpoints from high setpoints to low setpoints. As explained earlier, we use this fact to design a control policy that strives to select the optimal daily setpoint with respect to the dynamic building factors, such as the outdoor temperature.



a b c

Figure 7. Electricity, gas, and total energy consumption of the HVAC system in the small size building (new construction, built after 2004) in Chicago, Illinois.

Through simulating all permutations of control setpoints at each day and applying a brute-force search, we derived the actual optimal setpoint and the associated energy consumption for each day (called 'brute-force optimal setpoints' in Figure 8) and stored them as the validation metric for the adaptive hybrid metaheuristic algorithm. Figure 8 presents the optimal setpoints derived from the brute-force

search (blue circle signs) and the setpoints derived from the two components (i.e., k-nearest stochastic hill climbing and regression tree) of the proposed adaptive hybrid metaheuristic algorithm on a daily basis over the entire year. The self-tuning component only improves the performance of the two other components, and thus does not directly output optimal setpoints. As it can be seen, the algorithm starts searching the space via the k-nearest neighbor stochastic hill climbing component (red cross signs) and once enough data points in a neighborhood are collected, the regression decision tree component (black star signs) fits a mathematical model to the neighborhood and finds the local optimal setpoints. Consequently, the performance of the algorithm improves as more data points are collected through both utilization of regression decision tree component and the self-tuning component (via improving the internal hyper parameters selection).

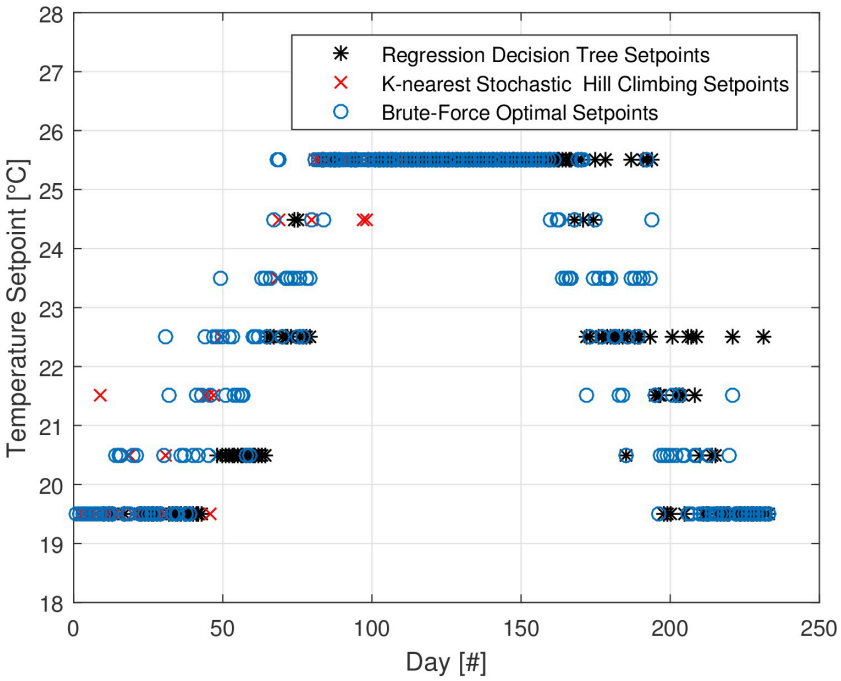


Figure 8. Setpoints from the implementation of different components of our adaptive hybrid metaheuristic algorithm

To better understand the performance of the proposed algorithm, we compare the optimal setpoint curve obtained from the *adaptive hybrid metaheuristic algorithm* with the brute-force optimal setpoints and also exhaustive optimal setpoints are derived through fitting a regression decision tree to the brute-force data to generate a function that maps the outdoor temperature to the optimal setpoints. Figure 9 presents the setpoints derived from the three methods. As it can be seen, there are multiple optimal setpoints at a given outdoor temperature calculated by the brute-force search. This happens because of the impact of other influential factors, such as outdoor humidity and sun radiation. Therefore, the brute-force search's results are not mathematically a function of the outdoor temperature only, since multiple optimal setpoints for similar outdoor temperatures could exist. The exhaustive optimal setpoints' curve, which provides a single optimal setpoint for each outside temperature, does not exactly match with the hybrid metaheuristics optimal setpoints' curve because the adaptive hybrid metaheuristic algorithm was not trained with all permutations of all the variables. Consequently, the energy savings from the exhaustive optimal setpoints is below the brute-force search optimal setpoints and above the adaptive hybrid metaheuristic. Later in this section, we demonstrate the energy savings from all of these algorithms in

different climates. It should be also noted that the maximum possible saving from adaptive hybrid metaheuristic after being trained by all permutations of the variables is the exhaustive optimal setpoints savings.

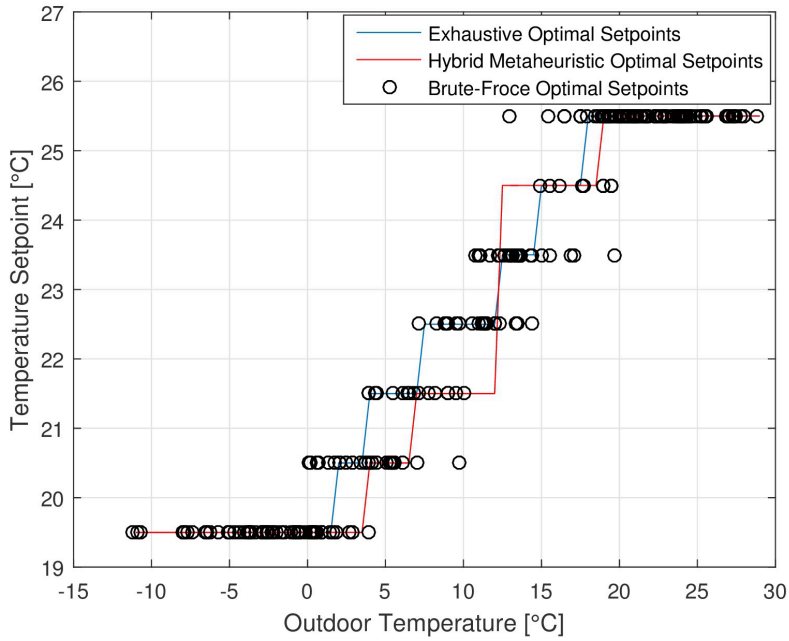


Figure 9. Optimal setpoints as a function of outside temperature

As explained in Section 3, we used four levels of implementation in the evaluation of the proposed algorithm. Although the *adaptive hybrid metaheuristic algorithm* utilizes the self-tuning mechanism to further improve its accuracy, other levels do not have the self-tuning component and hyper parameters are constant. In order to determine the hyper-parameters, we tested different values in an adequate range for each of the hyper parameters for finding values that minimize the energy consumption over the entire year for all of the climates. It should be noted that we used a whole-year simulation for each value of the hyper parameter, which was computationally expensive and potentially impractical for real buildings. The ranges were step size range of 1 to 3 °C, with 1 °C increment, k nearest neighbor of 5 to 9 with increment of 2, the machine learning distance of 2 to 3.5 with increment of 0.5.Accordingly, the first level (i.e., generic metaheuristic algorithm) was implemented with a step size of 2 °C. The second level (k-nearest neighbor metaheuristic algorithm) was implemented with a step size of 2 °C and k-nearest neighbor of 5. The third level (hybrid metaheuristic algorithm) was implemented with a step size of 2 °C, k-nearest neighbor of 5, and the machine learning distance of 3. The learning process of the adaptive hybrid metaheuristic algorithm starts its tuning process with the same hyper parameters as the hybrid metaheuristic algorithm does (a step size of 2 °C, k-nearest neighbor of 5, and the machine learning distance of 3). However, the self-tuning process searches the step size range of 1 to 3 °C, with 1 °C increment, k nearest neighbor of 5 to 9 with increment of 2, the machine learning distance of 2 to 3.5 with increment of 0.5 to reassign hyper parameters.

Figure 10 presents the daily energy consumption difference between the four levels of adaptive hybrid metaheuristic algorithm and the actual optimal values driven from brute-force search for the building in climate zone 5A. As it can be seen, the adaptive hybrid metaheuristic algorithm has the relatively lowest daily energy consumption difference compared to the brute-force search among all four levels. As explained earlier, the fact that dynamic adjusting of the hyper parameters in accordance with

the dynamic factors (e.g., outdoor temperature) allows for more savings. Compared to the other levels, as it can be seen in Figure 10, our adaptive hybrid metaheuristic algorithm results in more energy use reduction in both high and low temperatures. It is due to the fact that the self-tuning component is the cause for the improved performance at different values of outdoor air temperature. Other levels had fixed hyper parameters that were tuned for the entire year (as opposed to daily tuning) and consequently, have a variation in energy consumption difference over the year. As it can be seen, there is a relatively lower daily energy consumption difference in higher temperatures, but large daily energy consumption difference in lower temperatures

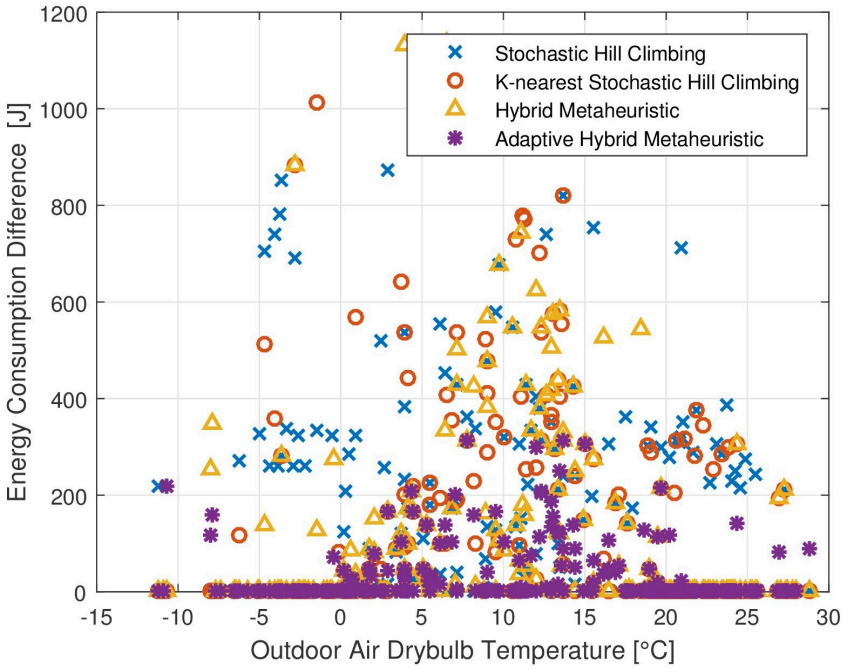


Figure 10. Energy consumption difference for different levels of the proposed adaptive hybrid metaheuristic relative to the optimal brute-force search algorithm

In order to evaluate the performance of our adaptive hybrid metaheuristic algorithm, we carried out a goodness of fit analysis for the setpoints relative to actual optimal values driven from the brute-force search optimal setpoints over a year for four different levels (i.e., generic metaheuristic, k-nearest neighbor metaheuristic, hybrid metaheuristic, and adaptive hybrid metaheuristic) of the algorithm. Figure 11 demonstrates the goodness of fit (by calculating the normalized root mean square error) of setpoints driven from the different levels of implementation of the proposed algorithm to the brute-force search optimal setpoints for each day in all of the U.S. climate zones. In average, the goodness of fit for the generic stochastic hill climbing, the k-nearest stochastic hill climbing, hybrid metaheuristic, and the adaptive hybrid metaheuristic algorithms were 0.098, 0.082, 0.0648, and, 0.0396, respectively. The lower values of goodness of fit metric represents higher accuracy of fit. Accordingly, the monotonic improvements of goodness of fit for different levels demonstrate that the level of implementation helped to get close to the brute-force search optimal setpoints.

However, as it can be seen in Figure 11, the performance of the generic stochastic hill climbing algorithm, k-nearest stochastic hill climbing, and hybrid metaheuristic algorithm do not follow a similar trend (i.e., at each climate the goodness of fit values for algorithms are ordered similar to the average of climates case), except for the adaptive hybrid metaheuristic algorithm, which showed a superior

performance in all climate zones with a negligible difference in Phoenix. In addition, the goodness of fit metric of the adaptive hybrid metaheuristic algorithm was higher for relatively colder climates compared to the other levels (from about 0.001 difference in goodness of measure in hot climates to almost 0.02 difference in cold climates). Considering the fact that the major difference between the adaptive hybrid metaheuristic algorithm and other three levels is the self-tuning component, dynamic adjustments of the hyper parameters in response to outdoor temperature variations is the reason of the variation in the goodness of fit values.

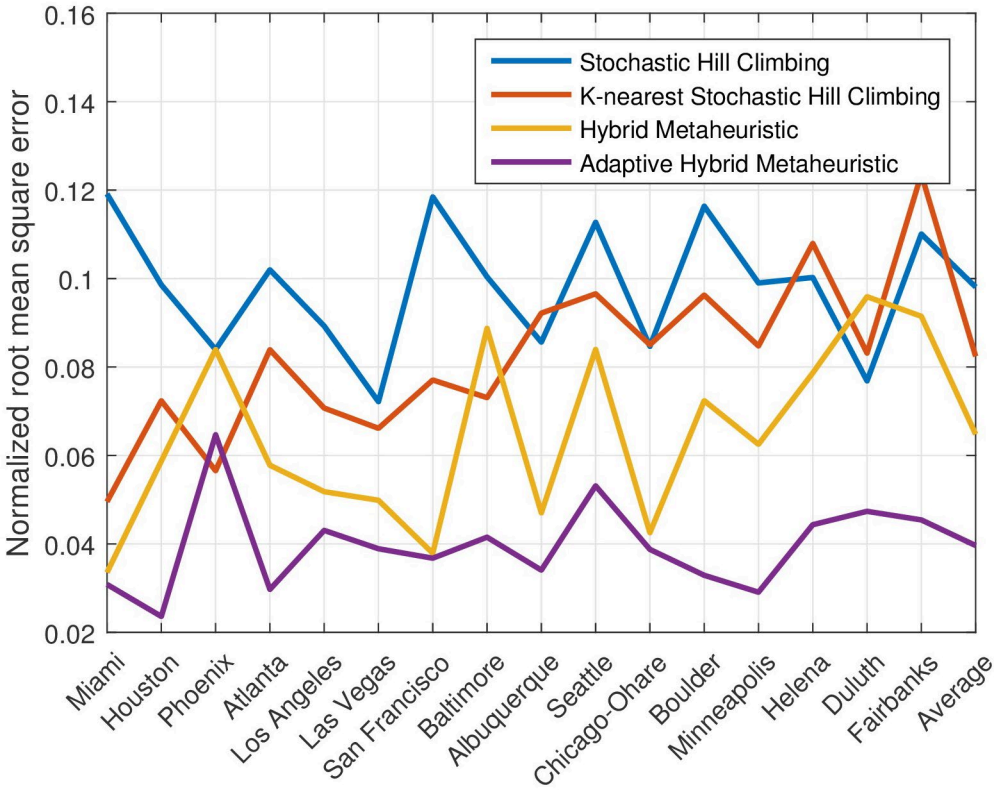


Figure 11. Goodness of fit of different levels of the proposed approach

Figure 12 shows the energy savings obtained for different levels of the proposed adaptive hybrid metaheuristic algorithm, actual optimal values driven from brute-force search, and the exhaustive optimal setpoints (a function that maps outdoor temperature to optimal setpoints developed based on the brute-force data) compared to the baseline settings (default values of the DOE simulation model with setpoint of 22.5 °C). The brute-force search resulted in average savings of 37.42%. The exhaustive optimal setpoints savings which represents the highest possible savings is 36.02%. In average, the generic stochastic hill climbing, the k-nearest stochastic hill climbing, hybrid metaheuristic, and the adaptive hybrid metaheuristic algorithms energy consumption consumed less energy by 20.97%, 23.34%, 27.50%, and 31.17% compared to the baseline (22.5 °C and 3 K), respectively. Each level of the implementation resulted in approximately 3% improvement compared to its previous level.

The superior performance of the adaptive hybrid metaheuristic is due to the fact that it utilizes the self-tuning process for optimizing its performance over the duration of data collection. There is a 5% gap between the adaptive hybrid metaheuristic algorithm savings and the exhaustive search savings from all permutations, which is primarily due to the fact that exhaustive search algorithm is trained using the

complete set of setpoints' permutations. As it can be seen in Figure 12, a control policy based on selecting optimal daily setpoint as a function of average daily outdoor temperature results in an average annual energy savings of 30 – 50% depending on climate for small office buildings except for the climate 3C (San Francisco) with 20% savings. The 10% gap between the climate 3C and other climates is mainly because the climate 3C has small variations in outdoor temperatures and thus, outdoor temperature is not a driving factor. In this climate, there is a 10% difference between simple stochastic hill climbing algorithm and the k-nearest neighbor stochastic hill climbing algorithm which points out using only one similar day results in an energy loss. K-nearest neighbor algorithm considerably improves learning by looking at several similar points.

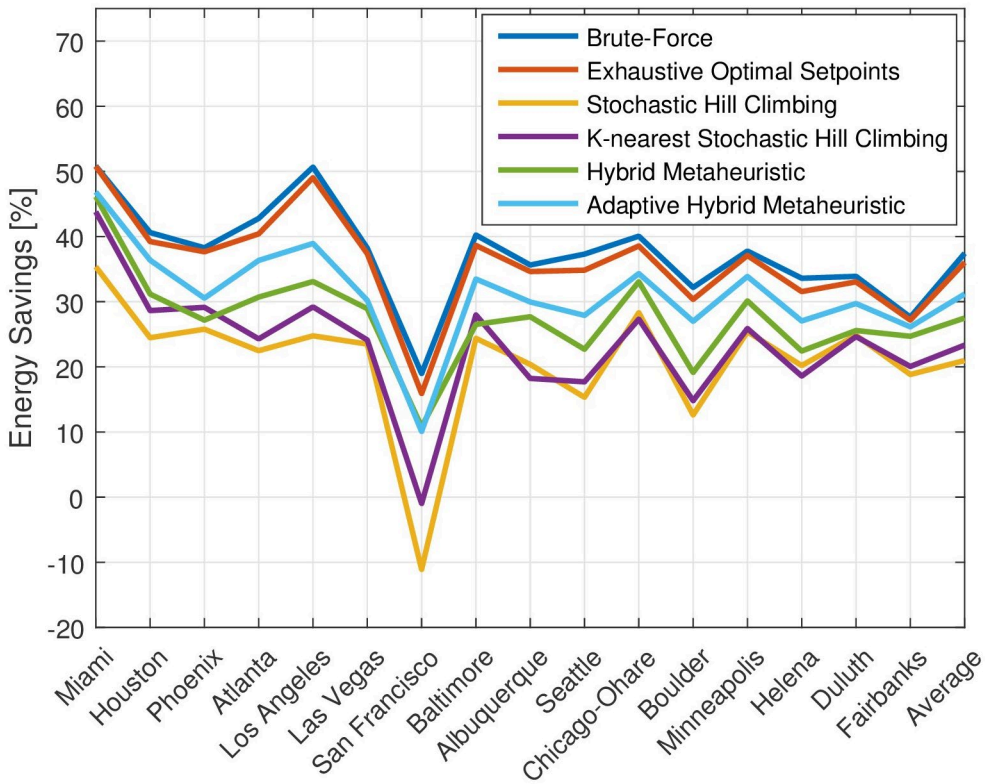


Figure 12. Energy consumption comparison between different levels of implementing the algorithm, brute-force, and the exhaustive optimal setpoints

4. DISCUSSION ABOUT THERMAL COMFORT INTEGRATION

In this section, we explain how our algorithm can be used in a control policy that integrates personal thermal comfort requirements. The block diagram of inputs, the algorithm, and the control policy is presented in Figure 13. The stream of the input data (e.g., building energy usage data and the dynamic variables) to the metaheuristic algorithm is usually collected in a building management system (BMS) database. These inputs are used to calculate an optimal setpoint at the building level on a daily basis. The HVAC controller selects a setpoint for each zone, as close as possible to the optimal setpoint (zone setpoint assignment in Figure 13) subject to the thermal comfort constraints on the zone setpoints. With the recent advancements in thermal comfort learning techniques (e.g., participatory sensing of comfort or using physiological measurements for learning comfort), real-time access to the thermal comfort of the occupants is becoming even more feasible [26-28]. The information provided by these methods could be

used to define comfortable zone setpoints and added as an input to the algorithm. Needless to say, thermal comfort integration into the HVAC control loop depends on the approach for learning personal thermal comfort and the control paradigm [29]. In this control paradigm, the thermal comfort requirements are the comfortable range of zone temperature setpoints. However, learning the thermal comfort requirements and demonstrating the performance of the control policy using our algorithm are not in the scope of this study.

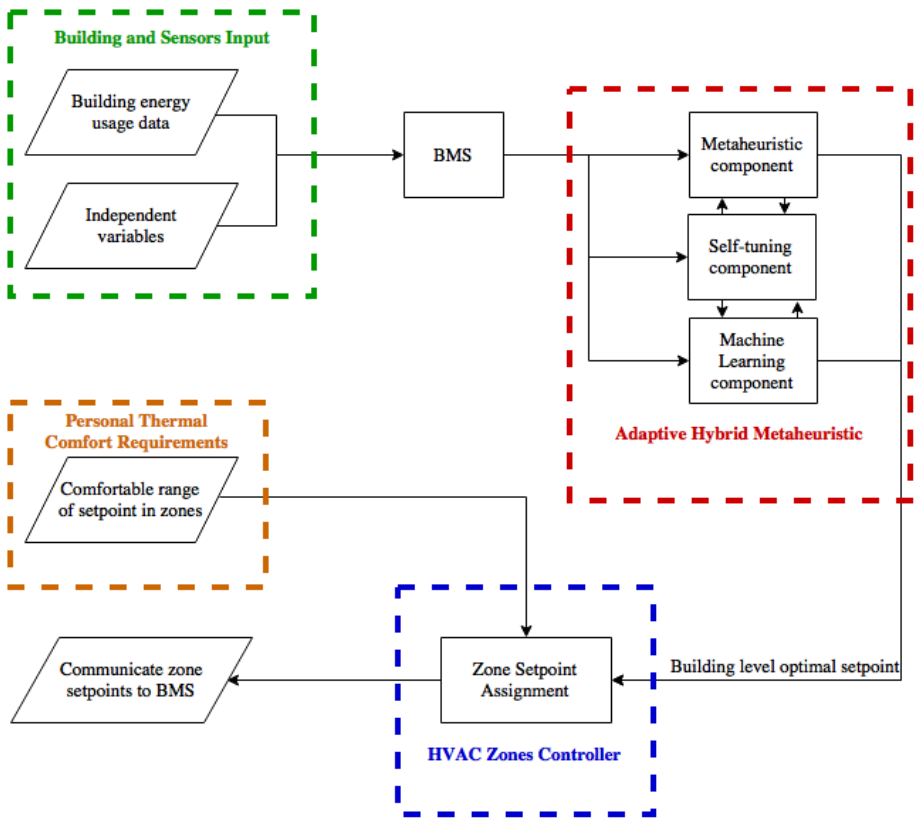


Figure 13. Proposed control policy for zone setpoint selection block diagram

Since an HVAC controller selects setpoints that are bounded by the limits defined by occupants' thermal comfort requirements, the occupants' thermal comfort is met. In addition, due to the fact that all zones' setpoints are selected close to a setpoint (optimal building level setpoint), thermal heat transfer between the zones is also expected to be reduced and consequently, HVAC system delivers its services with a higher energy efficiency. It should also be pointed out that since the setpoints are assigned as the target values of the zone thermostats, the minimum air flow rates, which are already set on the zone's Constant Air Volume (CAV) and Variable Air Volume (VAV) boxes, meet the indoor air quality constraints.

5. LIMITATIONS AND FUTURE WORK

In this paper, we introduced an adaptive hybrid metaheuristic algorithm to search and learn the setpoints for the optimal control of HVAC systems. We used the DOE small office building simulation model, which was served by CAV HVAC system with furnace for heating and packaged air conditioning unit for

cooling to validate our proposed algorithm. The algorithm is not limited to a specific type of building and HVAC equipment, since the underlying concept of optimizing HVAC control setpoints based on occupants thermal comfort remains the same in different types of HVAC systems. We studied and reported the savings for the small office buildings, which use CAV type of HVAC systems. The energy savings may differ in different types of buildings and HVAC systems.

Simulations results were only studied for workdays in this paper, however dynamic and variant occupancy has a direct impact on the HVAC energy consumption and need to be further investigated [30]. In parallel to the adaptive hybrid metaheuristic algorithm, an occupancy prediction module could provide the expected occupancy for a building for next day based on the historical building occupancy. Computational complexity of the recursive hyper parameter tuning component in the proposed control policy in different data sensing systems and data acquisition rates might negatively impact the potential energy savings and thus might result in a trade-off between complexity and energy savings [31-33], which requires further investigations. Although we observed about 3% energy savings improvement with increasing each level of algorithm implementation, the saving was still 5% lower than the optimal conditions. This gap can further be decreased by integrating other influential dynamic variables. In addition, the gap would naturally decrease over time as more permutations of the conditions are observed and thus, the performance of the algorithm would be enhanced and hence more energy efficiency could be achieved. The energy savings analysis presented in this paper did not include any thermal comfort constraints. Therefore, the savings are the maximum possible savings that our algorithm can achieve as the thermal comfort constraints on the setpoints could potentially result in more energy usage and reduction in savings.

We used a knowledge based approach for determining the initial temperature setpoint for heuristic component since we assumed there is no prior information about the HVAC operations hence the learning started on the first day. However, if the historical HVAC operational data with a variety of temperature setpoints are available, supervised learning methods (e.g., regression tree) can be used to select a more energy efficient initial setpoint. If the HVAC energy-setpoint relationship changes over time, the proposed model requires the time dimension to be modeled as another variable. Since the DOE simulation models do not have a dynamic behavior (e.g., lose or gain energy efficiency with respect to the setpoints), we could not integrate time-based modeling. We plan to integrate time-based modeling in a future study, deploying our algorithm in field implementations.

6. CONCLUSIONS

In this paper, we introduced an online learning algorithm (i.e., adaptive hybrid metaheuristic algorithm) to adaptively search and learn the setpoints for the optimal control of an HVAC system. The proposed algorithm is a model-free algorithm as it does not require a physical model. This addresses the challenges with model-based optimal controllers. Since it learns the optimal setpoints in an online fashion, it does not require substantial historical operational data. The algorithm consists of a metaheuristic component (i.e., a k-nearest neighbor stochastic hill climbing for a rough search of the space), a machine learning component (i.e., a regression decision tree that given the searched space finds the optimal setpoints), and a self-tuning component (i.e., a recursive algorithm that searches for the optimal the hyper parameters space of the metaheuristic and the machine learning components). The metaheuristic component starts to search the control parameter space (i.e., setpoints) for selecting more energy efficient setpoints through calculating a gradient based on the similar historical data points. Once the data points in a neighborhood of certain target variable are sufficient, the machine learning component fits a mathematical model to that neighborhood and finds the local optimal in the neighborhood. In parallel to these two components, a self-tuning component recursively strives to improve the performance of the two other components.

The control policy uses the data (e.g., gas/electricity consumption, weather, and occupancy) in real-time to set optimal setpoints at thermal zones' thermostats. In addition, this approach could allow for

integration of occupants' real-time thermal comfort requirements and optimal HVAC control through constraining the zone temperature setpoints. The control policy uses smart selection of daily setpoints as its control basis, making the control schema complementary to the legacy building management systems.

We used the DOE reference small office building in all U.S. climate zones for simulating the operations of control policies via EnergyPlus. The results indicated that the average highest possible savings is 36.02% for the 16 climate zones. In average, different levels of the implementation of the algorithm (i.e., the generic stochastic hill climbing, the k-nearest stochastic hill climbing, hybrid metaheuristic, and the adaptive hybrid metaheuristic) reduce the energy consumption by 20.97%, 23.34%, 27.50%, and 31.17% compared to the baseline (22.5 °C and 3 K), respectively. Each level of the implementation of the algorithm resulted in approximately 3% improvements compared to the previous level. adaptive hybrid metaheuristic has a superior performance because it utilizes a self-tuning process for optimizing its performance over the duration of implementation. In addition, the adaptive hybrid metaheuristic has a superior performance in all climate zones for the goodness of measure (i.e., normalized root mean square error) with a value of 0.047.

7. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 1351701. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

8. REFERENCES

[1] L.E. Doman, V. Arora, A. Metelitsa, M. Leahy, J. Barden, M. Ford, M. Mellish, A. Jones, B. Murphy, K. Lillis, International Energy Outlook 2013, IEO2013 Report, July (2013).

[2] L. Pérez-Lombard, J. Ortiz, C. Pout, A review on buildings energy consumption information, Energy and Buildings. 40 (2008) 394-398.

[3] R.W. Haines, M. Myers, HVAC Systems Design Handbook, Fifth Edition , McGraw-Hill Education, 2009.

[4] ASHRAE Standard, Standard 62.1-2010 (2010). Ventilation for Acceptable Indoor Air Quality, Atlanta, GA, American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc (2010).

[5] A. Ghahramani, K. Zhang, K. Dutta, Z. Yang, B. Becerik-Gerber, Energy savings from temperature setpoints and deadband: Quantifying the influence of building and system properties on savings, Applied Energy. 165 (2016) 930-942.

[6] N. Nassif, S. Moujaes, A cost-effective operating strategy to reduce energy consumption in a HVAC system, International Journal of Energy Research. 32 (2008) 543-558.

[7] N. Nassif, S. Kajl, R. Sabourin, Optimization of HVAC control system strategy using two-objective genetic algorithm, HVAC&R Research. 11 (2005) 459-486.

[8] G. Zheng, M. Zaheer-Uddin, Optimization of thermal processes in a variable air volume HVAC system, Energy. 21 (1996) 407-420.

[9] A. Ghahramani, K. Dutta, Z. Yang, G. Ozcelik, B. Becerik-Gerber, Quantifying the influence of temperature setpoints, building and system features on energy consumption (2015) 1000-1011.

[10] A. Kusiak, M. Li, F. Tang, Modeling and optimization of HVAC energy consumption, Applied Energy. 87 (2010) 3092-3102.

[11] M. Feldmeier, J.A. Paradiso, Personalized HVAC control system (2010) 1-8.

[12] A. Kusiak, G. Xu, Z. Zhang, Minimization of energy consumption in HVAC systems with data-driven models and an interior-point method, Energy Conversion and Management. 85 (2014) 146-153.

[13] J. Brooks, S. Kumar, S. Goyal, R. Subramany, P. Barooah, Energy-efficient control of under-actuated HVAC zones in commercial buildings, Energy and Buildings. 93 (2015) 160-168.

[14] A. Ghahramani, F. Jazizadeh, B. Becerik-Gerber, A knowledge based approach for selecting energy-aware and comfort-driven HVAC temperature set points, Energy and Buildings. 85 (2014) 536-548.
[15] A. Kusiak, F. Tang, G. Xu, Multi-objective optimization of HVAC system with an evolutionary computation algorithm, Energy. 36 (2011) 2440-2449.
[16] J. Nishiguchi, T. Konda, R. Dazai, Data-driven optimal control for building energy conservation (2010) 116-120.
[17] K.F. Fong, V.I. Hanby, T. Chow, HVAC system optimization for energy management by evolutionary programming, Energy and Buildings. 38 (2006) 220-231.
[18] M. Deru, K. Field, D. Studer, K. Benne, B. Griffith, P. Torcellini, B. Liu, M. Halverson, D. Winiarski, M. Rosenberg, US Department of Energy commercial reference building models of the national building stock (2011).
[19] S. Russell, P. Norvig, A. Intelligence, A modern approach, Artificial Intelligence.Prentice-Hall, Egnlewood Cliffs. 25 (1995).
[20] K.P. Murphy, Machine learning: a probabilistic perspective, MIT press, 2012.
[21] H. Zhao, F. Magoulès, A review on the prediction of building energy consumption, Renewable and Sustainable Energy Reviews. 16 (2012) 3586-3592.
[22] Energy Information Administration, 2012 Commercial Buildings Energy Consumption Survey (CBECS), Summary Tables, Tables B1. Summary table: total and means of floorspace, number of workers, and hours of operation (2015).
[23] V. Garg, K. Chandrasen, S. Tetali, J. Mathur, EnergyPlus simulation speedup using data parallelization concept (2010) 1041-1047.
[24] R.S. Briggs, R.G. Lucas, Z.T. Taylor, 4611 Climate Classification for Building Energy Codes and Standards: Part 2--Zone Definitions, Maps, and Comparisons, ASHRAE Transactions-American Society of Heating Refrigerating Airconditioning Engin. 109 (2003) 122-130.
[25] J. Michaels, T. Leckey, Commercial buildings energy consumption survey, URL http://www.eia.doe.gov/emeu/cbecs (2003).
[26] A. Ghahramani, G. Castro, B. Becerik-Gerber, X. Yu, Infrared thermography of human face for monitoring thermoregulation performance and estimating personal thermal comfort, Building and Environment. 109 (2016) 1-11.
[27] A. Ghahramani, C. Tang, B. Becerik-Gerber, An online learning approach for quantifying personalized thermal comfort via adaptive stochastic modeling, Building and Environment. 92 (2015) 86-96.
[28] A. Ghahramani, C. Tang, Z. Yang, B. Becerik-Gerber, A Study of Time-Dependent Variations in Personal Thermal Comfort via a Dynamic Bayesian Network, in: Sustainable Human–Building Ecosystems (2015) pp. 99-107.
[29] S. Ahmadi-Karvigh, A. Ghahramani, B. Becerik-Gerber, L. Soibelman, One size does not fit all: Understanding user preferences for building automation systems, Energy and Buildings. 145 (2017) 163-173.
[30] H.N. Rafsanjani, C. Ahn, Linking Building Energy-Load Variations with Occupants' Energy-Use Behaviors in Commercial Buildings: Non-Intrusive Occupant Load Monitoring (NIOLM), Procedia Engineering. 145 (2016) 532-539.
[31] H.N. Rafsanjani, C.R. Ahn, M. Alahmad, Development of Non-Intrusive Occupant Load Monitoring (NIOLM) in Commercial Buildings: Assessing Occupants' Energy-Use Behavior at Entry and Departure Events, in: Sustainable Human–Building Ecosystems (2015) pp. 44-53.
[32] H.N. Rafsanjani, C.R. Ahn, M. Alahmad, A Review of Approaches for Sensing, Understanding, and Improving Occupancy-Related Energy-Use Behaviors in Commercial Buildings, Energies. 8 (2015) 10996-11029.

[33] S. Ahmadi-Karvigh, B. Becerik-Gerber, L. Soibelman, A framework for allocating personalized appliance-level disaggregated electricity consumption to daily activities, Energy and Buildings. 111 (2016) 337-350.