Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 8849541, 20 pages
https://doi.org/10.1155/2020/8849541

Research Article

Data-Driven Predictive Control of Building Energy Consumption under the IoT Architecture

Ji Ke,[1] Yude Qin,[1] Biao Wang,[1] Shundong Yang,[2] Hao Wu,[1] Hang Yang,[1] and Xing Zhao[1]

[1]*College of Electronic & Control engineering Chang'an University, Xi'an 710061, China*
[2]*KunYi Co., Ltd., Wuxi 214000, China*

Correspondence should be addressed to Biao Wang; wangbiao@chd.edu.cn

Received 5 June 2020; Revised 10 November 2020; Accepted 27 November 2020; Published 15 December 2020

Academic Editor: Hongzhi Guo

Model predictive control is theoretically suitable for optimal control of the building, which provides a framework for optimizing a given cost function (e.g., energy consumption) subject to constraints (e.g., thermal comfort violations and HVAC system limitations) over the prediction horizon. However, due to the buildings' heterogeneous nature, control-oriented physical models' development may be cost and time prohibitive. Data-driven predictive control, integration of the "Internet of Things", provides an attempt to bypass the need for physical modeling. This work presents an innovative study on a data-driven predictive control (DPC) for building energy management under the four-tier building energy Internet of Things architecture. Here, we develop a cloud-based SCADA building energy management system framework for the standardization of communication protocols and data formats, which is favorable for advanced control strategies implementation. Two DPC strategies based on building predictive models using the regression tree (RT) and the least-squares boosting (LSBoost) algorithms are presented, which are highly interpretable and easy for different stakeholders (end-user, building energy manager, and/or operator) to operate. The predictive model's complexity is reduced by efficient feature selection to decrease the variables' dimensionality and further alleviate the DPC optimization problem's complexity. The selection is dependent on the principal component analysis (PCA) and the importance of disturbance variables (IoD). The proposed strategies are demonstrated both in residential and office buildings. The results show that the DPC-LSBoost has outperformed the DPC-RT and other existing control strategies (MPC, TDNN) in performance, scalability, and robustness.

1. Introduction

One major challenge in today's society concerns energy savings and CO_2 footprint in existing and new buildings. To date, the building sector has witnessed immense development in the way by which building systems are managed [1, 2],which aimed at alleviating the significant environmental impact of this sector (40% of the world energy consumption and a third of the associated CO_2 emissions [3]). Decreasing this impact could be achieved by elegant controlling the resources [4]; building energy management systems (BEMS) provides sustainable and efficient solutions.

An expected BEMS aims to increase energy efficiency while maintaining the required comfort levels and enhance environmental effects. However, based on a large number of practical implementations, it is found that the current problems in existing BEMS are mainly concentrated in the following aspects: (1) the traditional BEMS mostly has relatively sole functions. For example, the systems lack effective monitoring and linkage management of the dynamic environment and energy-related equipment, (2) the family of BEMS is still far from the standardization of communication protocols and data formats, and (3) many systems only collect and store data employing local databases for monitoring and lack supervisory applications (advanced control, human-machine interactions, data analysis). The dilemma attributes to the usage of supervisory control and data acquisition (SCADA) architecture in existing BEMS [5].

The popularity of the Internet of Things (IoT) and its successful industrial applications provide a new perspective for us to deal with the dilemma above [6–8]. Utilizing the IoT technology, a massive amount of data is aggregated into a unified energy management platform. GÃ¼nter Alce proposes a new concept of IoT interaction [9]. However, some challenges have arisen. How to manage big data (transferring, storing, preprocessing, optimization, and control under a suitable IoT framework) [10]. On the other hand, delivering useful information to different stakeholders based on their use is another challenge [1, 11–13]. These challenges pose new questions: what is the complex SCADA-based BEMS framework under IoT, and how to build it? On top of these, what are suitable control strategies for achieving optimal BEMS performance?

Numerous studies have proven that an advanced control strategy could significantly reduce energy use and alleviate greenhouse gas emissions, see, e.g., [14–16]. However, many buildings currently adopt rule-based control (RBC) with limited energy-saving capabilities [17–19]. Many studies have proved that the building sector can significantly benefit from replacing the current RBC for more advanced control strategies like model predictive control (MPC) [18]. MPC's perfect performance is achieved by accounting to minimize consumed energy and maintain high comfort indexes while considering physical constraints, weather forecasts, and building dynamics. In recent years, many energy-efficient MPC approaches have been validated to control the building systems [20–24]. Despite these tries, RBC-based control remains business as usual in the building sector. A key factor prohibiting this technology transfer to the commercial sector is the cost, time, and effort associated with capturing first-principle-based dynamical models of the building. Also, a gap always exists between the modeled and the real building, and the domain expert must then manually tune the model to match the measured data from the building [25, 26].

An alternative approach for implementing MPC is using control-oriented data-driven predictive models. In the literature, this approach is called data predictive control (DPC) [25]. In [27, 28], the authors proposed MPC closed-loop optimization strategies based on neural networks (NN) for energy-saving control in buildings both in commercial and residential buildings, respectively. However, these approaches are not easily scrabbled to different types of buildings [29]. NN is employed in the closed-loop control scheme to determine control performance indexes instead of neural network-based system state dynamics. Unfortunately, since NN, a nonlinear nature, the complementary MPC-based optimization problem becomes computationally more demanding when the neural network's complexity is high.

To overcome this complexity above, the regression tree-based approaches were employed in the literature to develop data-driven predictive models. Authors in [30, 31] developed RT and random forest for building control in different settings. However, the simulation results demonstrated that these models were trapped in limitations due to overfitting and high variance [5]. In [18], a well-performing approximate MPC via machine learning has been developed based on two multivariate regression algorithms, namely, deep time-delay neural networks (TDNN) and regression trees (RT) on Hollandsch Huys, which is an office building in Belgium. This approach mentioned above is an advantage which is the simplified control laws that retain comparable performances with MPC. However, the RT-based controller scored worse in performance than a well-tuned PID controller, which dates back to modeling inaccuracy. To overcome the drawbacks of previous works above, we present an ensemble learning algorithm, called least-squares boosting (LSBoost), which integrate multiple decision trees to produce robust models. The residential building model data in [18] will also be used in our simulation and validation. What is different from prior studies is our work focus on data-driven optimization control of BEMS both in residential and office buildings under the IoT framework.

In this paper, we develop a data-driven energy optimization control strategy based on an improved LSBoost under a layered building energy IoT framework, which improves the occupants' comfort and reduces energy consumption. We validate the proposed control scheme by numerical simulation with two types of buildings: residential building and office building. The work has the following contributions:

(1) We develop a novel four-tier building energy internet architecture. This architecture is used for managing data from both IoT devices and BEMS, by using a cloud-based user-friendly human-machine interaction interface. The motivating factor behind the developed architecture will be elaborated details in Section 2. The platform above can provide useful data representations to different stakeholders (end-user, building energy manager, and/or operator), enabling flexibility and scalability.

(2) An optimization strategy is the first strategy proposed for building energy management based on the improved LSBoost. The LSBoost algorithm is used to enhance the building model's interpretability and reduce complexity without losing accuracy. The optimization problem takes the optimal index of human comfort into account the constraints to ensure a good living and office environment.

The remainder of this paper is organized as follows. A novel building energy internet architecture is presented in Section 2. In Section 3, two types of buildings, namely, residential and office buildings, are built. Section 4 defines the finite receding horizon control problem with the DPC framework. We compare the performance of the DPC-LSBoost with the MPC, the TDNN, and the DPC-RT in Section 5. Conclusions and further work are provided in Section 6.

2. Building Energy IoT System Architecture

This section describes a complex cloud SCADA-based BEMS framework under the IoT, which is necessary for successfully implementing DPC in public buildings.

2.1. Cloud SCADA System. When the existing SCADA-based BEMS framework meets the IoT, the local servers will feel

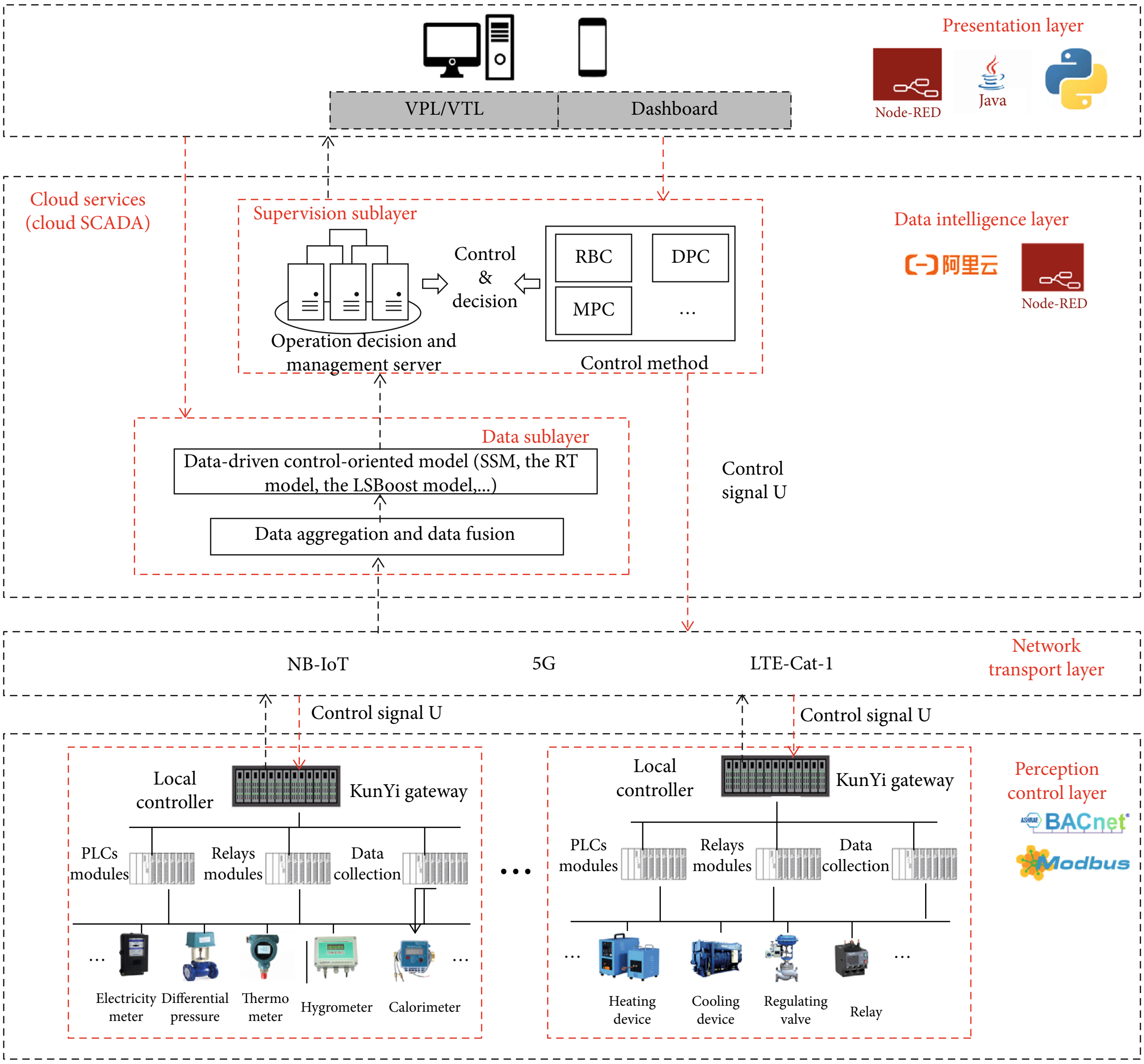


Figure 1: The overall architecture of the IoT in the building energy system.

helpless against a massive amount of data. Also, the existing SCADA-based BEMS is still far from the standardization of communication protocols and data formats, which is unfavorable for advanced control strategies implementation. We decided to develop a cloud-based SCADA system and its ecosystem to deal with the problem above. The cloud-based solution's motivation is its compatibility with user-friendly and easy access to the real data, instead of additional hardware investments [32].

2.2. Four-Tire Building Energy IoT System Architecture. The existing SCADA-based control layers in a BEMS constitute three separated layers [33], and those are (1) field layer (sensors, actuators, controllers), (2) automation layer (signal processing, controlling, alarms activating), and (3) management layer (system data presentation, trending, logging, and archival). The IoT also consists of three separate layers: (1) perception layer, (2) network layer, and (3) application layer. However, we introduce a four-tire client-server software architecture web platform consisting of four layers: perception control layer, network transmission layer, data intelligence layer, and representation layer.

The motivation factor behind the four-tire architecture is the complementary advantages for the SCADA architecture and the IoT architecture. One of the main advantages of using a SCADA configuration is that the control and communication flows can be presented sequentially [5]. However, the strength of the IoT configuration is the ability of data processing in-depth. Based on the reasons mentioned above, we define the first 3 layers. In addition, to make IoT data form building useful to different stakeholders, we decide to develop the presentation layer as the fourth layer. The idea is inspired by a building lifecycle data management strategy in [1]. The architecture of the IoT in the building energy system is shown in Figure 1.

(1) *Perception control layer*: in BEMS, this layer is endowed with two primary functions:

(a) Collecting sensor data of environment parameters (such as indoor and outdoor temperature, humidity, and wind speed), power consumption, pressure difference, water flow, and heat

(b) Receiving control signals from the field controllers or executing agencies to ensure that the control objectives, i.e., heating unit refrigeration unit, works properly.

(2) *Network transmission layer*: the Internet of Things communication technology such as NB-IoT, 5G, is utilized to ensure the sensor data upload and control signal U transmission.

(3) *Data intelligence layer*: this layer consists of two sublayers: the data sublayer and the supervision layer. The data collected by sensors will be filtered and fused first, and the abnormal signals are checked to ensure the data's integrity and accuracy, which is stored in the database. Then, the existing data is used to establish the control-oriented building models. The supervision sublayer is based on the data-driven predictive model, according to the set optimization objectives, using the optimization control technology developed in this paper (such as the MPC, the DPC-RT, the DPC-LSBoost, and the TDNN), to form the control strategy, such as ensuring the building's indoor environment control requirements while making the building energy consumption lowest.

(4) *Presentation layer*: the presentation layer endows two main functions: visual programming Language (VPL) interface and textual programming language (TPL) interface—dashboard. The dashboard is an information management tool for different stakeholders, including the environmental parameters setting, real-time monitoring data display, the PMV value, and energy consumption prediction.

3. Building Modeling and Analysis

This section describes the linear time invariant (LTI) state space model (SSM) for residential buildings used in this study.

Firstly, the internal structure of complex building is modeled. Its purpose is to accurately build the HVAC system and internal housing structure. Moreover, the house is easily affected by the natural environment. The disturbance of the external environment to the building should be considered. Common disturbances include ambient temperature, light intensity, wind speed, humidity, and other disturbance information so that the mathematical model can be close to the real building.

3.1. Residential Building Modeling

3.1.1. Model Description. The building model is located in a six-bedroom townhouse in Bruges, Belgium. The residential building consists of 6 guest rooms, 5 windows, and 11 single buildings with external walls. For the temperature control system of residential buildings, the central steam furnace is used for heating. For the building's parameters, including building area, room orientation, and other information, please refer to the literature [18].

Table 1: Dimensions of key variables in the building model.

Notation	Description	Values
n_x	Number of states	286
n_u	Number of inputs	6
n_y	Number of outputs	6
n_r	Number of output references	6
n_d	Number of measured disturbances	44

At the beginning of building the model, the Modelica building envelope model is implemented by using idea library, but its complexity cannot be directly used as a state-space model. A large number of collected data are nonlinear and need to be linearized before they can be used. For example, the heat generated by solar radiation: the equation of sunlight transmission and absorption through windows is highly nonlinear, so if you want to deal with it, you have to use a nonlinear filtering algorithm. For these unprocessed data, to remove the burr, the processing algorithm is extended Kalman filter [34]. After linearization, the state space expression can be constructed. For a complete description of building state-space expressions, please read the paper [35]. The sampling interval for humidity, temperature, wind speed, light, and other sensors is 15 minutes in the sensor and control layer. Therefore, the discrete space expression is constructed as follows:

$$x_{k+1} = Ax_k + Bu_k + Ed_k, \tag{1a}$$

$$y_k = Cx_k + Du_k. \tag{1b}$$

In the above equation, x_k, u_k, and d_k, respectively, represent the state, input, and disturbance variables at time k; y is the output variable; the model's sampling frequency is $Ts = 900$ sec. The disturbance signal d_k presents the heat absorbed and the direct and diffuse solar radiation transmitted by each window such as radiation temperature of ambient and sky temperature, ambient temperature, and ground temperature. Table 1 summarizes the dimensions of the building model variables used.

3.1.2. Model Analysis. House analysis is the analysis of the established model (SSM). From the house's perspective, entering a changing curve to reflect the change of the indoor temperature of the model without the control of the controller. Entering U

$$\begin{aligned} U_{50*1348} &= [R_{6*1348} D_{44*1348}], \\ R_{6*1348} &= 20 + 3 * \sin\,(t + k * ts)(k = 1, 2 \cdots 1348), \end{aligned} \tag{2}$$

with R_{6*1348} as the input temperature input, the external environmental disturbance as $D_{44*1348}$, k as the sampling

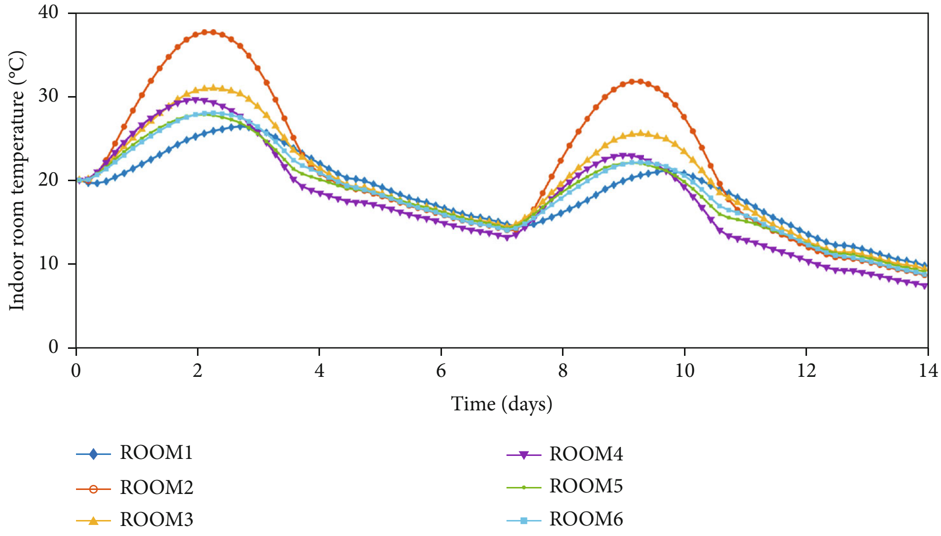


Figure 2: 14-day temperature variations of 6 rooms.

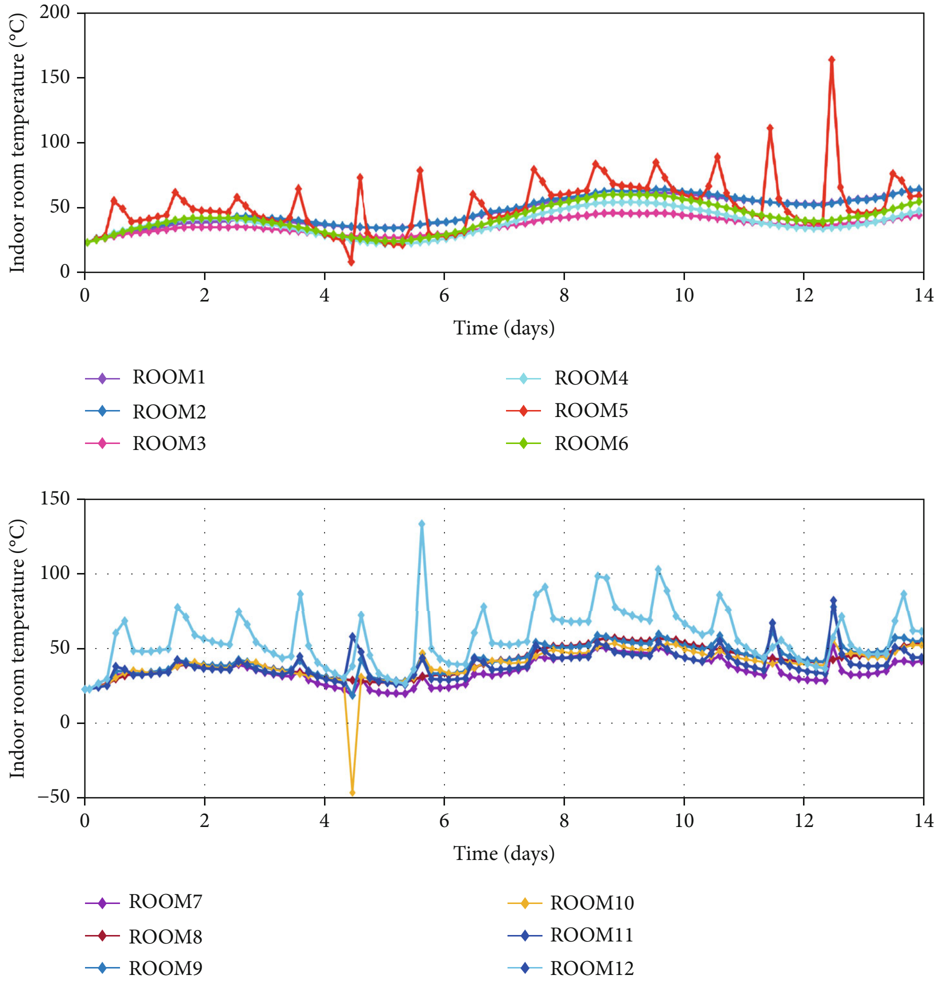


Figure 3: 14-day temperature variations with office building.

Table 2: Notation and meaning of the variables used in optimization control.

Notation	Units	Description	Control setup
x	[K]	Building temperatures	States
y	[K]	Controlled temperature	Outputs
r	[K]	Reference temperature	References
u	[W]	Radiators heat flows	Inputs
d	[K,W]	Temperatures, heat flows, and radiation gains	Disturbances
s	[K]	Comfort band violations	Slack variables
ub	[K]	Upper comfort boundary	Constraints
$1b$	[K]	Lower comfort boundary	Constraints

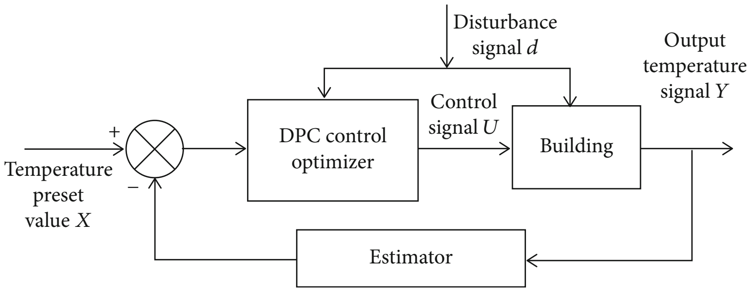


Figure 4: Schematic representation of the building optimal control closed-loop system using the DPC controller.

time, and the sampling interval as 4 hours $ts = 4\,(\text{hour})$. The total number of simulation days is 14 days. Two weeks can reflect the changing trend of the model temperature, so as to better design the controller. Figure 2 shows the temperature change of 6 rooms in 14 days.

The overall trend of changes in the 6 rooms is consistent with the gradual decrease in temperature. However, it is quite different from the preset temperature. The maximum temperature of the ROOM2 can reach 37°C, which is 10°C higher than the other two models. The same goes for other room temperature trends. Therefore, for the model, the stability of the model is the most important. A major feature of the choice of the building body model is stability and robustness. Such a building conforms to the human habitation. And when the controller is not added, the temperature of the room will gradually decrease, and, finally, drop to 7°C.

3.2. Office Building Modeling

3.2.1. Model Description. The office building is modeled by Hollandsch Huys, and the building is located in Hasselt, Belgium. Hollandsch Huys represents a class of geotab buildings [36]. Hollandsch Huys is a 5-storey building, including 3-storey office areas located in the ground floor, first floor, and second floor; underground garage; and top loft. When building the office building model, considering the complexity of the model and the personnel distribution, it is mainly to build a model for the three floors of the office area. Please refer [37] for the main parameters of relevant building structures.

Table 3: Corresponding level of CIHB index.

CIHB	Level	Corresponding to human feeling
>85	4	Very hot and uncomfortable Need to protect against heatstroke
~ 85	3	Too hot; need to heatstroke prevention
~ 79	2	Hot, uncomfortable, needs to be cooled
~ 75	1	Warm, comfortable
59 ~ 70	0	Most comfortable and acceptable feeling
~ 58	-1	Cold and uncomfortable
~ 50	-2	Cold and uncomfortable. Keep warm
~ 38	-3	Very cold, keep warm, and cold protection
≤25	-4	Extreme cold, prevent frostbite

Similarly, the office building's SSM is established by using the method described in 3.1. The office building's SSM construction is consistent with that of the residential building construction, please refer to Equations (1a) and (1b). Compared with residential buildings, the SSM of office buildings is more complex, and the variable dimension is higher. The dimensions of the Hollandsch Huys building model variables n_x, n_u, n_y, n_r, and n_d are 700, 20, 12, 20, and 301, respectively.

3.2.2. Model Analysis. For an excellent mathematical model, we hope that the model we build can be applied, so we need to simulate the established SSM and simulate the output-

```
Require:$A, B, C, D, E$ matrix, temperature boundary $lb, wb$, DPC horizon $N$, data set, model (select control method such as $RT$, $LSb$
$oost$)
Ensure:$U$
Step 1:Build a prediction model.
(A)Data dimensionality reduction, feature selection(Algorithm 2 and Algorithm 3).
(B)Training data to build a model.
if Model==$RT$then.
    Build RT model $M_{RT}$.(Algorithm 4).
else if Model==$LSBoost$then.
    Build LSBoost model $M_{LSBoost}$.(Algorithm 5).
ends
Step 2:Model optimization cossntrol.
while$k < N$do.
(A)Calculate the $u_k{}'$ from $M_{RT}$ or $M_{LSBoost.}$
(B)Through Equation 1, calculate $y_k{}'$.
(C)Update $u_k{}'$,$y_k{}'$.
(D)Solve Equation 5 and obtain $(u^*_{k|k}, u^*_{k+1|k}, \cdots, u^*_{k+N-1|k})$.
(E)$u_k \longleftarrow u^*_{k|k}$ and $k \leftarrow k + 1$.
end
return$U = \{u_0, u_1, \cdots, u_{N-1}\}$
```

Algorithm 1. Building data-driven predictive control.

```
Require: feature vector $x$ important features $p$.
  Step 1:Compute the covariance matrix of the feature vector x:$\Sigma = (1/\mathrm{m})x^T x$
  Step 2:SVD decomposition of the covariance matrix $\Sigma$,$\Sigma = USV^T * U$, $S$ are the principal component coefficients and variances.
  Step 3: Matrix S accounts for the proportion of total features,and call it as $v_i$,
          $v_i = S_{i,i}/tr(S)$
  $*tr(S)$ is the trace of the matrix $S$.
  Step 4:Select the principal component variance corresponding to the $q$ most significant singular values,
          $\max q\,\mathrm{s.t.} \sum_{i=1}^{q} v_i \le \eta$
  Step 5:Compute the normalized contribution $v^j$ of the j-th feature $x_j$ on selected principal components.
          $v^j = \sum |U_{j,1,\cdots,q}| / \max_{1\le k\le n_x} (\sum_{i=1}^{q} |U_{k,1}|)$
  Step 6: The $p$ most important features that satisfy $\psi$ are selected,
return$p = \{i \mid v^j \ge \psi, \forall \in N_0^{n_x}\}$
```

Algorithm 2. PCA feature selection.

```
Require: Perturbation matrix $E$ in equation 1
Ensure: most important features $q$
  Step 1:Compute the $l_1$ norm of each column vector in matrix $E$, given as $IOD_i$.
          $IOD_i = \sum_{j=1}^{n_x} \|E_{i,j}\|_{l_1}\ i \in 1, \cdots, n_d$
  Step 2:$IOD_i$ Feature normalization,
          $AIOD_i = (IoD_i / \sum_{i=1}^{n_d} IoD_i) i \in 1, \cdots, n_d$
  Step 3:Select the disturbance feature $q$ that satisfies the threshold $\zeta$.
  return$q = \{i \mid AIoD_i \ge \zeta\}\ i \in 1, \cdots, n_d$
```

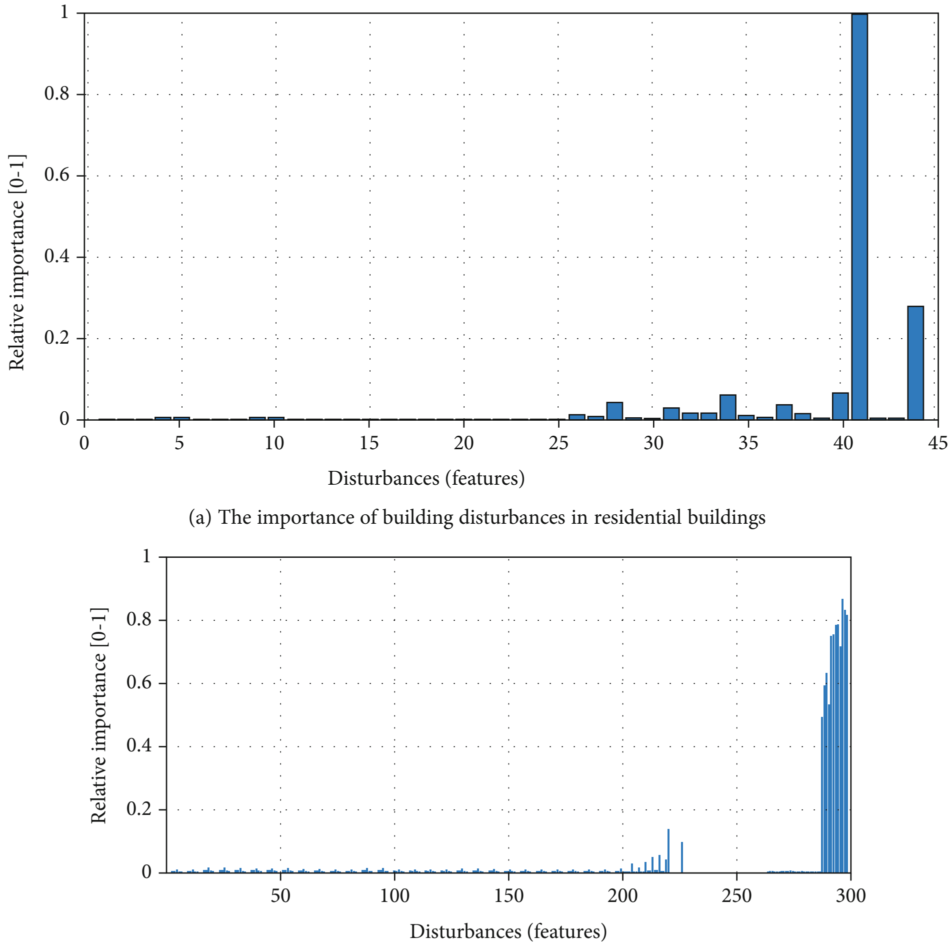
Algorithm 3. The importance of disturbing variables (IoD).

output relationship to evaluate the established model's quality. The input we take this time is U

$$
\begin{aligned}
U_{321*1348} &= \left[R_{20*1348} \quad D_{301*1348} \right], \\
R_{20*1348} &= 23 + 3 * \sin\,(t + k * ts)(k = 1, 2 \ldots 1348),
\end{aligned}
\tag{3}
$$

with $R_{20*1348}$ as the input temperature, external environment disturbance as $D_{301*1348}$, k as the sampling time, the sampling interval as 4 hours, ts = 4hours, and the total number of simulation days as 14 days. The simulation results are shown in Figure 3.

It can be seen from Figure 3 that the SSM is constructed. The temperature change is greatly affected by external



(a) The importance of building disturbances in residential buildings

(b) The importance of building disturbances in the office building

Figure 5: The IoD profile on two types of buildings.

disturbance. As long as the input has a temperature change, the output temperature will be severely affected. The building model itself is a time-delay system, so the temperature of the output will change greatly. It is difficult to output the input of tracking. From Figure 3, The temperature of each room may change abruptly, and either it becomes very high or very low. This established model has a certain distance from the actual model. However, it is still in a stable state for the entire system. Therefore, it is necessary to design a controller for the built house building model. The most significant purpose of the controller is to achieve stability and stability within the human comfort zone.

4. Data-Driven Predictive Control

The supervision sublayer design's primary purpose is to design the controller, which plays the role of the control decision. Because the control effect of the designed controller will directly affect the indoor temperature and human comfort, the controller design should be deeply analyzed from the complexity, real-time, and robustness. The more commonly used controllers in the market should be comprehensively analyzed, and finally, the DPC-LSBoost controller should be selected. Because the DPC-LSBoost controller has a perfect explanation, the regression tree constructed is straightforward and easy to understand, convenient for management, and decision-making.

Table 4: Comparison of the disturbance features of the two buildings.

	No. of the original	No. of the (PCA)	No. of the (E)	No. of the selected	Reduction rate (%)
Residential	44	12	8	8	81.81
Office	301	14	16	12	96.01

4.1. Control Optimization Design for Comfort Objective. In the building energy management system, most of the designed design controllers need to meet certain comfort and economic practicality. Therefore, when designing the controller, the reference input is a range instead of a specific value, which is called the comfort zone. This paper adopts the ISO-7730 standard [38], which specifies the upper limit of temperature between $ub \in [23, 26]$ and the lower limit of $lb \in [20, 23]$. Equation (4) defines the mathematical expression of the temperature comfort zone.

$$lb_k - s_k \leq y_{i,k} \leq ub_k - s_k, \tag{4}$$

with s as a slack variable, k as a time series, and $y_{i,k}$ represents the ith room temperature at time k. It is necessary to ensure that the output temperature is within the comfort zone, so as to minimize the s_k and the room energy consumption. However, the ideal comfort and energy consumption are contradictory. In order to solve this problem, the control problem is transformed into an optimization problem. Table 2 lists the symbols and meanings of variables frequently used in this section.

Figure 4 is a control structure diagram designed to solve this optimization problem. The purpose of control is to achieve minimum energy consumption and maximum human comfort, which involves two variables: the control signal U and the output temperature Y. In summary, Equation 5 established an optimization function.

$$\min_{u_0, \cdots, u_{N-1}} \sum_{k=0}^{N-1} \left(Q_s \| s_k \|_2^2 + Q_u \| u_k \|_2^2 \right), \tag{5a}$$

$$s.t. x_{k+1} = Ax_k + Bu_k + Ed_k, k \in N_0^{N-1}, \tag{5b}$$

$$y_k = Cx_k + Du_k, k \in N_0^{N-1}, \tag{5c}$$

$$lb_k - s_k \leq y_k \leq ub_k - s_k, k \in N_0^{N-1}, \tag{5d}$$

$$x_0 = x(t), \tag{5e}$$

$$d_0 = d(t), \tag{5f}$$

$$\frac{59 + 3 \cdot 2\sqrt{v} - 32 - 0.143 + 0.143RH}{0.81 + 0.143 + 0.143RH} \leq y_k \leq \frac{70 + 3.2\sqrt{v} - 32 - 0.143 + 0.143RH}{0.81 + 0.99RH}, \tag{5g}$$

with $N_a^b = \{a, a+1, \cdots, b\}$ as a set of integers, and x_k,u_k,y_k and d_k represent state, input, output, and disturbance variables, respectively. The prediction range is N, and k is the k-th moment in the prediction range. (5b) and (5c) are the time-invariant state space expressions of the building (5d). The lower boundary lb_k and the upper boundary ub_k are taken into consideration. (5g) introduces the popular Comfort Index of Human Body (CIHB) in recent years [39] and divides it into 9 levels to evaluate comfort Table 3. The index also considers the effects of average temperature, relative humidity and wind speed on human comfort. Equation (6) is shown below.

$$\text{CIHB} = 1.8y - 0.55(1.8y - 0.26)(1 - RH) - 3.2\sqrt{V} + 32 \tag{6}$$

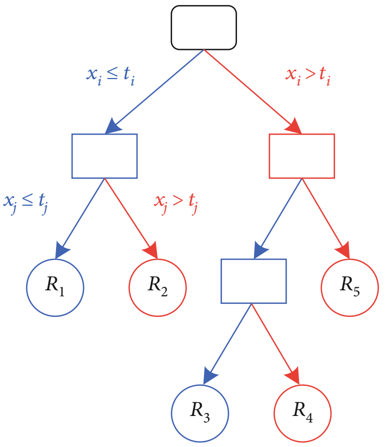


Figure 6: After dividing the regression tree twice, we get 5 sets $R_1, \cdots, R_5$.

with y as the average temperature °C, RH as the average humidity (%), and V as the wind speed (m/s). According to Table 3, comfort level 0 is the most liveable environment for the human body. The CIHB index should be the most reasonable at $59 \sim 70$, which is converted into an inequality (5 h) about temperature, to construct a constraint (5e). Limit the maximum and minimum boundaries of the control signal u_k. (5f) and (5g) set the initial parameters. (5a) indicates that the objective function finally constructed by the optimization problem outputs a sequence $u_0, u_1, \cdots, u_{N-1}$ under the influence of 7 constraints, so that the output control amount is minimized, the objective function $\|\cdot\|_2^2$ represents the square of the second norm, s_k is a slack variable, u_k is a control variable, Q_s represents the weight of human comfort, and Q_u represents the weight of energy consumption. The weighting matrices Q_s and Q_u are given as positive definite diagonal matrices. Set it to $Q_s/Q_u = 10^7$. The first term in the objective function is the square with the lowest degree of comfort violation, and the second term is the square with the lowest energy consumption.

The architecture of DPC is shown in Algorithm 1. The DPC-LSBoost is a DPC algorithm with the LSBoost model. The DPC-RT is a DPC algorithm with the RT model.

4.2. Feature Selection. This section proposes a simple and systematic approach for the efficient feature selection (FS) of predictive models in the context of building energy control applications. Because the method introduced in this section is versatile, it can be used to identify and select the most relevant variables in a dynamic building model, reducing model complexity, or reducing the cost of sensing equipment in practice. For current building data, feature selection based on principal component analysis (PCA) is first proposed. The simplicity and the PCA algorithm efficiency are well known, so we choose the method described in [40] to perform feature selection on the dataset we build. Algorithm 2 shows the PCA feature selection progress.

```
Require: Data in Equation (7) and a Loss Function Equation (11)
Ensure:$T_{\min}$
  Step 1:Using Equation (11) to recursive binary splitting makes a large tree $T_0$ on the training data.
  Step 2:Use K-fold cross-validation to choose best tree.
  For k =1,...,K:
  Apply cost complexity pruning(prune) to the large tree in order to obtain a sequence of best subtrees, as a function of $\alpha$.
  end
  Step 3:After k-th fold, An optimal tree $T_{iin}$ is selected.$T_{\min} = \underset{\mathrm{T}_k}{\mathrm{argmin}}\alpha_k, k = 1, \cdots, K$
returnT$_{\min}$
```

Algorithm 4. Regression tree algorithm.

Now, we consider that there are N sets of observations in a data set, and each set of observations contains s features and n outputs, written as a mathematical expression as follows:

$$\begin{aligned} x^i &:= \left[x_1^i, x_2^i, \cdots, x_s^i\right] \in R^s \\ y^i &:= \left[y_1^i, y_2^i, \cdots, x_n^i\right] \in R^n \cdot \\ & i \in \{1, 2, \cdots, N\} \end{aligned} \tag{7}$$

The x^i shown in Equation (7) encapsulates all parameters that change over time, for example, the current state quantity $x(t)$, the current and future disturbance variable $d(t), \cdots, d(t + kTs)$ in Equation 1, and comfort boundary signals $l b(t), \cdots, lb(t + kTs)$ and $ub(t), \cdots, ub(t + NTs)$. Among them, the feature selection for disturbance is mainly considered in the degree of influence of the disturbance variable d_t on the system. Among them, the feature selection of the disturbance is mainly considered to the degree of influence of the disturbance variable d_t on the system.

In algorithm 2,the data variables have a large dimension, so the PCA feature selection is utilized to reach a more appropriate dimension, and the accuracy thresholds $\eta = 0.99$ and $\psi = 0.99$ are chosen.

Then, with the house disturbance model, the matrix E of the LTI model constructed in Equation 1 considers the disturbance's influence on the system.

Figure 5 shows the impact of construction disturbance both in residential and office buildings. The higher the index of $AIOD_i$ means the higher impact of the disturbance on the system performance.

Therefore, from the above two types of algorithms, the most relevant features can be filtered, and the intersection of the two sets is taken as the FS that is finally selected.

$$FS = p \cap q, \tag{8}$$

where p is the important feature set, and q is the important disturbance set.

So, the distribution features of the models can be obtained, as shown in Table 4. The features of residential buildings and office buildings are 81.81% and 96.01%, respectively. It is shown that the more features mean better results using this feature selection method. It also means that many of these features are redundant.

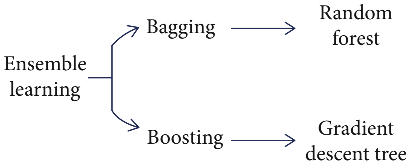


Figure 7: Ensemble learning classification.

4.3. Design of the DPC-RT Controller.

This section focuses on the prediction modeling of multiple output regression tree. Because of a lot of advantages of the regression tree, the controller adopts a very representative The controller adopts a very representative regression tree method in machine learning because of the RT advantages. Tree method in machine learning. The regression tree, as the name implies, is to use tree model to do regression problems, and each leaf will output a prediction value. The predicted value is generally the mean value of the output of the training set elements contained in the leaf,

$$y_m = \mathrm{ave}\left(y^i \mid x^i \in \mathrm{leaf}_m\right), \tag{9}$$

with y_m as the predicted output value of the m-th leaf. When $x_i \in \mathrm{leaf}_m$, the training set outputs y_i. ave means averaging.

The nodes of the tree split are shown in Figure 6. With each split, the regression tree divides the current data set into two subsets. For example, in i-th divided nodes, the left branch tree R_L contains data divided by $x_i \le t_i$, and the right branch tree R_R contains data divided by $x_i > z_i$. Then, the optimal segmentation point of each node is determined by minimizing the sum of the mean square errors of the two branches. The equation is

$$(x_k, t_k) = \mathrm{argmin} \sum_{\{i|x^i \in R_L\}} \left(y_1^i - \bar{y}_L\right)^2 + \sum_{\{i|x^i \in R_R\}} \left(y_1^i - \bar{y}_R\right)^2, \tag{10}$$

with $\bar{y}_L, \bar{y}_R \in R$, respectively, that represents the average output of all points of the left branch tree R_L and the right branch tree R_R and finds the smallest x_k corresponding t_k by traversing in sequence. In this way, we can introduce the

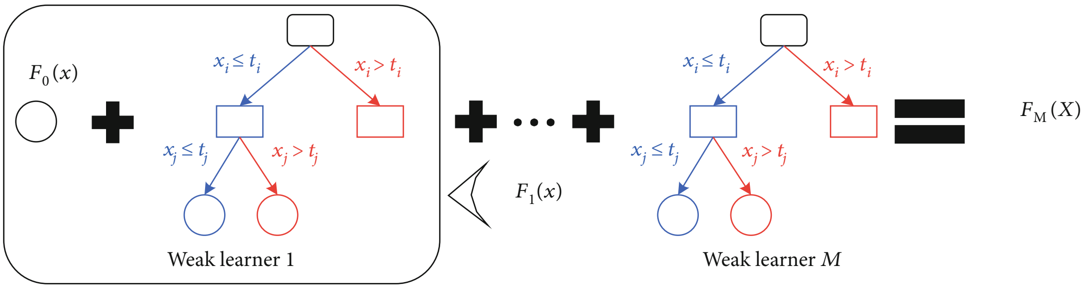


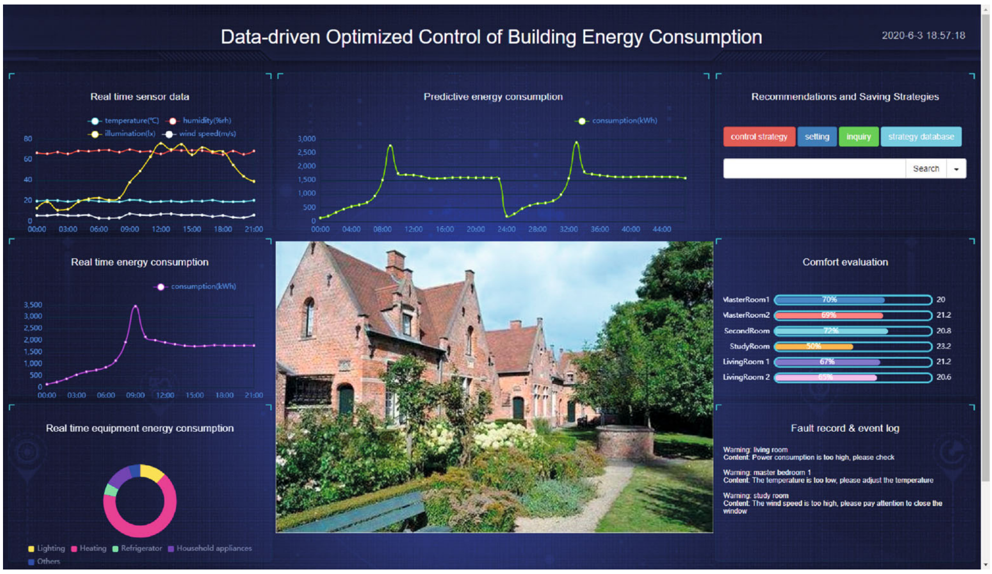
FIGURE 8: Schematic representation of the LSBoost model.

```
Require: Data $\{(x, y)\}_{i=1}^{n}$ and a Loss Function $L(y_i, F(x))$
Ensure: $F_M(x)$
    Step 1:Initialize model with a constant value:
    Step 2:for m =1 to M:
    (A) Compute
    $r_{im} = -[\partial L(y_i, F(x_i))/\partial F(x_i)]_{F(x)=F_{m-1}(x)}$ for $i = 1, \cdots, n$
    (B) Fit a regression tree to the $R_{im}$ values and create terminal regions $R_{im}$ for $j = 1 \cdots J_m$.
    (C) For $j = 1 \cdots J_m$ compute
    $\gamma_{im} = \underset{\gamma}{\operatorname{argmin}} \sum_{x_i \in R_{ij}} L(y_i, F_{m-1}(x_i) + \gamma)$
    (D) Update
    $F_m(x) = F_{m-1}(x) + \nu \sum_{j=1}^{J_m} \gamma_{im} I(x \in R_{jm})$
    Step 3:
    return $F_M(x)$
```

ALGORITHM 5. LSBoost algorithm.

TABLE 5: Complexity comparison of multiple methods.

	PCA	RT	LSBoost	IOD	DCP-RT	DCP-LSBoost
Time complexity	$O(knd)$	$O(n \log(n) d)$	$O(n \log(n) dk)$	$O(n2)$	$O(n \log(n) dk)$	$O(n \log(n) d)$
Spatial complexity	$O(kn)$	$O(D)$	$O(Dk)$	$O(nd)$	$O(kn + dk + nd)$	$O(kn + nd + D)$



Warning: living room
Content: Power consumption is too high, please check

Warning: master bedroom 1
Content: The temperature is too low, please adjust the temperature

Warning: study room
Content: The wind speed is too high, please pay attention to close the window

FIGURE 9: A snapshot of the developed building energy management system (the building image data source: [18]).

Table 6: Machine learning parameters and dimension overview.

Notation	Variable description	ML setup	Residential dimensions RT and LSBoost	Residential dimensions TDNN	Office dimensions RT and LSBoost	Office dimensions TDNN
$\tilde{x}$	Training input	All features	27	36	41	62
y	Output	Selected features	6	6	12	12
lb	Comfort zone lower border	Selected features	1	1	1	1
$\tilde{d}$	Disturbance	Selected features	8	8	12	12
t	Time	Transformed features	3	3	3	3
u	Training output/training output	Targets	6	6	20	20

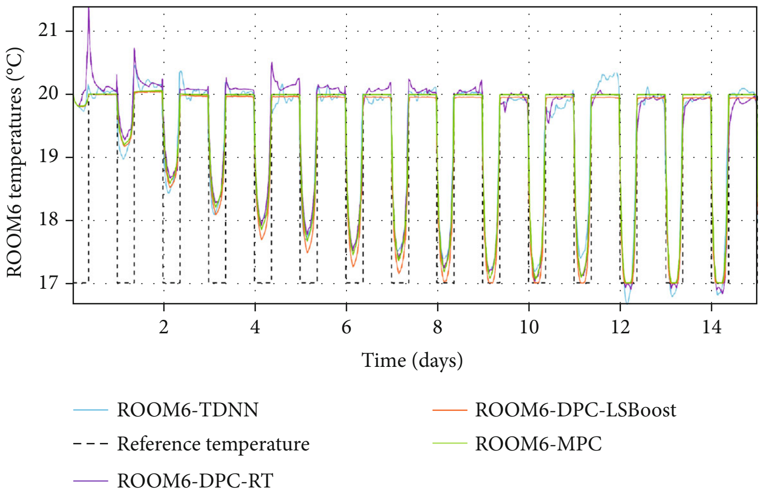


Figure 10: Comparison of the investigated temperature control performance in ROOM6.

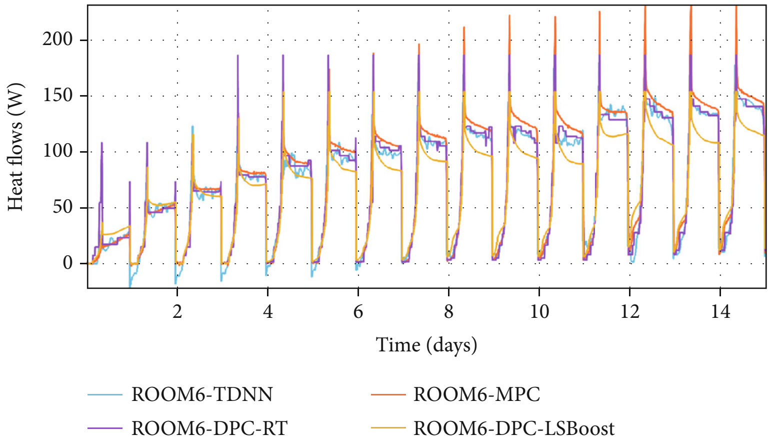


Figure 11: Comparison of the investigated controllers with respect to energy consumption in ROOM6.

weight matrix $Q \in R^{n*n}$ and introduce it into the quadratic optimization function as an adjustable parameter.

$$(x_k, t_k) = \operatorname{argmin} \sum_{\{i|x^i \in R_L\}} \left(y^i - \bar{y}_L\right)^{\mathrm{T}} Q\left(y^i - \bar{y}_L\right) + \sum_{\{i|x^i \in R_R\}} \left(y^i - \bar{y}_R\right)^{\mathrm{T}} Q\left(y^i - \bar{y}_R\right). \tag{11}$$

Both Equation (10) and Equation (11) provide two solutions to get the optimal $(x_k > t_k)$. The more times the tree is split, the more accurate the result. In terms of (9) and (10), the end conditions for building a tree are the same.

The process described above may produce good predictions on the training set but is likely to overfit the data, leading to poor test set performance. A smaller tree with fewer splits (that is, fewer regions $R_1, \cdots, R_m$) might lead to lower

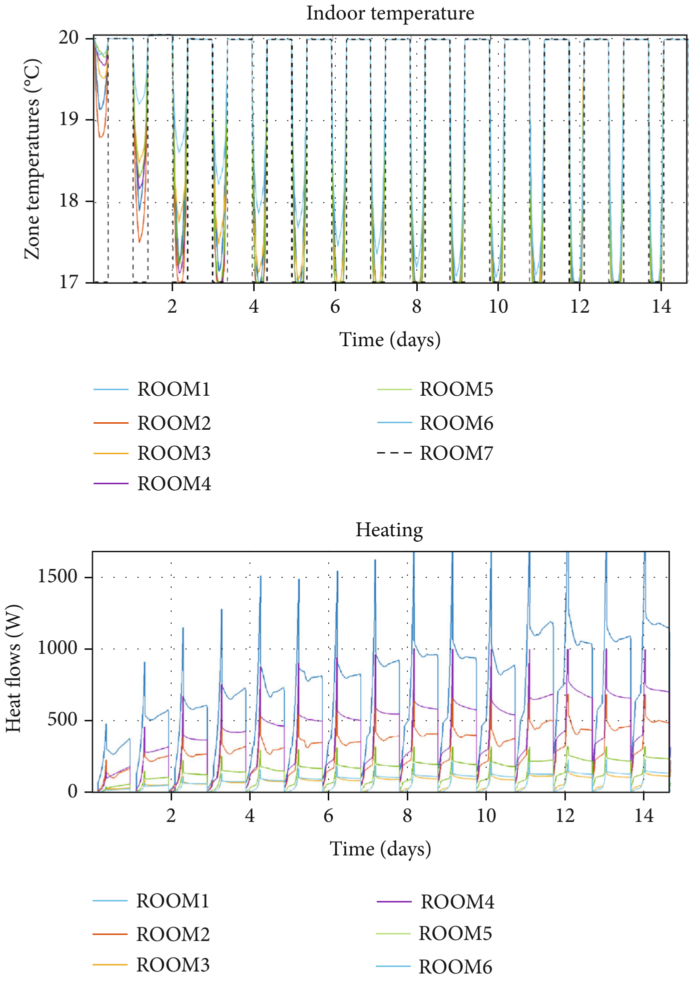


Figure 12: Control profiles of the DPC-LSBoost controller.

variance and better interpretation at the cost of a little bias. Therefore, a better strategy is to grow a very large tree T_0 and then prune it back in order to obtain a subtree. Intuitively, our goal is to select a subtree that subtree leads to the lowest test error rate.

$$T_{\min} = \min \sum_{m=1}^{|T|} \sum_{i:x_i \in R_m} \left(y^i - \bar{y}_m\right)^2 + \alpha|T|. \tag{12}$$

For each value of α, there corresponds a subtree $T \in T_0$ such that is as small as possible. Here, $|T|$ indicates the number of terminal nodes of the tree T, Rm is the rectangle (i.e., the subset of predictor space) corresponding to the mth terminal node, and y^m is the predicted response associated with R_m-, that is, the mean of the training observations in R_m. It turns out that as we increase α from zero in prune, branches get pruned from the tree in a nested and predictable fashion, so obtaining the whole sequence of subtrees as a function of α is easy. This process is summarized in Algorithm 4.

In short, when the data set is a continuous variable, the objective function of the optimal segmentation of each input feature is determined firstly. And then the input element with the lowest cost is used as the segmentation variable. In this way, we obtain the tree model $T_{\min}$ from Algorithm 4.

4.4. Design of the DPC-LSBoost Controller. Although the regression tree has the advantages of faster training and prediction speed, it is also good at obtaining the nonlinear relationship in the dataset; however, it still suffers from regression tree's poor scalability that needs to be solved. We change the regression tree structure from a single tree to multiple trees, enhancing the system's stability and robustness. Figure 7 illustrates that the enhanced tree belongs to the branch of ensemble learning and includes two types of boosting and bagging. The main focus is on reducing bias. The latter is mainly about reducing variance. Representative learning algorithms are random forest and gradient descent tree. The full name of LSBoost is least-squares boosting which is a boosting algorithm in ensemble learning. It inherits the advantages of regression trees and is developed on the classification and regression tree (CART) algorithm. Actually, a regression tree is a weak learner.

The LSBoost is an improvement on the gradient boosting decision tree (GBDT) algorithm. It has been improved from a previous classification algorithm to a regression algorithm.

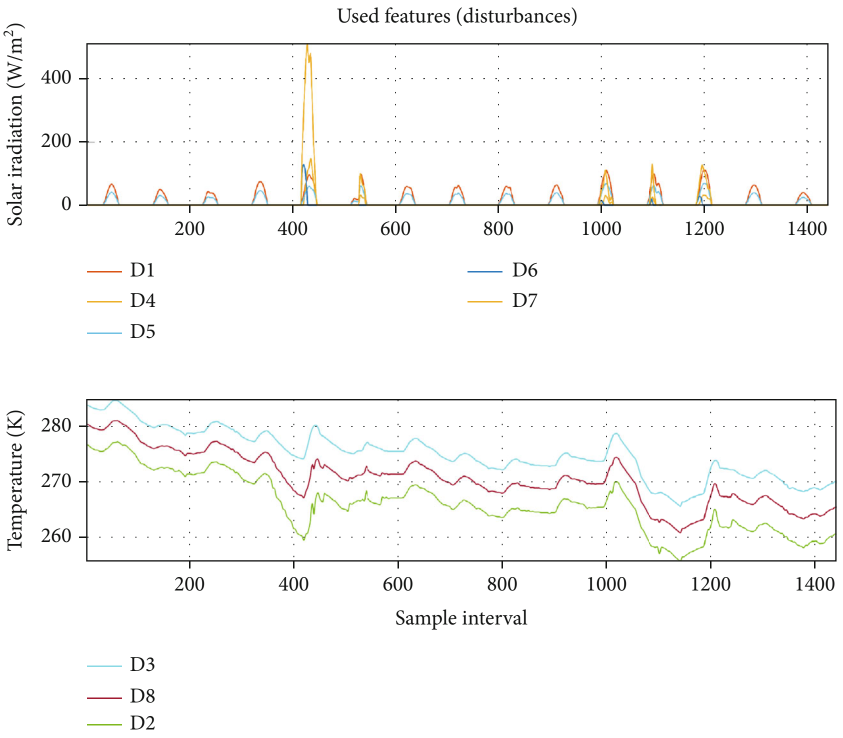


Figure 13: Selected most relevant external disturbance signals (8 signals).

Table 7: Detailed description of the disturbance signal.

Variable	Unit	Disturbance description
D1	[W]	Direct sunlight in horizontal plane Weight sun radiation
D2	[K]	Temperature between ground and sky temperature 1 Weight sun radiation
D3	[K]	Temperature between ground and sky temperature 2
D4	[W/m^2]	Direct sun radiation on vertical surface with orientation 3
D5	[W/m^2]	Diffuse sun radiation on vertical surface with orientation 3
D6	[W/m^2]	Direct sun radiation on vertical surface with orientation 4
D7	[W/m^2]	Direct sun radiation on vertical surface with orientation 5
D8	[K]	Ambient temperature

Like other boosting trees, LSBoost is training hundreds or thousands of weak learners like CART and iteratively updates the error and eventually becomes a strong learner, which is also the advantage of ensemble learning. Namely, each tree is part of the training of the current optimal. All the optimal are combined to build the strongest integrated tree. The schematic description of the LSBoost model is shown in Figure 8.

The difference between LSBoost and GBDT is that GBDT chooses to use the Gini index when the tree splits nodes, while LSBoost uses the minimum error square as the loss function at the tree split nodes as shown in Equation (13).

$$L(y_i, F(x)) = \frac{1}{2}(y_i - F(x_i))^2, \qquad (13)$$

with x_i that represents the i -th set of feature data in the training set, x_i as the observation value corresponding to x_i in the training set, and $F(x)$ as the current prediction data. The loss function setting here is not fixed. The degree of fit can be checked through the trained data, evaluate through some indicators, such as R-square and RMSE, and choose a suitable loss function for the current data.

LSBoost uses Equation 13 as a loss function to facilitate the data differentiation, simplify operations, reduce computational complexity, and reduce training time. For the entire system, it speeds up the system response and enhances robustness. Therefore, the constructed LSBoost algorithm is as follows.

Algorithm 5 shows the method of constructing the LSBoost controller by training on the data set. The input training data is the same as Equation (7), the number of trainings is M times, the loss function uses Equation (13), and the second step (B) is the RT weak learner established. v is the learning rate, ranging from 0 to 1, with a default value of 0.1. Finally, the prediction value $F_M(x)$ after M trainings is output.

4.5. Algorithm Complexity Analysis. The algorithm complexity can reflect the actual operation of the algorithm, which is divided into time complexity analysis and space complexity

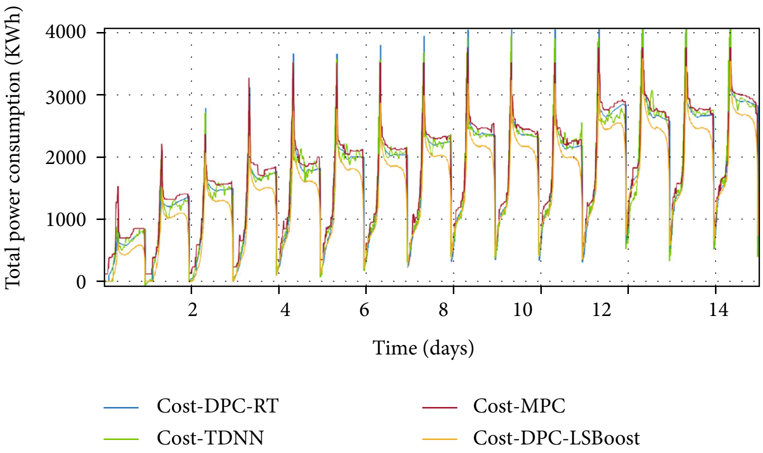


Figure 14: Comparison of total power consumption with the investigated controllers.

analysis. For this reason, the DPC based on Algorithm 1 is used to analyze the influence of RT and LSBoost algorithms on the system. Table 5 shows the comparative analysis of time complexity and spatial complexity in multiple methods.

D means the maximum depth of tree. n means the number of samples in the training set. d means the dimension of the data. k means the number of principal components.

5. Simulation and Verification

The Internet of Things platform used in this simulation experiment is a network service platform built by Nod-RED, MQTT broker, and other tools. Figure 9 demonstrates the current operation state, such as real-time sensor data display, energy consumption prediction curve, and log.

In this section, the case study's simulation results for indoor temperature control and energy consumption of residential buildings and office buildings for 15 days are demonstrated. We mainly focus on validating the proposed control strategies' performance for all investigated controllers (the TDNN, the MPC, the DPC-RT, and the DPC-LSBoost). The simulation objects selected this time are residential buildings (Section 3.1) and office buildings (Section 3.2). Based on the feature selection introduced in Section 4.4, we construct the reduced feature space x̃ dimension of the LSBoost model and the RT model to participate in training, following Equation (14).

$$n_y + 2(n_{\tilde{r}} + n_{\tilde{d}}) + n_t, \tag{14}$$

with n_y as the number of output variables, and n_t as time converted into three sinusoidal signals with different frequencies, which correspond to days, weeks, and months, respectively. $n_{\tilde{r}}$ is the reference input, and $n_{\tilde{d}}$ is the number of disturbance signals after feature selection. An overview of control variables and machine learning parameters is given in Table 6.

For the residential building, the dimension of $\tilde{x}$ is calculated by Equation (14): $*n_y = 6, n_{\tilde{r}} = 1,\ n_{\tilde{d}} = 8$,and $n_t = 3$. For the office building, the dimension of $\tilde{x}$ is calculated by Equation (14): $*n_y = 12, n_{\tilde{r}} = 1, n_{\tilde{d}} = 12$,and $n_t = 3$. Moreover, the reference input lb and the disturbance d at the current time and the next time are required during training. For more details, please see Table 6.

Table 8: Performance comparison of multiple controllers in the residential building.

Methods	Heating cost (kWh)	Cooling cost (kWh)	Total cost (kWh)	PMV viol (-)	Prediction time (s)
MPC	658.15	0	658.15	0	81.6
TDNN	660.67	1.29	661.96	1.2	11.6
DPC-RT	613.68	0	613.68	0.02	9.9
DPC-LSBoost	583.02	0.02	583.05	0	9.6

Table 9: Performance comparison of RMSE, R-square, and mean error for the RT and the LSBoost in the residential building.

	RMSE	R-square	Mean error
DPC-RT	0.0088	98.51%	52.1883
DPC-LSBoost	0.1244	99.99%	38.97

TDNN consists of one input layer, two hidden layers, and one output layer. Set the delay parameter $N = 22$, iterate 1000 times, and learn rate $\alpha = 0.01$. The main parameter of MPC is to set prediction horizon $N = 22$. For ideal training results, the dataset is divided into training set, validation set, and test set, which are 80%, 10%, and 10%, respectively.

5.1. Residential Building Simulation Analysis. This section presents the simulation results for a 6-room residential building's performance validation with the investigated controllers. The closed-loop profiles of 15 days are chosen from the simulation test. To clearly show the control effects of the building, ROOM6 is selected as the control object to analyze the temperature control and energy consumption, and

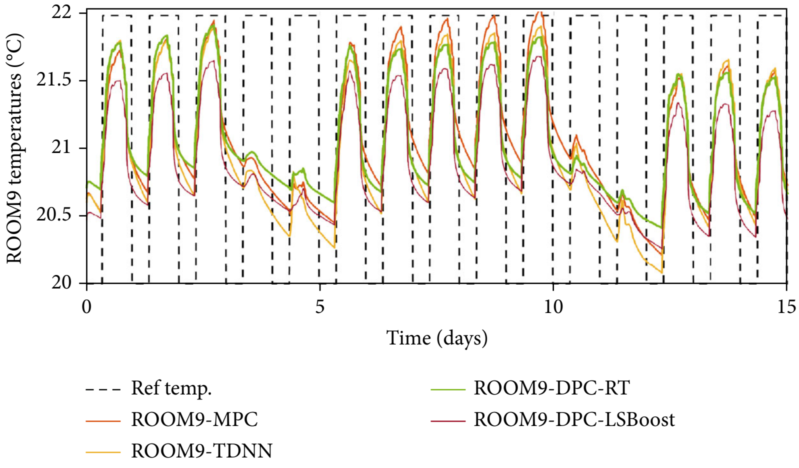


Figure 15: Comparison of the investigated temperature control performance in ROOM9.

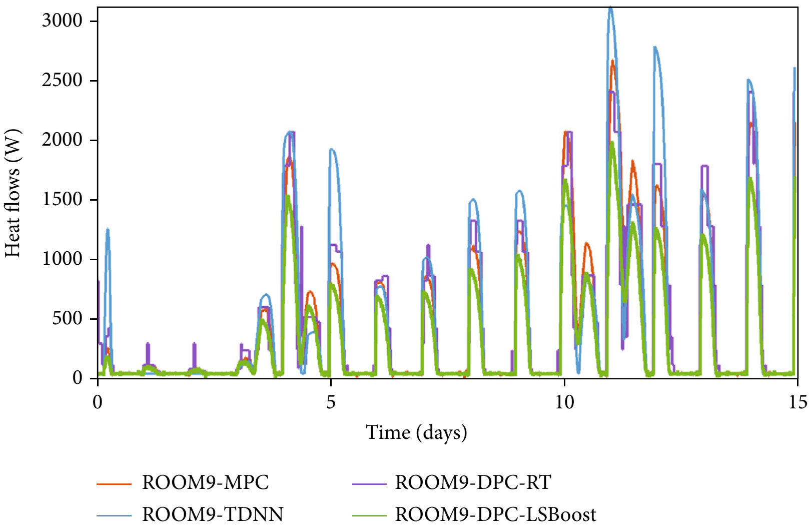


Figure 16: Comparison of the investigated controllers with respect to energy consumption in ROOM9.

the results are shown in Figures 10 and 11. The other rooms have similar behavior.

Figure 10 shows the comparison of temperature control effects with different control actions in ROOM6. It can be seen that both the MPC and the DPC-LSBoost have better control performance than the others. Under the MPC controller's action, the reference room temperature is well tracked. The temperature obtained by the TDNN fluctuates greatly, especially from the tenth to the thirteenth day. The control effect obtained by the DPC-RT is relatively general. Temperature changes abruptly on the first day, and there are more burrs in the waveform, but they will still closely follow the input. With the DPC-LSBoost, room temperature can be tracked well, make up for the DPC-RT's shortage, and achieve a good control effect. Under the MPC controller's action, the temperature change in the room is very smooth, and the temperature difference is small. The reference room temperature is well tracked to achieve a good control effect.

Figure 11 shows the effects of the investigated controllers with energy consumption. It can be seen that the DPC-LSBoost has the lowest energy cost.

Figure 12 shows the temperature and energy consumption of 6 rooms under the DPC-LSBoost control method. It can be found that the system will adjust the controller to varying degrees according to the state of the room. Compared with the second and sixth rooms, the first room will spend much energy stabilizing the temperature. Through this kind of fine management and control, each room's temperature can be controlled independently.

The indoor temperature changes are greatly affected by external disturbances. For the investigated residential buildings, there are a total of 44 external disturbances. Through the feature selection, the eight most relevant features are selected. Figure 13 shows the eight disturbances profiles. Three of them are the external ambient temperature (K), and five are the effects of solar radiation on various rooms in the house. The abscissa is a time interval of 15 days, a total

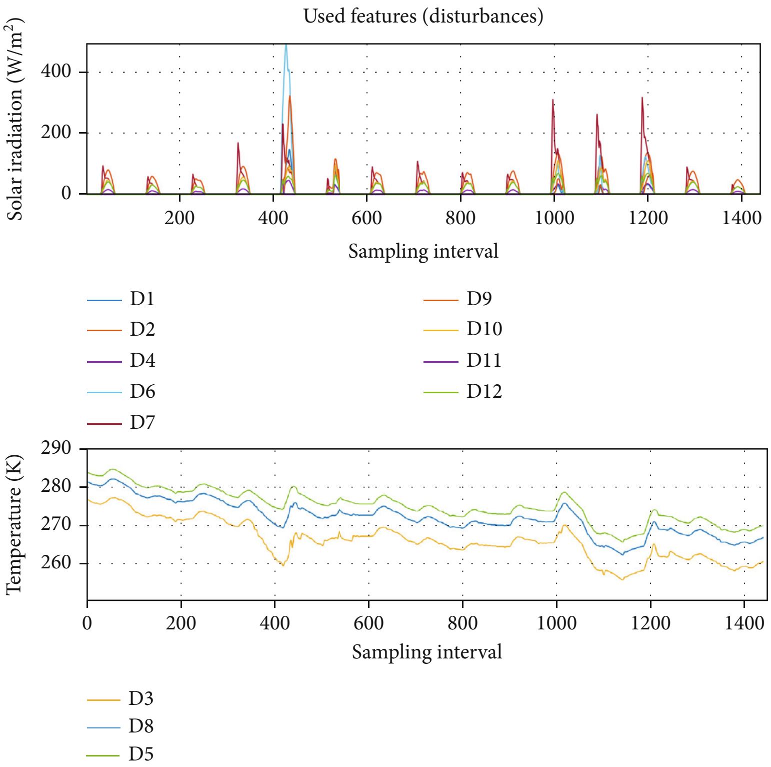


FIGURE 17: Selected most relevant external disturbance signals (12 signals).

TABLE 10: Detailed description of selected disturbance features.

Variable	Unit	Disturbance description
D1	[W/m^2]	Direct sun radiation on vertical surface with orientation 1
D2	[W/m^2]	Diffuse sun radiation on vertical surface with orientation 1
D3	[K]	Weight sun radiation temperature between ground and sky temperature 1
D4	[W/m^2]	Diffuse sun radiation on vertical surface with orientation 2
D5	[K]	Weight sun radiation temperature between ground and sky temperature 2
D6	[W/m^2]	Direct sun radiation on vertical surface orientation 3
D7	[W/m^2]	Diffuse sun radiation on vertical surface with orientation 3
D8	[K]	Weight sun radiation temperature between ground and sky temperature 3
D9	[W/m^2]	Direct sun radiation on vertical surface with orientation 4
D10	[W/m^2]	Diffuse sun radiation on vertical surface with orientation 4
D11	[W/m^2]	Direct sun radiation on vertical surface with orientation 5
D12	[W/m^2]	Diffuse sun radiation on vertical surface with orientation 6

of 1440 time samples. Table 7 describes the specific information of the eight disturbances D1-D8.

For power consumption, a holistic analysis is required. Figure 14 shows the comparison results of the total power consumption of the four controllers for 15 days. Figure 14 shows that the TDNN consumes the most energy, followed by the MPC and the RT, and the lowest energy consumption is the DPC-LSBoost. The detailed comparison results are shown in Table 8. Power consumption is analyzed from five dimensions: heating cost (kWh), cooling cost (kWh), total cost (kWh), PMV, and prediction time (s). It can be seen from the table that the overall power consumption of the DPC-LSBoost is the least, which is reduced by 78.909 kWh compared to the TDNN, and the overall energy consumption is reduced by 11.92% compared to the TDNN. With the prediction time, the DPC-LSBoost has the shortest time cost. The quantitative comparison of RMSE, *R*-square, and mean error for model accuracy is summarized in Table 9. It is observed that the DPC-LSBoost has the better model fitting capability.

5.2. Office Building Simulation Analysis. The analysis methods for office buildings and residential buildings are consistent. However, the temperature setting is between 20°C and 22°C in the office building.

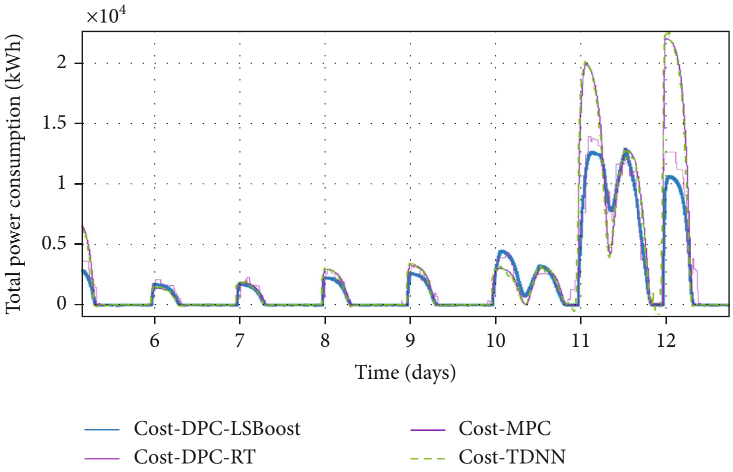


Figure 18: Comparison of the investigated controllers with respect to the total energy consumption.

Table 11: Performance comparison of multiple controllers in the office building.

Methods	Heating cost (kWh)	Cooling cost (kWh)	Total cost (kWh)	PMV viol (-)	Prediction time (s)
MPC	713.37	0.02	713.39	0	302.2
TDN	728.31	12.58	740.8	0	9.2
DPC-RT	582.86	0.0	582.87	0	8.7
DPC-LSBoost	514.2	1.41	515.61	0	7.5

Similar to residential buildings, the TDNN, the MPC, the DPC-RT, and the DPC-LSBoost methods are used for comparative simulation verification. We chose ROOM9 as the control object. The 15-day comparison results are shown in Figures 15and 16.

Figure 15 shows the room temperature controlled by all control strategies. The tracking effect is not desirable from the results, although they are all within the comfort zone. Figure 16 shows the energy consumption comparison of the four control methods. It is also found that the daily energy consumption of the MPC and the TDNN is relatively high, but the temperature change is not large, and even there is a certain energy loss due to the DPC-RT algorithm's single tree structure, which contributes to the control satisfaction violation. The DPC-LSBoost controller consumes the lowest daily energy and makes the room temperature more stable with little fluctuation.

The indoor temperature change shown in Figure 3, which is greatly affected by external disturbances. The 12 most relevant features are selected from 301 external disturbances by *FS* for the investigated office buildings. Figure 17 below shows the 12 perturbed features. Three of them are outside ambient temperature (K), and there are 9 solar radiation effects on each room. Table 10 gives detailed information on the 12 interferences D1-D12.

Table 12: Performance comparison of RMSE, *R*-square, and mean error for the RT and the LSBoost in the office building.

	RMSE	*R*-square	Mean error
DPC-RT	0.9877	93.68%	17.8487
DPC-LSBoost	0.0328	99.99%	0.8229

Figure 18 shows the total power consumption with the investigated controllers. The TDNN and the MPC almost have similar control profiles with a higher peak value of the curve, especially from the tenth day, the power consumption began to soar, and the power consumption reached its peak in 12 days. Table 11 shows the comparative analysis of the control energy consumption and predicted time of the four controllers. It is found that the TDNN has the highest energy consumption, which is as high as 740.89 kWh. Compared with the TDNN, both the DPC-RT and the DPC-LSBoost are reduced significantly, which are 582.87 kWh and 515.61 kWh, respectively. It can be seen from the table that the overall power consumption of the DPC-LSBoost is the least, which is 225.28 kWh reduced compared to the TDNN, and the overall energy consumption is reduced by 30.4% compared to the TDNN. From the 15-day simulation test, the DPC-LSBoost takes the shortest prediction time comparing with other algorithms.

The quantitative comparison with three indicates (RMSE, *R*-square, and mean error) is demonstrated in Table 12. The same conclusion is achieved that the LSBoost has the better model fitting capability.

6. Conclusion and Prospect

This paper reports an innovative study combining the data-driven predictive control strategy with a complex cloud SCADA-based building energy management platform, which attempts to standardize communication protocols and data formats and further implement advanced control strategies. The platform also provides useful data representations to different stakeholders (end-user, building energy manager, and/or operator), enabling the platform flexibility and scalability.

We present two algorithms, based on RT and LSBoost, to create control-oriented models for the DPC. Moreover, an efficient feature selection method, which depends on the principal component analysis and the importance of disturbance variables, is leveraged to decrease the model's dimension and further alleviate the DPC optimization problem's complexity. We then apply the DPC to two different case studies for energy consumption in residential and office buildings. The numerical simulation shows that the DPC-LSBoost provides lower energy consumption while maintaining the required thermal comfort compared to the MPC, the TDNN, and the DPC-RT. With the same environmental comfort demand, compared with the TDNN, the peak power consumption with the DPC- LSBoost can be reduced by 11.92% and 30.4%, even compared to the DPC-RT 4.99% and 11.54% that are achieved. These advantages make the DPC-LSBoost an attractive tool for large-scale cyber-physical energy systems to reduce energy consumption. Also, in the context of prediction time, comparing with the MPC, the prediction time of the DPC-LSBoost is reduced by 72 s and 294.7 s, respectively.

Future work will focus on the combination of IoT with DPC (IoT-DPC), which will apply to more complex buildings. IoT-DPC applications are not limited to building energy management and include critical infrastructures such as water supply networks, district heating, and cooling.

Data Availability

(1) The building data of Hollandsch Huys is shown in chapter 2 of the report from the links as https://lirias.kuleuven.be/retrieve/453505 and https://github.com/drgona/BeSim/tree/master/buildings/HollandschHuys, and (2) the building data of the residential building is in the third chapter of this paper, linked as https://www.sciencedirect.com/science/article/pii/S0306261918302903 and https://github.com/drgona/BeSim/tree/master/buildings/Reno.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Key Fund of Shaanxi Province Natural Science Basic Research Program (2019LZ-06) and the Key Project of National Internet of Things Integrated Innovation and Integration (2018-470).

References

[1] M. M. Abdelrahman, S. Zhan, and A. Chong, *A three-tier architecture visual-programming platform for building-lifecycle data management*, SimAUD, Preprint, 2020.

[2] J. K. W. Wong, H. Li, and S. W. Wang, "Intelligent building research: a review," *Automation in Construction*, vol. 14, no. 1, pp. 143–159, 2005.

[3] International Energy Agency taff, *Transition to Sustainable Buildings: Strategies and Opportunities to 2050*, OECD, 2013.

[4] H. Allcott and S. Mullainathan, "Behavior and energy policy," *Science*, vol. 327, no. 5970, pp. 1204-1205, 2010.

[5] J. Drgoňa, J. Arroyo, I. C. Figueroa et al., "All you need to know about model predictive control for buildings," *Annual Reviews in Control*, 2020.

[6] F. Al-Turjman, A. Kamal, M. H. Rehmani, A. Radwan, and A.-S. K. Pathan, "The green internet of things (g-iot)," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 6059343, 2 pages, 2019.

[7] M. Amjad and F. Iradat, "An active network-based open framework for iot," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 5741708, 8 pages, 2019.

[8] J. Paek, O. Gnawali, M. A. M. Vieira, and S. Hao, "Embedded iot systems: network, platform, and software," *Mobile Information Systems*, vol. 2017, Article ID 5921523, 2 pages, 2017.

[9] G. Alce, A. Espinoza, T. Hartzell, S. Olsson, D. Samuelsson, and M. Wallergård, "Ubicompass: an iot interaction concept," *Advances in Human-Computer Interaction*, vol. 2018, 12 pages, 2018.

[10] S. Choudhury, Q. Ye, M. Dong, and Q. Zhang, "Iot big data analytics," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.

[11] S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: a study of infrastructure as a service (IAAS)," *International Journal of Engineering and Information Technology*, vol. 2, no. 1, pp. 60–63, 2010.

[12] T. Dillon, C. Wu, and E. Chang, "Cloud computing: issues and challenges," in *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, pp. 27–33, Perth, WA, Australia, April 2010.

[13] P. Mell and T. Grance, *The Nist Definition of Cloud Computing*, Communications of the ACM, 2011.

[14] M. del Mar Castilla, J. D. Álvarez, F. Rodrguez, and M. Berenguel, *Comfort control in buildings, chapter 4*, Springer, Berlin, Germany, 2014.

[15] D. Gyalistras, M. Gwerder, F. Oldewurtle, C. Jones, and M. Morari, "Analysis of energy savings potentials for integrated room automation," in *Clima-RHEVA World Congress*, number CONF, 2010.

[16] K. W. Roth, D. Westphalen, J. Dieckmann, S. D. Hamilton, and W. Goetzler, *Energy Consumption Characteristics of Commercial Building Hvac Systems Volume III: Energy Savings Potential*, US Department of Energy, 2002.

[17] C. Aghemo, J. Virgone, G. V. Fracastoro et al., "Management and monitoring of public buildings through ict based systems: control rules for energy saving with lighting and hvac services," *Frontiers of Architectural Research*, vol. 2, no. 2, pp. 147–161, 2013.

[18] J. Drgoňa, D. Picard, M. Kvasnica, and L. Helsen, "Approximate model predictive building control via machine learning," *Applied Energy*, vol. 218, pp. 199–216, 2018.

[19] H. E. Mechri, A. Capozzoli, and V. Corrado, "Use of the anova approach for sensitive building energy design," *Applied Energy*, vol. 87, no. 10, pp. 3073–3083, 2010.

[20] Y. Ma, F. Borrelli, B. Hencey, B. Coffey, S. Bengea, and P. Haves, "Model predictive control for the operation of building cooling systems," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 796–803, 2011.

[21] F. Oldewurtel, A. Parisio, C. N. Jones et al., “Use of model predictive control and weather forecasts for energy efficient building climate control,” *Energy and Buildings*, vol. 45, pp. 15–27, 2012.

[22] F. Oldewurtel, A. Parisio, C. N. Jones et al., “Energy efficient building climate control using stochastic model predictive control and weather predictions,” in *Proceedings of the 2010 American Control Conference*, pp. 5100–5105, Baltimore, MD, USA, June-July 2010.

[23] J. Široký, F. Oldewurtel, J. Cigler, and S. Prívara, “Experimental analysis of model predictive control for an energy efficient building heating system,” *Applied Energy*, vol. 88, no. 9, pp. 3079–3087, 2011.

[24] Z. Váňa, J. Cigler, J. Široký, E. Žáčeková, and L. Ferkl, “Model-based energy efficient control applied to an office building,” *Journal of Process Control*, vol. 24, no. 6, pp. 790–797, 2014.

[25] A. Jain, M. Behl, and R. Mangharam, “Data predictive control for building energy management,” in *2017 American Control Conference (ACC)*, pp. 44–49, Seattle, WA, USA, May 2017.

[26] J. R. New, J. Sanyal, M. Bhandari, and S. Shrestha, “Autotune e + building energy models,” *Proceedings of SimBuild*, vol. 5, no. 1, pp. 270–278, 2012.

[27] A. Afram, F. Janabi-Sharifi, A. S. Fung, and K. Raahemifar, “Artificial neural network (ann) based model predictive control (mpc) and optimization of hvac systems: a state of the art review and case study of a residential hvac system,” *Energy and Buildings*, vol. 141, pp. 96–113, 2017.

[28] M. Macarulla, M. Casals, N. Forcada, and M. Gangolells, “Implementation of predictive control in a commercial building energy management system using neural networks,” *Energy and Buildings*, vol. 151, pp. 511–519, 2017.

[29] P. M. Ferreira, A. E. Ruano, S. Silva, and E. Z. E. Conceicao, “Neural networks based predictive control for thermal comfort and energy savings in public buildings,” *Energy and Buildings*, vol. 55, pp. 238–251, 2012.

[30] A. Jain, R. Mangharam, and M. Behl, “Data predictive control for peak power reduction,” in *BuildSys '16: Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, pp. 109–118, New York, NY, USA, November 2016.

[31] F. Smarra, A. Jain, T. de Rubeis, D. Ambrosini, A. D’Innocenzo, and R. Mangharam, “Data-driven model predictive control using random forests for building energy optimization and climate control,” *Applied Energy*, vol. 226, pp. 1252–1272, 2018.

[32] J. Drgoňa, D. Picard, and L. Helsen, “Cloud-based implementation of white-box model predictive control for a geotabs office building: a field test demonstration,” *Journal of Process Control*, vol. 88, pp. 63–77, 2020.

[33] T. Yang, D. Clements-Croome, and M. Marson, “Building energy management systems,” *Encyclopedia of Sustainable Technologies*, vol. 36, pp. 291–309, 2017.

[34] P. Radecki and B. Hencey, “Online building thermal parameter estimation via unscented kalman filtering,” in *2012 American Control Conference (ACC)*, pp. 3056–3062, Montreal, QC, Canada, June 2012.

[35] D. Picard, F. Jorissen, and L. Helsen, “Methodology for obtaining linear state space building energy simulation models,” in *Proceedings of the 11th International Modelica Conference*, Versailles, France, September 2015.

[36] E. Van Kenhove, J. Laverge, W. Boydens, and A. Janssens, “Design optimization of a geotabs office building,” *Energy Procedia*, vol. 78, pp. 2989–2994, 2015.

[37] D. Picard and L. Helsen, *Report on the Building Energy Simulation Models of an Office Building, a Retirement Home, a School, and a Block of Flats*, KU Leuven, 2017.

[38] B. W. Olesen and K. C. Parsons, “Introduction to thermal comfort standards and to the proposed new version of en iso 7730,” *Energy and Buildings*, vol. 34, no. 6, pp. 537–548, 2002.

[39] G. L. Lei, Y. C. Yu, Z. P. Liu, and X. M. Liu, “Human comfort index forecast of Nanchang,” *Jiangxi Meteorological Science & Technology*, vol. 22, no. 2, pp. 40-41, 1999.

[40] F. Song, Z. Guo, and D. Mei, “Feature selection using principal component analysis,” in *2010 International Conference on System Science, Engineering Design and Manufacturing Informatization*, pp. 27–30, Yichang, China, November 2010.