Overview of Cyber-Physical Temperature Estimation in Smart Buildings: From Modeling to Measurements

(Invited Paper)

Xiaoming Chen, Xin Li
Electrical and Computer Engineering Department
Carnegie Mellon University
Pittsburgh, PA 15213, USA
Email: {xchen3,xinli}andrew.cmu.edu

Sheldon X.-D. Tan
Department of Electrical and Computer Engineering
University of California, Riverside
CA 92521, USA
Email: stan@ece.ucr.edu

***Abstract*—Smart buildings are playing a more important role in everyday lives of people. One important goal of creating smart buildings is to offer highly comfortable services to the occupants at the lowest cost. In-building temperature modeling and measurement is a critical task to facilitate high-quality service with low cost. In this paper, we give an overview of state-of-the-art cyber-physical temperature estimation methods in smart buildings. For temperature modeling, by adopting an orthogonal matching pursuit algorithm, the modeling cost and accuracy can be both improved compared with the conventional least-squares fitting method. For temperature measurement, by combining temperature modeling with few sensor measurements using a Bayesian model fusion framework, the spatial temperature distribution can be accurately estimated.**

I. Introduction

Nowadays, as people's demands in the comfort level and energy efficiency increase speedily, advanced technologies greatly promote the development of smart buildings. Smart buildings are playing a more important role in everyday lives of people, and people are living in a more comfortable and smart environment. Smart buildings can offer numerous opportunities to improve the energy efficiency and comfort level through efficient temperature and energy control systems. Offering highly comfortable services to the occupants at the lowest energy consumption and environmental impact has become one of most important goals of modern smart buildings.

In order to achieve a high energy efficiency and provide a high thermal comfort level to the occupants, we first need to create a temperature and energy model to predict the in-building temperature and energy consumption. Building simulation is widely used at the design stage to achieve this goal [1]. Through building simulation, we can achieve many advantages for designing energy-efficient smart buildings. For instance, temperature and energy models can be created, so that we can identify the most important factors that influence the building performance most. This step is usually called a sensitivity and uncertainty analysis [2]. Through the sensitivity and uncertainty analysis, we can investigate various strategies to improve the energy efficiency. However, models created by building simulation are prior models which are based on prior knowledge obtained at the design stage. It is difficult to consider real-time and real-world information (e.g., weather and environment variations) in building simulation. On the contrary, sensor-based real-time temperature monitoring is a practical solution to this problem.

Real-time temperature monitoring and control in smart buildings usually involves an automatic temperature management system. As heating and cooling account for 30% to 50% of the total building energy consumption [3], efficiently implementing such a smart temperature management system can significantly reduce the building energy cost, and thus, the energy efficiency can be improved. Apparently, the efficiency of such a system heavily relies on monitored or measured in-building temperature. As a result, accurate temperature measurement or monitoring is a critical task for smart buildings.

In modern smart buildings, smart temperature management is usually implemented by a so-called cyber-physical system (CPS) [4]. CPS is a tight integration of computation, communication, and control to form an interaction loop between the physical world and cyber elements. Typically, the physical world is sensed by sensors and the sensory data are processed by the cyber elements. Decisions are made and then sent to actuators to control the physical world. In recent years, there emerges several researches that utilized the CPS technique such as Internet of Things (IoT) and wireless sensor network (WSN) to monitor or measure the in-building temperature [5], [6]. There is a shortage with these IoT/WSN-based approaches. For a large building, a wireless network with a large number of sensors is required, which suffers from high cost and high complexity. We also need to carefully design the network architecture, network protocol, data processing algorithm, etc. to make the system highly scalable. In addition, sensors can age and have drifts [7], so a large network with a large number of sensors also suffers from high maintenance cost.

In this paper, we give an overview of state-of-the-art cyber-physical temperature modeling and estimation approaches for smart buildings. First, by adopting a sparse regression method

978-1-4673-9955-5/16/$31.00 ©2016 IEEE

named orthogonal matching pursuit (OMP), the modeling cost and accuracy can be both improved compared with the conventional least-squares (LS) fitting method [8]. We will further show that, by adopting a Bayesian model fusion (BMF) method, the prior temperature model can be combined with few sensor measurements to accurately predict the in-building temperature [9]. By combining prior models and sensor measurements, we can take advantages of both the simulation-based and IoT/WSN-based approaches, and thus, low cost and high accuracy are both achieved.

The rest of this paper is organized as follows. In Section II, we present the background about temperature modeling in smart buildings. We introduce the sparse regression algorithm for temperature modeling in Section III. The BMF method for temperature estimation is explained in Section IV. Finally, we conclude in Section V.

II. Background

In this section, we will review the background about temperature modeling in smart buildings, covering two important topics: (i) problem formulation, and (ii) uncertainty/sensitivity analysis.

A. Problem Formulation

Given a building, its in-building temperature at different locations depends on many factors, including indoor factors such as the heating/cooling power and outdoor factors such as the weather and atmospheric temperature. We use a vector $\mathbf{X} = [x_0, x_1, x_2, \cdots, x_K]^T$ ($x_0 = 1$) to denote all these independent factors that may affect the in-building temperature, where K is the number of independent factors. Let N be the number of locations in the building. The temperature of location n ($n = 1, 2, \cdots, N$), $t^{(n)}$, can be expressed as a function of all these independent factors, i.e.,

$$t^{(n)} = t^{(n)}(x_0, x_1, \cdots, x_K). \tag{1}$$

Please note that $\mathbf{X}$ includes both global factors for all locations and local factors for each location. In practice, the following linear model is widely used to approximate the temperature of each location [10]–[13]:

$$t^{(n)} = \sum_{k=0}^{K} \alpha_k^{(n)} x_k = \alpha_0^{(n)} + \sum_{k=1}^{K} \alpha_k^{(n)} x_k, \tag{2}$$

where $\{\alpha_k^{(n)}; k = 0, 1, \cdots, K\}$ are the model coefficients of the nth location. For convenience, Eq. (2) can be converted into a matrix form for all locations:

$$\mathbf{T} = \mathbf{A}\mathbf{X}, \tag{3}$$

where $\mathbf{A} \in \mathbb{R}^{N \times (K+1)}$ is a matrix containing all the model coefficients, i.e., $A_{n,k} = \alpha_k^{(n)}$, and $\mathbf{T} = [t^{(1)}, t^{(2)}, \cdots, t^{(N)}]^T$ is the temperature vector. The model coefficients $\mathbf{A}$ can be approximately solved by linear regression methods [14] such as LS fitting [15] based on simulation data, which will be introduced in the below.

B. Uncertainty/sensitivity Analysis

In the design stage, we cannot exactly know many important factors (e.g., temperature/weather, properties of materials, etc.) that affect the in-building temperature of a building. In this case, all the independent factors must be modeled as a set of random variables, instead of deterministic values. Therefore, the in-building temperature cannot be deterministically solved in the design stage, posing the need of uncertainty/sensitivity analysis. The purpose of uncertainty/sensitivity analysis is to determine the temperature variation of each location and the most important factors contributing to it.

Under the the linear model assumption of Eq. (2), the sensitivity of each factor x_k ($k = 0, 1, \cdots, K$) for each location n ($n = 1, 2, \cdots, N$) is just the model coefficient $\alpha_k^{(n)}$. Consequently, to perform uncertainty/sensitivity analysis, we need to solve the model coefficient matrix $\mathbf{A}$. We assume that the prior distribution of each factor x_k ($k = 1, 2, \cdots, K$) is known in the design stage, which is denoted as $pdf(x_k)$, where $pdf(\cdot)$ means the probability density function. The prior distribution of each factor can be any distribution, e.g., normal distributions, uniform distributions, triangular distributions, etc. Typically, an uncertainty/sensitivity analysis flow includes the following three major steps [16].

1) According to the defined prior distributions of all the independent factors, generate a number of (say, S, $S > K + 1$) samples $\mathbf{X}_{(1)}, \mathbf{X}_{(2)}, \cdots, \mathbf{X}_{(S)}$ by design of experiments [17] such as Latin hypercube sampling [18].
2) Perform S independent building simulations for these S samples. Collect the temperature data of each location of the S samples from the simulation results.
3) Solve the model coefficients for each location by LS fitting [15] based on the following over-determined linear system:

$$\mathbf{Y}[\alpha_0^{(n)}, \alpha_1^{(n)}, \cdots, \alpha_K^{(n)}]^T = [t_{(1)}^{(n)}, t_{(2)}^{(n)}, \cdots, t_{(S)}^{(n)}]^T, \tag{4}$$

where $\mathbf{Y} = [\mathbf{X}_{(1)}, \mathbf{X}_{(2)}, \cdots, \mathbf{X}_{(S)}]^T \in \mathbb{R}^{S \times (K+1)}$ is the regressor matrix containing all the sampling values, which is generated in step (1), and $t_{(s)}^{(n)}$ is the simulated temperature of the nth location for the sample $\mathbf{X}_{(s)}$, which is collected in step (2).

III. Temperature Modeling by Sparse Regression

In the conventional LS fitting method, the number of samples (i.e., S) must be larger than the number of independent factors (i.e., $K+1$) to form an over-determined linear system. As a result, if K is large, a large number of samples must be generated and simulated in order to train a high-dimensional regression model. This requirement leads to a prohibitively high computational cost due to a large number of building simulations.

In most practical applications, there are only a small number of important factors that can significantly affect the in-building temperature. In other words, for each location, most components of $\alpha_0^{(n)}, \alpha_1^{(n)}, \cdots, \alpha_K^{(n)}$ are close to zero. As a result, we can adopt sparse regression methods to solve the model coefficients from a small set of simulation samples. In this paper, we introduce an L_0-norm regularization method named OMP [8], [19]–[21], one of the well-known sparse regression methods, to find the sparse solution of the model coefficients.

The L_0-norm regularization method converts the solving of the linear system in Eq. (4) into the following optimization problem:

$$\begin{aligned} \text{minimize} \quad & ||\mathbf{r}||_2 = ||\mathbf{Y}\boldsymbol{\alpha}^{(n)} - \mathbf{t}^{(n)}||_2, \\ \text{subject to} \quad & ||\boldsymbol{\alpha}^{(n)}||_0 \le \lambda, \end{aligned} \tag{5}$$

where $\boldsymbol{\alpha}^{(n)} = [\alpha_0^{(n)}, \alpha_1^{(n)}, \cdots, \alpha_K^{(n)}]^T$, $\mathbf{t}^{(n)} = [t_{(1)}^{(n)}, t_{(2)}^{(n)}, \cdots, t_{(S)}^{(n)}]^T$, and $||\cdot||_0$ and $||\cdot||_2$ are the L_0-norm and L_2-norm of a vector, respectively. L_0-norm means the number of nonzeros in a vector, which also means the sparsity of the vector. By constraining the L_0-norm of $\boldsymbol{\alpha}^{(n)}$, the optimization in Eq. (5) attempts to find a sparse solution $\boldsymbol{\alpha}^{(n)}$ such that the L_2-norm of the residual is minimized. The parameter λ in Eq. (5) explores the tradeoff between the sparsity of $\boldsymbol{\alpha}^{(n)}$ and the modeling accuracy. A large λ usually results in a small residual. However, a small residual does not necessarily mean a small modeling error, because the model may be over-fitted for the given samples. In practice, the optimal λ can be determined by the cross-validation technique [14]. It has been proved that the optimization problem in Eq. (5) is non-deterministic polynomial-time (NP) hard [19], so it is difficult to solve. OMP is an efficient heuristic algorithm to solve it. Since Eq. (5) has exactly the same form for each location, in what follows, we will ignore the location index (n), which will not lead to confusion.

A. Orthogonal Matching Pursuit

In this subsection, we will describe the details of OMP, including a basic theory of factor selection and an iterative flow to solve the model coefficients.

1) Factor Selection: The purpose of OMP is to identify a subset of factors which have the largest impact on the temperature t. OMP uses the inner product between the temperature t and factor x_i to measure the importance of x_i $(i = 0, 1, \cdots, K)$. If all the independent factors are appropriately normalized, the inner product between t and x_i exactly equals the model coefficient α_i, i.e.,

$$\langle t, x_i \rangle = \sum_{k=0}^{K} \alpha_k \langle x_i, x_k \rangle = \alpha_i. \tag{6}$$

This is the reason why the inner product can be adopted as a good criterion to measure the importance of each factor.

In practice, we do not know the analytical form of t so the inner product in Eq. (6) should be numerically calculated from a set of samples. Based on the generated random samples, the inner product is approximated by

$$\langle t, x_i \rangle = \frac{1}{S}\mathbf{Y}_i^T \mathbf{t}, \tag{7}$$

where $\mathbf{Y}_i \in \mathbb{R}^{S\times 1}$ represents the ith column of the matrix $\mathbf{Y}$.

2) Iterative Algorithm: OMP applies an iterative algorithm to solve the optimization problem in Eq. (5). At each iteration step, it finds a factor which is most correlated with the temperature by evaluating the inner product defined by Eq. (7). The iteration continues until λ important factors are found. Finally, the temperature t is approximated by the selected factors.

At the beginning of the iteration process, the first important factor x_{s1} is selected such that $|\langle t, x_{s1}\rangle|$ is the largest over the set $\{|\langle t, x_k\rangle|; k = 0, 1, \cdots, K\}$. Once x_{s1} is chosen, t is approximated by x_{s1}:

$$t \approx \alpha_{s1} x_{s1}, \tag{8}$$

where α_{s1} is calculated from the following LS fitting problem:

$$\underset{\alpha_{s1}}{\text{minimize}} \; ||\alpha_{s1}\mathbf{Y}_{s1} - \mathbf{t}||_2. \tag{9}$$

Next, OMP removes the component $\alpha_{s1}x_{s1}$ from t and calculates the residual:

$$\mathbf{r} = \mathbf{t} - \alpha_{s1}\mathbf{Y}_{s1}. \tag{10}$$

The second important factor is selected from all the unselected factors such that it corresponds to the largest inner product magnitude. Let x_{s2} be the second important factor. Now, t is approximated by both x_{s1} and x_{s2}:

$$t \approx \alpha_{s1}x_{s1} + \alpha_{s2}x_{s2}, \tag{11}$$

where α_{s1} and α_{s2} are solved from the following LS fitting problem:

$$\underset{\alpha_{s1}, \alpha_{s2}}{\text{minimize}} \; ||\alpha_{s1}\mathbf{Y}_{s1} + \alpha_{s2}\mathbf{Y}_{s2} - \mathbf{t}||_2. \tag{12}$$

The aforementioned iteration process will continue until λ factors are selected. The model coefficients corresponding to

Algorithm 1 The OMP algorithm.

Input: The linear system $\mathbf{Y}\boldsymbol{\alpha} = \mathbf{t}$ and the L_0-norm constraint λ.

Output: The model coefficients $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \cdots, \alpha_K]^T$.

Initialize the index set $\mathcal{S} = \emptyset$, and the residual $\mathbf{r} = \mathbf{t}$.

for $p = 1$ to λ **do**

  Select the index s such that $|\mathbf{Y}_s^T\mathbf{r}|$ takes the largest value over the set $\{|\mathbf{Y}_i^T\mathbf{r}|; i \notin \mathcal{S}\}$. Update the index set $\mathcal{S} = \mathcal{S} \cup \{s\}$.

  Solve the following LS fitting problem:

$$\underset{\alpha_i, i\in\mathcal{S}}{\text{minimize}} \; ||\sum_{i\in\mathcal{S}} \alpha_i \mathbf{Y}_i - \mathbf{t}||_2, \tag{13}$$

  and then approximate t by the selected factors:

$$t \approx \sum_{i\in\mathcal{S}} \alpha_i x_i. \tag{14}$$

  Update the residual $\mathbf{r} = \mathbf{t} - \sum_{i\in\mathcal{S}} \alpha_i \mathbf{Y}_i$.

end for

For $i \notin \mathcal{S}$, set $\alpha_i = 0$.

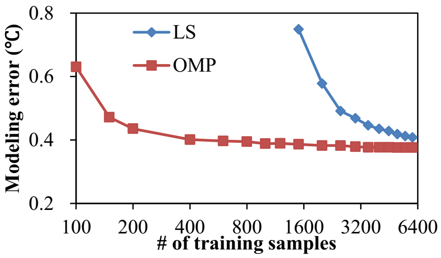


Fig. 1: The modeling error is shown as a function of the number of training samples.

TABLE I: Comparison on modeling error and cost.

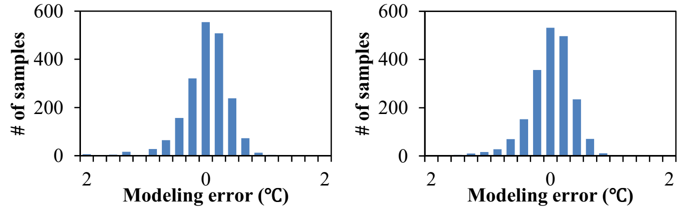
	LS	OMP
# of samples	5500	300
Modeling error	0.41 °C	0.41 °C
Simulation time	42.8 h	2.3 h
Fitting time	11.0 s	0.2 s
Total cost	42.8 h	2.3 h

the unselected factors are set to zero. Algorithm 1 summaries the overall flow of OMP. Once we have solved the model coefficients for all the locations, the temperature model in Eq. (3) is created.

B. Case Study

We use a building example to demonstrate the efficiency of the OMP algorithm for building temperature modeling. A building with 90 locations is created. We assume that there is no temperature variation within a single location. This example has 1106 independent factors. The probability distributions of these factors include Gaussian and uniform distributions. Latin hypercube sampling [18] is used to generate 8000 random samples where 6000 samples are used for model training and the other 2000 samples are used for error estimation. The building is simulated using EnergyPlus [22] on a desktop with a 4-core Intel i7 CPU and 16GB memory. In what follows, we will use the average temperature of a specific location as an example to compare OMP with LS in terms of both the modeling accuracy and computational cost.

Fig. 1 shows how the modeling error varies with the number of training samples. For both LS and OMP, the modeling error decreases as the number of samples increases. However, given the same number of samples, OMP is able to achieve substantially higher accuracy than LS, especially if only few samples are available. Table I further compares the modeling error and cost between LS and OMP. The overall modeling cost includes two parts: (i) simulation time and (ii) fitting time. Table I indicates that the overall modeling cost is dominated by the simulation time. OMP achieves 18.6× speedup over LS, without surrendering any modeling accuracy. Fig. 2 further plots the histogram of the modeling error. It further demonstrates that OMP requires much fewer training samples than LS in order to achieve the same modeling accuracy.



(a) Modeling error of LS with 5500 training samples

(b) Modeling error of OMP with 300 training samples

Fig. 2: Histogram of the modeling error is estimated from the testing samples.

IV. Temperature Measurement by Bayesian Model Fusion

In the previous section, based on the prior distribution of $\mathbf{X}$, we have created a temperature model for each location, i.e., Eq. (3). The temperature model can be used for in-building temperature estimation and uncertainty/sensitivity analysis. In other words, if we know the value of $\mathbf{X}$, we can predict the in-building temperature by Eq. (3). However, for a practical building, the exact value of $\mathbf{X}$ cannot be obtained easily, and, hence, it is difficult to estimate the in-building temperature by Eq. (3) directly.

To address this challenge, we assume that a small number of (say, M, $M < N$) locations are equipped temperature sensors so that we can obtain accurate temperature measurements with a small noise in these locations. Let $(\mathscr{Y})$ and $(\mathscr{N})$ be the index sets for those locations with and without sensors, respectively. For the locations with sensors, we have the following temperature model:

$$\mathbf{A}^{(\mathscr{Y})}\mathbf{X} = \mathbf{T}^{(\mathscr{Y})}, \tag{15}$$

where $\mathbf{T}^{(\mathscr{Y})}$ is obtained from sensor measurements. If $\mathbf{X}$ can be solved from Eq. (15), the temperature of the locations without sensors can be simply predicted by a matrix vector multiplication:

$$\mathbf{T}^{(\mathscr{N})} = \mathbf{A}^{(\mathscr{N})}\mathbf{X}. \tag{16}$$

This method is expected to predict the in-building temperature under various conditions (i.e., different $\mathbf{X}$), because when creating the temperature model Eq. (3), we assume that $\mathbf{X}$ follows some distribution instead of a deterministic value. As a result, the fitted temperature model should be applicable to any case that is in the scope of the prior distribution of $\mathbf{X}$.

Eq. (15) is under-determined in practice, because M is typically much smaller than K. Hence, Eq. (15) cannot be solved by a conventional linear solver. In addition, $\mathbf{X}$ is not sparse so sparse regression cannot be used, either. In what follows, we will introduce a BMF method [9], [23], [24] to solve $\mathbf{X}$ from Eq. (15) by maximum-a-posteriori (MAP) estimation. As such, the temperature of the locations without sensors can be calculated by Eq. (16).

The key idea of BMF is to combine the prior knowledge obtained from temperature modeling with few sensor measurements. It contains two core steps: 1) statistically defining the prior knowledge learned from temperature modeling, and 2) applying MAP estimation to solve $\mathbf{X}$ through Bayesian inference.

A. Prior Knowledge Definition

As assumed in Section II, the prior distribution of each independent factor x_k $(k = 1, 2, \cdots, K)$ is known. The joint distribution of $\mathbf{X}$ is simply the product of all the individual density functions:

$$pdf(\mathbf{X}) = \prod_{k=1}^{K} pdf(x_k). \tag{17}$$

Consider the modeling error in the linear regression model of Eq. (3):

$$\mathbf{AX} = \mathbf{T} + \mathbf{e}, \tag{18}$$

where $\mathbf{e} = [e^{(1)}, e^{(2)}, \cdots, e^{(N)}]^T$ is the modeling error of all the locations. In temperature modeling, for location n $(n = 1, 2, \cdots, N)$, the modeling error vector of all the S samples can be obtained as follows:

$$\mathbf{E}^{(n)} = \mathbf{Y}\boldsymbol{\alpha}^{(n)} - \mathbf{t}^{(n)}, \tag{19}$$

where $\mathbf{E}^{(n)} = [e^{(n)}_{(1)}, e^{(n)}_{(2)}, \cdots, e^{(n)}_{(S)}]^T$. Once the model coefficients are calculated by sparse regression, we can define the prior distribution of the modeling error. The components of $\mathbf{E}^{(n)}$ typically show a zero-mean Gaussian distribution. Hence, we use a zero-mean Gaussian distribution to model each $e^{(n)}$. The standard deviation of $e^{(n)}$ which is denoted as σ_n equals the numerically calculated standard deviation of $\mathbf{E}^{(n)}$. The correlation coefficient between $e^{(i)}$ and $e^{(j)}$ which is denoted as $\rho_{i,j}$ equals to the numerically calculated correlation coefficient between $\mathbf{E}^{(i)}$ and $\mathbf{E}^{(j)}$.

B. Maximum-A-Posteriori Estimation

The goal of MAP is to solve $\mathbf{X}$ from Eq. (15) by maximizing the posterior distribution $pdf\left(\mathbf{X}|\mathbf{T}^{(\mathscr{Y})}\right)$. Based on the Bayes' theorem, $pdf\left(\mathbf{X}|\mathbf{T}^{(\mathscr{Y})}\right)$ is proportional to the product of the prior distribution $pdf(\mathbf{X})$ and the likelihood function $pdf\left(\mathbf{T}^{(\mathscr{Y})}|\mathbf{X}\right)$:

$$pdf\left(\mathbf{X}|\mathbf{T}^{(\mathscr{Y})}\right) \propto pdf\left(\mathbf{T}^{(\mathscr{Y})}|\mathbf{X}\right) pdf(\mathbf{X}), \tag{20}$$

where the prior distribution $pdf(\mathbf{X})$ is given in Eq. (17). Based on Eq. (18) and the prior distribution of the modeling error defined in Section IV-A, the likelihood function $pdf\left(\mathbf{T}^{(\mathscr{Y})}|\mathbf{X}\right)$ can be expressed as a multivariate Gaussian distribution which is related to the prior distribution of the modeling error:

$$\begin{aligned} pdf\left(\mathbf{T}^{(\mathscr{Y})}|\mathbf{X}\right) = & \frac{1}{\left(\sqrt{2\pi}\right)^M \sqrt{|\mathbf{\Sigma}|}} \cdot \\ \exp & \left[-\frac{1}{2}\left(\mathbf{AX} - \mathbf{T}^{(\mathscr{Y})}\right)^T \mathbf{\Sigma}^{-1} \left(\mathbf{AX} - \mathbf{T}^{(\mathscr{Y})}\right)\right], \end{aligned} \tag{21}$$

where $\mathbf{\Sigma}$ is the covariance matrix of the modeling error for the locations with sensors, i.e., $\Sigma_{i,j} = \rho_{pi,pj}\sigma_{pi}\sigma_{pj}$, where pi and pj are the ith and jth elements in the index set $(\mathscr{Y})$.

The posterior distribution can be calculated by combining Eq. (17), Eq. (20) and Eq. (21):

$$\begin{aligned} pdf\left(\mathbf{X}|\mathbf{T}^{(\mathscr{Y})}\right) \propto & \left[\prod_{k=1}^{K} pdf(x_k)\right] \cdot \\ \exp & \left[-\frac{1}{2}\left(\mathbf{AX} - \mathbf{T}^{(\mathscr{Y})}\right)^T \mathbf{\Sigma}^{-1} \left(\mathbf{AX} - \mathbf{T}^{(\mathscr{Y})}\right)\right]. \end{aligned} \tag{22}$$

$\mathbf{X}$ can be solved from Eq. (22) by MAP estimation such that $pdf\left(\mathbf{X}|\mathbf{T}^{(\mathscr{Y})}\right)$ is maximized. As it is not necessarily to assume that each component of $\mathbf{X}$ is Gaussian, it may be impossible to derive an analytical solution to maximize the posterior distribution. Instead, the MAP estimation can be numerically solved.

C. Sensor Placement

The accuracy of the BMF algorithm strongly depends on the measured temperature values $\mathbf{T}^{(\mathscr{Y})}$, which in turn depends on the locations of the M temperature sensors. Consequently, a good sensor placement is required for the BMF framework. A simple idea is to evenly distribute the M sensors over the N locations so that the measured temperature values are representative. To achieve this goal, we introduce a modified Latin hypercube sampling (M-LHS) method [25] to generate well-controlled locations of temperature sensors. The method is quite simple. If there are rooms in the building model, we first evenly distribute the M sensors over these rooms, and then for each room, we evenly distribute the sensors over all the locations in this room. If there is no rooms in the building model, we only need to evenly distribute the M sensors over all the N locations.

D. Case Study

We use a building example with 144 locations to demonstrate the efficiency of the BMF method. This building has 264 independent factors where 94 and 170 factors follow Gaussian and uniform distributions, respectively. We use 1000 samples for training the temperature model and 50 samples for error estimation. For the 50 testing samples, the simulated results from EnergyPlus are treated as the "actual" temperature. We assume that sensor measurements have noise and the noise is modeled by a zero-mean Gaussian distribution with standard deviation of 0.3 degree. In other words, the measured temperature of a location equals the simulated temperature plus a randomly generated Gaussian noise.

Fig. 3 shows how the estimation error for locations without sensors varies with the number of sensors. The estimation error of BMF decreases as the number of sensors increases, and saturates at about 0.12 degree when there are a sufficient number of sensors. Given the same number of sensors, M-LHS is able to achieve higher accuracy than that of random placement, especially when only few sensors are available. For instance, when using 12 sensors, the estimation error of M-LHS is 0.1 degree smaller than that of random placement. To achieve the same accuracy of M-LHS using 12 sensors, random placement requires 23 or 24 sensors. In practice, the

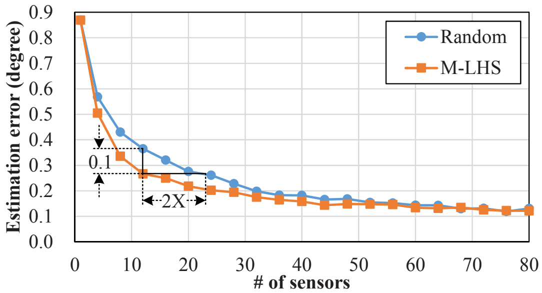
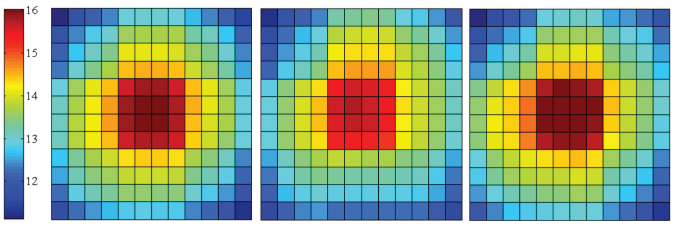


Fig. 3: The estimation error for locations without sensors is shown as a function of the number of sensors.



(a) Actual temperature (b) Estimation by random placement (c) Estimation by M-LHS

Fig. 4: The estimated temperature profiles are compared among "actual" temperature, random placement, and M-LHS. 16 sensors are used in this example.

BMF method with M-LHS placement can obtain a reasonable estimation accuracy by only using 10 to 20 sensors.

Fig. 4 compares the temperature profiles obtained by random placement and M-LHS with the "actual" temperature profile for one example. It provides an intuitive view of the temperature distribution which further shows that M-LHS obtains more accurate estimation than random placement.

V. Conclusion

Temperature modeling and estimation in smart buildings is an important task for offering comfortable services to the occupants and reducing energy consumption of buildings. In this paper, we have given an overview of state-of-the-art cyber-physical temperature estimation methods, from the modeling aspect to the measurement aspect. Specifically, we have introduced two algorithms, OMP and BMF, for temperature modeling and estimation, respectively. Through the two advanced methods, we are able to achieve high-accuracy and low-cost temperature modeling and measurement in smart buildings.

References

[1] G. Augenbroe, "Trends in building simulation," *Building and Environment*, vol. 37, no. 8-9, pp. 891–902, 2002.

[2] A. Saltelli, S. Tarantola, and F. Campolongo, "Sensitivity analysis as an ingredient of modeling," *Statistical Science*, vol. 15, no. 4, pp. 377–395, 2000.

[3] Energy efficiency trends in residential and commercial buildings. [Online]. Available: http://apps1.eere.energy.gov/buildings/publications/pdfs/corporate/building_trends_2010.pdf

[4] E. Lee, "CPS foundations," in *Design Automation Conference (DAC)*, June 2010, pp. 737–742.

[5] S. Kelly, N. Suryadevara, and S. Mukhopadhyay, "Towards the Implementation of IoT for Environmental Condition Monitoring in Homes," *Sensors Journal, IEEE*, vol. 13, no. 10, pp. 3846–3853, Oct 2013.

[6] M. Lazarescu, "Design of a WSN Platform for Long-Term Environmental Monitoring for IoT Applications," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, vol. 3, no. 1, pp. 45–54, March 2013.

[7] Y. Wang, A. Yang, Z. Li, P. Wang, and H. Yang, "Blind drift calibration of sensor networks using signal space projection and Kalman filter," in *International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, April 2015, pp. 1–6.

[8] X. Chen, X. Li, and S. X.-D. Tan, "From Robust Chip to Smart Building: CAD Algorithms and Methodologies for Uncertainty Analysis of Building Performance," in *International Conference on Computer-Aided Design (ICCAD)*, 2015, pp. 457–464.

[9] X. Chen and X. Li, "Virtual Temperature Measurement for Smart Buildings via Bayesian Model Fusion," in *International Symposium on Circuits and Systems (ISCAS)*, 2016.

[10] C. J. Hopfe and J. L. Hensen, "Uncertainty analysis in building performance simulation for design support," *Energy and Buildings*, vol. 43, no. 10, pp. 2798–2805, 2011.

[11] A. S. Silva and E. Ghisi, "Uncertainty analysis of user behaviour and physical parameters in residential building performance simulation," *Energy and Buildings*, vol. 76, no. 0, pp. 381–391, 2014.

[12] D. G. Sanchez, B. Lacarrire, M. Musy, and B. Bourges, "Application of sensitivity analysis in building energy simulations: Combining first- and second-order elementary effects methods," *Energy and Buildings*, vol. 68, Part C, no. 0, pp. 741–750, 2014.

[13] J. S. Hygh, J. F. DeCarolis, D. B. Hill, and S. R. Ranjithan, "Multivariate regression as an energy assessment tool in early building design," *Building and Environment*, vol. 57, no. 0, pp. 165–175, 2012.

[14] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, *The elements of statistical learning: data mining, inference and prediction*, 2nd ed. Springer, 2005.

[15] A. Charnes, E. Frome, and P.-L. Yu, "The equivalence of generalized least squares and maximum likelihood estimates in the exponential family," *Journal of the American Statistical Association*, vol. 71, no. 353, pp. 169–171, 1976.

[16] W. Tian, "A review of sensitivity analysis methods in building energy analysis," *Renewable and Sustainable Energy Reviews*, vol. 20, no. 0, pp. 411–419, 2013.

[17] A. Atkinson, A. Donev, and R. Tobias, *Optimum experimental designs, with SAS*, ser. Oxford Statistical Science Series. OUP Oxford, 2007.

[18] M. D. McKay, "Latin hypercube sampling as a tool in uncertainty analysis of computer models," in *Conference on Winter Simulation*, 1992, pp. 557–564.

[19] I. Rish and G. Grabarnik, *Sparse modeling: theory, algorithms, and applications*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 2014.

[20] X. Li, "Finding deterministic solution from underdetermined equation: large-scale performance variability modeling of analog/RF circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 29, no. 11, pp. 1661–1668, Nov 2010.

[21] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655–4666, Dec 2007.

[22] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. Buhl, Y. Huang, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, M. J. Witte, and J. Glazer, "EnergyPlus: creating a new-generation building energy simulation program," *Energy and Buildings*, vol. 33, no. 4, pp. 319–331, 2001.

[23] X. Li, F. Wang, S. Sun, and C. Gu, "Bayesian Model Fusion: A statistical framework for efficient pre-silicon validation and post-silicon tuning of complex analog and mixed-signal circuits," in *International Conference on Computer-Aided Design (ICCAD)*, Nov 2013, pp. 795–802.

[24] F. Wang, W. Zhang, S. Sun, X. Li, and C. Gu, "Bayesian Model Fusion: Large-scale performance modeling of analog and mixed-signal circuits by reusing early-stage data," in *Design Automation Conference*, May 2013, pp. 1–6.

[25] W. Zhang, X. Li, F. Liu, E. Acar, R. Rutenbar, and R. Blanton, "Virtual Probe: A Statistical Framework for Low-Cost Silicon Characterization of Nanoscale Integrated Circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 30, no. 12, pp. 1814–1827, Dec 2011.