

Measurement and verification of building systems under uncertain data: A Gaussian process modeling approach



Michael C. Burkhardt^a, Yeonsook Heo^{b,*}, Victor M. Zavala^a

^a Mathematics and Computer Science Division, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439, USA

^b Department of Architecture, University of Cambridge, 1-5 Scroope Terrace, Cambridge CB2 1PX, UK

ARTICLE INFO

Article history:

Received 15 October 2013

Received in revised form 10 January 2014

Accepted 30 January 2014

Keywords:

Gaussian process modeling

Data uncertainty

Expectation maximization

Measurement and verification

ABSTRACT

Uncertainty in sensor data (e.g., weather, occupancy) complicates the construction of baseline models for measurement and verification (M&V). We present a Monte Carlo expectation maximization (MCEM) framework for constructing baseline Gaussian process (GP) models under uncertain input data. We demonstrate that the GP-MCEM framework yields more robust predictions and confidence levels compared with standard GP training approaches that neglect uncertainty. We argue that the approach can also reduce data needs because it implicitly expands the data range used for training and can thus be used as a mechanism to reduce data collection and sensor installation costs in M&V processes. We analyze the numerical behavior of the framework and conclude that robust predictions can be obtained with relatively few samples.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Gaussian process (GP) modeling is a powerful statistical modeling framework that proposes a structure for the covariance matrix of input variables to compute predictions of output variables [1]. A Bayesian framework is used to train hyperparameters of the input covariance matrix and to derive a predictive distribution for output variables at test input points. As a result, GP models can capture complex nonlinear relationships between multiple input and output variables and can provide mean predictions and associated uncertainty levels. These features make GP particularly attractive for measurement and verification (M&V), as they can help assess the amount and resolution of data required to reach desired uncertainty levels. Using real and simulated studies, Heo et al. [2,3] have demonstrated that GP models can systematically capture the effect of multiple weather variables (e.g., ambient temperature, relative humidity) as well as occupancy variables on energy demands and can effectively quantify uncertainty levels.

The standard GP modeling framework [1] has been applied in a number of building applications. For M&V, GP models have been developed to predict baseline energy use during the post-retrofit period [2,4]. In fault diagnosis and detection, GP models have been used to predict system performance baselines on the basis of measured operational data [5]. In addition, GP models have been used as surrogates of complex energy simulation models to reduce computational complexity of tasks such as model calibration [6], model-predictive control [7], and optimal design [8].

The standard GP framework assumes that input data are known with full certainty, but such is not the case in many situations. In particular, weather conditions and building occupancy (key variables suggested in M&V protocols [9,10]) are often obtained from noisy sensors or they might be too expensive to measure on site. For instance, government-published weather data are measured at distant meteorological stations and hence may not capture microclimate conditions around the specific building site. This situation can lead to significant discrepancy in predicted energy use [11,12]. In addition, building occupancy is difficult to measure in real time and with high accuracy [13]. Occupant densities in office buildings can vary between 4.3 m² and 22.8 m² per person, and this range significantly impacts internal heat gains [14,15]. Ultimately, variations in input data can distort energy predictions and can

* Corresponding author. Tel.: +44 1223760114.
E-mail address: yh305@cam.ac.uk (Y. Heo).

lead to inconsistent confidence levels, making M&V conclusions unreliable.

This work extends the standard GP framework reported in the literature by explicitly accounting for uncertain input data in the learning of the hyperparameters (i.e., model training). We use the framework of Quiñonero-Candela [16] in which the hyperparameters are learned by maximizing the likelihood function with marginalized input data (also known as the marginal evidence). The maximization of the marginal evidence is performed by using a Monte Carlo expectation maximization (MCEM) algorithm. We call the resulting framework GP-MCEM.

Using a simulation setting to emulate the energy performance of an advanced multivariable control system in an office building, we demonstrate that more consistent uncertainty estimates of energy demands can be obtained by using the GP-MCEM framework compared with standard GP approaches that neglect input uncertainty. In addition, we demonstrate that much more robust estimates of hyperparameters can be obtained. In particular, we argue that even an ideal GP approach (learning hyperparameters by using perfect input data) can suffer from prediction robustness when confronted with test points slightly outside the training set and, consequently, large amounts of data can be needed to mitigate this lack of robustness. The proposed GP-MCEM approach, on the other hand, implicitly expands the input data range used for training and can thus obtain more robust hyperparameters that perform well outside the training set and can thus reduce training data needs. We perform computational experiments with the GP-MCEM framework to validate performance with varying numbers of Monte Carlo samples.

The paper is structured as follows. In Section 2 we derive the GP-MCEM framework. In Section 3 we discuss the algorithmic implementation and convergence properties. In Section 4 we present a detailed numerical study to demonstrate the advantages of the proposed framework. The paper closes in Section 5 with conclusions and directions of future work.

2. GP-MCEM framework

In this section, we derive the framework to train GP models under uncertain input data.

2.1. Setting

Consider a training set with n output data points (e.g., daily energy demands) $\{y^i\}$, where $y^i \in \mathbb{R}$, $i = 1, \dots, n$. Consider also a set of uncertain input data points (e.g., occupancy level, ambient temperature, relative humidity) $\{\mathbf{x}^i\}$ where $\mathbf{x}^i \in \mathbb{R}^d$, $i = 1, \dots, n$. Each element of vector \mathbf{x}^i is given by x_j^i , $j = 1, \dots, d$ and d is the number of input variables. We also define the output vector $\mathbf{y} := \{y^i\} \in \mathbb{R}^n$ and the input matrix $X = \{\mathbf{x}^i\} \in \mathbb{R}^{n \times d}$, where the i th input vector \mathbf{x}^i forms the i th row of the matrix X .

Each input data observation \mathbf{x}^i is assumed to be random and independently normally distributed as $\mathbf{x}^i | \phi^i \sim \mathcal{N}(\mathbf{m}^i, \mathbf{S})$, $i = 1, \dots, n$. Here, $\phi^i = \{\mathbf{m}^i, \mathbf{S}\}$ are the statistics of observation \mathbf{x}^i where \mathbf{m}^i is the mean of each observation \mathbf{x}^i , $i = 1, \dots, n$. We assume that $\mathbf{S} := \text{diag}(s_1^2, \dots, s_d^2) \in \mathbb{R}^{d \times d}$ where $s_j^2 \in \mathbb{R}$ is the variance of variable \mathbf{x}_j^i , $j = 1, \dots, d$. Consequently, the variance for each j th variable is constant across the observations $i = 1, \dots, n$. In practical terms, this amounts to assuming a different distribution for each input variable $j = 1, \dots, d$, but the variance of each variable does not depend on the observation $i = 1, \dots, n$. In addition, it assumes that input variables are uncorrelated. From a conceptual standpoint, these assumptions are not necessary for

the proposed framework but they greatly simplify implementation of sampling procedures, which we will discuss later. The statistics (i.e., mean and variance) for X are grouped as $\Phi = \{\phi^i\}$; and, because of the assumed structure, we have that the conditional prior of X (conditional distribution of X given the statistics Φ) is given by

$$p(X|\Phi) = \prod_{i=1}^n p(\mathbf{x}^i|\phi^i), \quad (2.1)$$

where

$$p(\mathbf{x}^i|\phi^i) = (2\pi)^{-d/2} |\mathbf{S}|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{x}^i - \mathbf{m}^i)^T \mathbf{S}^{-1}(\mathbf{x}^i - \mathbf{m}^i)\right), \quad i = 1, \dots, n. \quad (2.2)$$

Here, the matrix determinant is denoted as $|\cdot|$. The conditional distribution of \mathbf{y} given the input data X and hyperparameters $\theta \in \mathbb{R}^p$ is given by

$$p(\mathbf{y}|X, \theta) = (2\pi)^{-n/2} |\mathbf{K}(\theta, X, X)|^{-1/2} \exp\left(-\frac{1}{2}\mathbf{y}^T \mathbf{K}(\theta, X, X)^{-1}\mathbf{y}\right). \quad (2.3)$$

Here, $\mathbf{K}(\theta, X, X) = \{k(\theta, \mathbf{x}^i, \mathbf{x}^j)\} \in \mathbb{R}^{n \times n}$ is the input matrix covariance, $\theta \in \mathbb{R}^p$ are the hyperparameters, and $k: \mathbb{R}^p \times \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ is the covariance function. While the methods developed here are applicable to any proper covariance function, we focus on the standard Gaussian covariance used in most of the literature [2,4–6,8],

$$k(\theta, \mathbf{x}^i, \mathbf{x}^j) = \theta_0 \exp\left(-\frac{\|\mathbf{x}^i - \mathbf{x}^j\|^2}{\theta_1}\right) + \theta_2 \delta_{ij}. \quad (2.4)$$

For alternative covariance functions see [1]. Here, we define the hyperparameter vector $\theta := [\theta_0, \theta_1, \theta_2]$, the Kronecker delta δ_{ij} , and the Euclidean norm $\|\cdot\|$.

2.2. Marginal evidence derivation

Quiñonero-Candela [16] proposes a couple of approaches to train a GP model under uncertain inputs. The first approach maximizes the joint posterior $p(X, \theta | \mathbf{y}, \Phi) \propto p(\mathbf{y}|X, \theta)p(X|\Phi)$ with respect to θ and X . This can in principle be done because $p(\mathbf{y}|X, \theta)p(X|\Phi)$ has an explicit formulation, given by the product of Eqs. (2.1) and (2.3). Consequently, the log of the joint posterior is separable. However, this approach proved to be unreliable because of the introduction of multiple local minima. The second approach, used in this work, consists of computing a maximum a posteriori estimate (MAP). This is achieved by maximizing the posterior $p(\theta | \mathbf{y}, \Phi) \propto p(\mathbf{y}|\theta, \Phi)p(\theta)$. Assuming a flat prior $p(\theta)$, this is equivalent to maximizing $p(\mathbf{y}|\theta, \Phi)$. In addition, we know that

$$\begin{aligned} p(\mathbf{y}|\theta, \Phi) &= \int p(\mathbf{y}, X|\theta, \Phi) dX \\ &= \int p(\mathbf{y}|\theta, X)p(X|\Phi) dX, \end{aligned} \quad (2.5)$$

where $\int p(\mathbf{y}|\theta, X)p(X|\Phi)dX$ is the so-called *marginal evidence*. Consequently, we can obtain the MAP estimate of the hyperparameters θ by maximizing the log marginal evidence,

$$\mathcal{L}(\theta) := \log \left(\int p(\mathbf{y}|\theta, X)p(X|\Phi)dX \right) = \log \left(\int p(\mathbf{y}, X|\theta, \Phi)dX \right). \tag{2.6}$$

From a practical standpoint, we note that the *marginal evidence is the likelihood function $p(\mathbf{y}|X, \theta)$ averaged over the input data*. The likelihood function is typically maximized in standard parameter estimation procedures in which input data observations are assumed to be given (certain). This observation is important because it highlights that maximizing the marginal evidence has the implicit effect of capturing the entire range of input data and not only data points as in standard approaches. As we will demonstrate later, this significantly aids prediction robustness and can be helpful in reducing data needs.

2.3. EM algorithm

The log marginal evidence shown in Eq. (2.6) has a complicated form (the expected value is inside the logarithm) and thus direct maximization is not straightforward. Such maximization, however, can be done indirectly by using the expectation-maximization algorithm, which we now proceed to explain. First note that we have the following lower bound for the log marginal evidence:

$$\begin{aligned} \log \left(\int p(\mathbf{y}, X|\theta, \Phi)dX \right) &= \log \left(\int q(X) \frac{p(\mathbf{y}, X|\theta, \Phi)}{q(X)} dX \right) \\ &\geq \int q(X) \log \left(\frac{p(\mathbf{y}, X|\theta, \Phi)}{q(X)} \right) dX \tag{2.7} \\ &:= \mathcal{F}(q(\cdot), \theta). \end{aligned}$$

Here, $q(X)$ represents an arbitrary distribution for X , and Jensen's inequality establishes the lower bound. The EM algorithm maximizes the lower bound $\mathcal{F}(q(\cdot), \theta)$ with respect to the density function $q(\cdot)$ and the hyperparameters θ . This can be done using a coordinate ascent technique. In the E-step we maximize $\mathcal{F}(q(\cdot), \theta)$ with respect to $q(\cdot)$, leaving θ fixed to a current guess, and in the M-step we maximize $\mathcal{F}(q(\cdot), \theta)$ with respect to θ , leaving $q(\cdot)$ fixed to the updated value.

For candidate distribution $q(\cdot)$, the EM algorithm uses the posterior distribution $p(X|\mathbf{y}, \theta, \Phi)$. This is motivated by the relationship,

$$\mathcal{L} = \mathcal{F} + \mathcal{D}(q(X), p(X|\mathbf{y}, \theta, \Phi)). \tag{2.8}$$

Here, $\mathcal{D}(q(X), p(X|\mathbf{y}, \theta, \Phi))$ is the Kullback–Leibler divergence that measures the distance between distributions $q(X)$ and $p(X|\mathbf{y}, \theta, \Phi)$. Note that because the divergence is a positive quantity, we have $\mathcal{L} \geq \mathcal{F}$. This relationship thus suggests that, with θ fixed, \mathcal{L} is maximized by setting the distribution $q(\cdot)$ equal to the posterior $p(X|\mathbf{y}, \theta_k, \Phi)$, where θ_k is a current guess. We thus define the current distribution guess as $q_k(\cdot) = p(X|\mathbf{y}, \theta_k, \Phi)$. This gives the E-step, and $\mathcal{F}(q_k(\cdot), \theta)$ can be written as

$$\mathcal{F}(q_k(\cdot), \theta) = \int p(X|\mathbf{y}, \theta_k, \Phi) \log \left(\frac{p(\mathbf{y}, X|\theta, \Phi)}{p(X|\mathbf{y}, \theta_k, \Phi)} \right) dX. \tag{2.9}$$

In the M-step, we now seek to maximize the lower bound $\mathcal{F}(q_k(\cdot), \theta)$ with respect to θ . We note that $p(\mathbf{y}, X|\theta, \Phi) \propto p(\mathbf{y}|X, \theta)p(X|\Phi)$. Consequently, Eq. (2.9) becomes

$$\begin{aligned} \mathcal{F}(q_k(\cdot), \theta) &= \int p(X|\mathbf{y}, \theta_k, \Phi) \log \left(\frac{p(\mathbf{y}|X, \theta)p(X|\Phi)}{p(X|\mathbf{y}, \theta_k, \Phi)} \right) dX \\ &= \int p(X|\mathbf{y}, \theta_k, \Phi) \log \left(\frac{p(\mathbf{y}|X, \theta)p(X|\Phi)}{p(\mathbf{y}|X, \theta_k)p(X|\Phi)} \right) dX \\ &= \int p(X|\mathbf{y}, \theta_k, \Phi) \log \left(\frac{p(\mathbf{y}|X, \theta)}{p(\mathbf{y}|X, \theta_k)} \right) dX \\ &= \int p(X|\mathbf{y}, \theta_k, \Phi) \log p(\mathbf{y}|X, \theta) dX - \int p(X|\mathbf{y}, \theta_k, \Phi) \log p(\mathbf{y}|X, \theta_k) dX \\ &:= \mathcal{Q}(\theta, \theta_k) + C. \end{aligned} \tag{2.10}$$

Here, the second equality follows from Bayes theorem which states that

$$p(X|\mathbf{y}, \theta_k, \Phi) \propto p(\mathbf{y}|X, \theta_k) p(X|\Phi). \tag{2.11}$$

We also have that

$$\mathcal{Q}(\theta, \theta_k) := \int p(X|\mathbf{y}, \theta_k, \Phi) \log p(\mathbf{y}|X, \theta) dX \tag{2.12a}$$

$$C := - \int p(X|\mathbf{y}, \theta_k, \Phi) \log p(\mathbf{y}|X, \theta_k) dX. \tag{2.12b}$$

Since C is a constant independent of θ , it is irrelevant in the maximization. The next guess for θ is thus obtained from the maximization or M-step,

$$\theta_{k+1} \leftarrow \underset{\xi}{\operatorname{argmax}} \mathcal{Q}(\xi, \theta_k). \tag{2.13}$$

After this, the conditional distribution $p(X|\mathbf{y}, \theta_k, \Phi)$ is updated in the E-step and the M-step (2.13) is repeated. This recursion corresponds to the exact EM algorithm.

3. MCEM algorithm

The integral in Eq. (2.12a) is analytically intractable but can be approximated by Monte Carlo sampling, which yields the MCEM algorithm. We obtain N data samples $X_k(\omega)$, $\omega = 1, \dots, N$ from $p(X|\mathbf{y}, \theta_k, \Phi)$ or, equivalently, from $p(\mathbf{y}|X, \theta_k) p(X|\Phi)$ to obtain an empirical approximation of the integral. We then compute an approximate M-step of the form

$$\theta_{k+1} \leftarrow \underset{\xi}{\operatorname{argmax}} \mathcal{Q}^N(\xi, \theta_k), \tag{3.14}$$

where

$$\mathcal{Q}^N(\xi, \theta_k) = \frac{1}{N} \sum_{\omega=1}^N \log p(\mathbf{y}|X_k(\omega), \xi). \tag{3.15}$$

We now summarize the MCEM algorithm. At iteration k , the E-step samples the estimated posterior distribution $p(X|\mathbf{y}, \theta_k, \Phi) \propto p(\mathbf{y}|X, \theta_k) p(X|\Phi)$, where the structure of the distributions is given by Eqs. (2.1) and (2.3). We can thus sample from $p(X|\mathbf{y}, \theta_k, \Phi)$ using a Markov chain Monte Carlo (MCMC) algorithm because all that is needed is a routine that evaluates the target distribution $p(\mathbf{y}|X, \theta_k) p(X|\Phi)$. We define a method of the form $\{X_k(\omega)\} \leftarrow \text{MCMC}(p(\mathbf{y}|X, \theta_k) p(X|\Phi), N)$. Using these samples, we then obtain an updated value for the hyperparameters by solving Eq. (3.14). The algorithm is presented below.

input : Data \mathbf{y} , statistics $\Phi = \{\mathbf{m}^i, \mathbf{S}\}$, number of MCMC samples N , tolerance $\tau > 0$, and initial guess θ .

output: Estimates for hyperparameters θ that maximize \mathcal{L} .

Set $\epsilon \leftarrow 2 \cdot \tau$

while $\epsilon \geq \tau$ **do**

$\{X(\omega)\} \leftarrow \text{MCMC}(p(\mathbf{y} | X, \theta) p(X | \Phi), N)$

$\theta' \leftarrow \theta$

$\theta \leftarrow \text{argmax}_{\xi} \mathcal{Q}^N(\xi, \theta)$

$\epsilon \leftarrow \|\theta - \theta'\|_1$

end

We highlight that the input posterior $p(X|\mathbf{y}, \theta_k, \Phi)$ does not have a practical form amenable for sampling. Therefore, the relationship (2.11) is important. This relationship indicates that the input posterior is proportional to the product of the output posterior $p(\mathbf{y}|X, \theta_k)$ and the input distribution $p(X|\Phi)$. This enables the use of MCMC methods such as slice sampling, which requires knowledge only of the product function $p(\mathbf{y}|X, \theta_k)p(X|\Phi)$ or the log form $\log(p(\mathbf{y}|X, \theta_k) + \log(p(X|\Phi)))$ which, in our case has a known form.

3.1. Convergence

The exact EM algorithm has a guaranteed improvement in the log marginal evidence \mathcal{L} at each iteration. This results from the fact that, in the E-step at iteration k , the lower bound is maximized with respect to θ and, consequently, the log marginal evidence increases, having $q(\cdot)$ fixed at $p(X|\mathbf{y}, \theta_k, \Phi)$. Once the hyperparameters are updated in the M-step to θ_{k+1} , the Kullback–Leibler divergence becomes nonzero, because the distribution in the previous iteration was $q_k(\cdot) = p(X|\mathbf{y}, \theta_k, \Phi)$ and not $p(X|\mathbf{y}, \theta_{k+1}, \Phi)$, which maximizes the log marginal evidence with respect to $q(\cdot)$. Consequently, the increase in the log marginal evidence will be larger than the increase in the lower bound.

The transition to MCEM sacrifices the guaranteed increase in the log marginal evidence because the M-step cannot be performed exactly. Numerous papers, however, establish theoretical convergence results for MCEM algorithms. Fort and Moulines establish convergence but require that the number of MCMC samples at each iteration increases without bound [19]. Booth and Hobert assert that, even as the sample size increases, the Monte Carlo error remains in the estimate [20]. Delyon et al. proposed a stochastic approximation EM algorithm that replaces the MC update with a weighted average of the current and previous $\mathcal{Q}(\cdot)$ functions [21]; in this way they obtain convergence without increasing the number of samples. In the following section, we explore the empirical behavior of the MCEM method as the number of samples increases.

We highlight that, even in the exact EM algorithm, one cannot monitor stationarity of \mathcal{L} because of its complicated form. Therefore, we stop the algorithm once the change in the hyperparameters is below a tolerance.

3.2. Computational details

The explicit form of function $\mathcal{Q}^N(\theta, \theta_0)$ is given by

$$\begin{aligned} \mathcal{Q}^N(\theta, \theta_k) &= \frac{1}{N} \sum_{\omega=1}^N \log p(\mathbf{y}|X_k(\omega), \theta) \\ &= \frac{1}{N} \sum_{\omega=1}^N \left(-\frac{n}{2} \log(2\pi) - \frac{1}{2} \log |\mathbf{K}(\theta, X_k(\omega), X_k(\omega))| - \frac{1}{2} \mathbf{y}^\top \mathbf{K}(\theta, X_k(\omega), X_k(\omega))^{-1} \mathbf{y} \right). \end{aligned} \quad (3.16)$$

Each term in the summation is the log likelihood function maximized in standard GP [1]. The first and second derivatives of this function are well known [22]. In this work, we use the trust-region Newton method implemented in Matlab [23]. The gradient is provided in explicit form and second derivatives are estimated by using finite differences. We use the slice sample procedure implemented in Matlab to compute the MCMC samples.

Once hyperparameters have been trained, it is desired to make predictions at a set of test data points. For prediction at a test point X_t , we propose to use a predictive distribution with mean and covariance,

$$\bar{\mu}(\theta, X, X_t) := \mathbf{K}(\theta, X, X_t) \mathbf{K}(\theta, X, X)^{-1} \mathbf{y}, \quad (3.17a)$$

$$\bar{\mathbf{K}}(\theta, X, X_t) := \mathbf{K}(\theta, X_t, X_t) - \mathbf{K}(\theta, X, X_t) \mathbf{K}(\theta, X, X)^{-1} \mathbf{K}(\theta, X, X_t). \quad (3.17b)$$

Here, X and X_t contain only the mean of the uncertain inputs (i.e., we assume zero variance in the input data). We emphasize that this *predictive* distribution does not consider the effect of input data uncertainty. In other words, our approach considers input uncertainty *only in the training phase*. As we demonstrate in the next section, this already gives significant improvements in prediction performance because it robustifies the estimates of the hyperparameters. Predictive distributions that consider input data uncertainty are much more elaborate and are presented in [16,24]. We leave this as a topic of future research.

4. Case study

In this section, we present a case study to demonstrate that the GP-MCEM approach yields much more robust predictions compared to standard GP approaches. In addition, we discuss numerical performance.

4.1. Prediction performance

We consider a simulated study in which an advanced model predictive control (MPC) system is used to optimize the operation of the HVAC system of an office building. Here, we seek to baseline the energy performance of the MPC system. The office building comprises a well-mixed single space conditioned by an air-handling unit (AHU). Dynamic models of temperature, species concentration (moisture and carbon dioxide), and pressure are

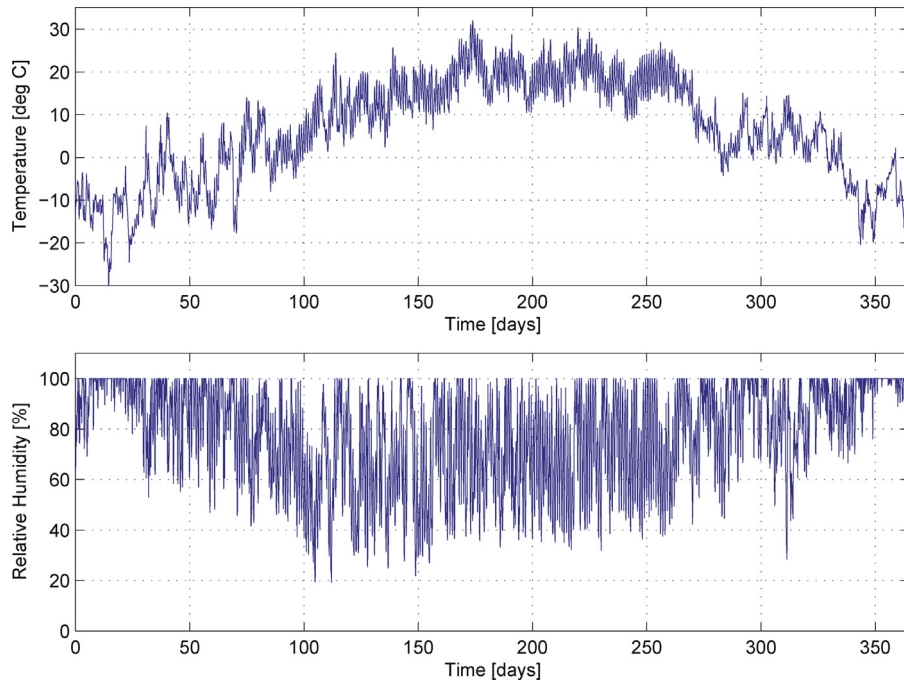


Fig. 1. Hourly ambient temperature and relative humidity profiles for Chicago area, 2006.

used to simulate the comfort and air quality conditions of the building space. The MPC system uses all the available degrees of freedom of the HVAC system (heating and cooling for AHU, moisture removal, ambient, recycle and exhaust flows) to minimize energy while satisfying comfort and air quality constraints. This optimization is done adaptively as external variables (occupancy and weather conditions) change in time. A detailed formulation of the MPC controller and a description of the building model are presented in [2,25,26]. The MPC controller introduces a high degree of coupling between the building control variables, external variables, and the total energy demand; and it induces nonlinear behavior. This makes the construction of baseline models complicated because energy performance becomes a strong function of multiple input variables such as occupancy and ambient conditions.

The mathematical formulation of the MPC controller has the form

$$\min_{\mathbf{u}(\cdot)} \int_t^{t+T} \chi(\tau) d\tau \quad (4.18a)$$

$$\text{s.t. } \frac{d\mathbf{x}}{dt}(\tau) = f_x(\mathbf{x}(\tau), \mathbf{z}(\tau), \mathbf{u}(\tau), \boldsymbol{\omega}(\tau)) \quad (4.18b)$$

$$0 = f_z(\mathbf{x}(\tau), \mathbf{z}(\tau), \mathbf{u}(\tau), \boldsymbol{\omega}(\tau)) \quad (4.18c)$$

$$0 \leq g(\mathbf{x}(\tau), \mathbf{z}(\tau), \mathbf{u}(\tau), \boldsymbol{\omega}(\tau)) \quad (4.18d)$$

$$\chi(\tau) = \Xi(\mathbf{x}(\tau), \mathbf{z}(\tau), \mathbf{u}(\tau)) \quad (4.18e)$$

$$\mathbf{x}(t) = \text{given}, \quad \tau \in [t, t+T]. \quad (4.18f)$$

Here, t is the current time, T is the prediction horizon, $\mathbf{x}(\cdot)$ are the dynamic states of the building (zone temperature, CO_2 and H_2O concentrations, building air mass), $\mathbf{z}(\cdot)$ are the algebraic states (relative humidity, supply air temperature), $\mathbf{u}(\cdot)$ are the degrees of freedom or controls (AHU electrical energy, ambient air flow, recycle flow), and $\boldsymbol{\omega}(\tau)$ are external variables such as occupancy heat and $\text{CO}_2/\text{H}_2\text{O}$ loads and ambient conditions (CO_2 , temperature, and relative humidity). The variable $\chi(\tau)$ represents the total HVAC electrical energy in the HVAC system.

The performance of the MPC controller (4.1) is simulated by running the system using a receding horizon of 24 h with time steps of 1 h for an entire year. We used real weather data in the Chicago area to perform the year-long simulations (see Fig. 1). The total daily HVAC energy is used as the output, and we consider two certain input variables (average daily relative humidity and average ambient temperature) and an uncertain input variable (average daily occupancy). For the certain inputs we consider the values presented in Fig. 1, and we assume zero variance. For the uncertain input we consider two cases. The first case assumes that the average daily occupancy varies *mildly* as $\mathcal{N}(500, 50^2)$. The second case assumes that occupancy varies *strongly* as $\mathcal{N}(500, 100^2)$. For each case, we generate a *training data set* and a *validation (test) data set* using two different samples from the corresponding distributions. The *true occupancy values* are the samples obtained from these distributions. Note that both the training and validation data sets use the same weather information; occupancy is the only variable that is assumed uncertain.

We compare the performance of three different GP strategies. The first strategy is the *ideal-GP* strategy in which we assume that occupancy is known with full certainty, and this is used for training the model. The second strategy is the *standard-GP* strategy in which we neglect the presence of occupancy variations; this is equivalent to assuming that occupancy remains constant between days and, consequently, is not used for training the model. The third strategy is the *GP-MCEM* strategy in which we train the model by capturing occupancy variations.

We first demonstrate that GP modeling can be used to accurately predict the energy outputs of the MPC controller given three different input variables. We use the *ideal-GP* strategy in which we use true occupancy values of the training sample for training. We then test the predictions of the model using the validation data set.

In the left panel of Fig. 2 we present the fit and the 95% confidence intervals for the training set obtained with the *ideal-GP* approach. The fit is accurate, demonstrating that GP is able to capture the multivariable interactions in an effective way. This is a remarkable feature given the complex structure of the building model and control system used. Also note that the confidence

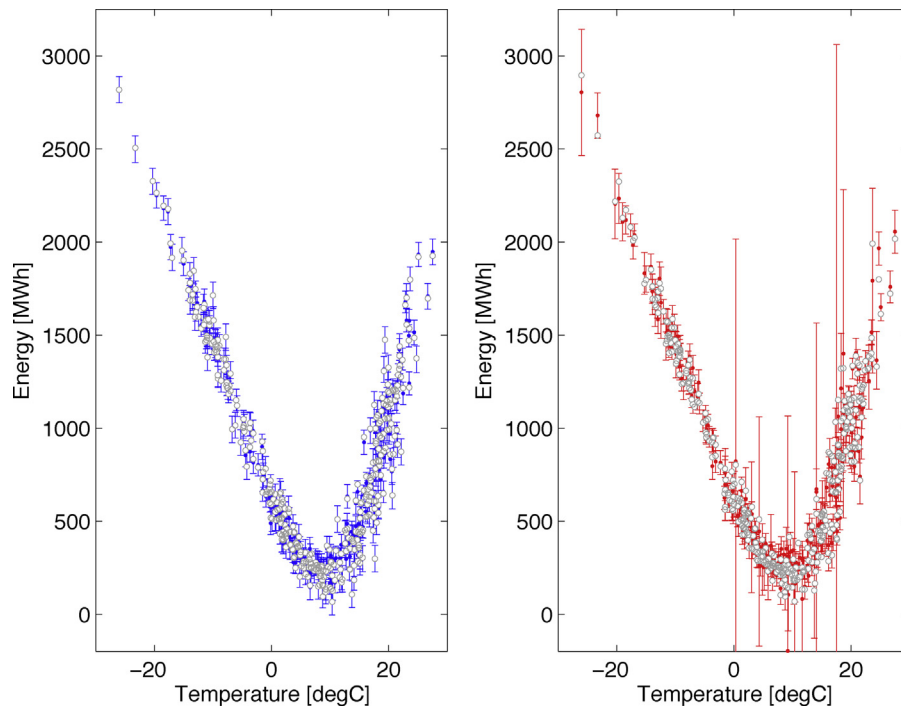


Fig. 2. Ideal-GP case for mild occupancy variation. Fit of training set (left) and prediction at validation set (right).

intervals are narrow implying that the weather and occupancy data are sufficient to explain energy behavior. In the right panel we present the GP predictions (using the hyperparameters obtained from the training set) obtained for the validation set. Surprisingly, we observe significant instability at certain points, reflected as large confidence intervals. In Fig. 3 we present the fit and prediction for the ideal-GP case with high occupancy variability. Note that the higher variability leads to energy demands outside the main trend but the fit of the GP model remains highly accurate. This further

illustrates the ability of GP modeling to capture multivariable interactions. Note, however, that the instability of the predictions in the validation points increases significantly. This is important because it implies that large amounts of training data might be needed to eliminate such instability.

The high sensitivity of the model to perturbations in input occupancy data can be explained by the fact that the training data set did not fully cover the variable range. Consequently, it does not capture the test points in the validation set. This is troubling from

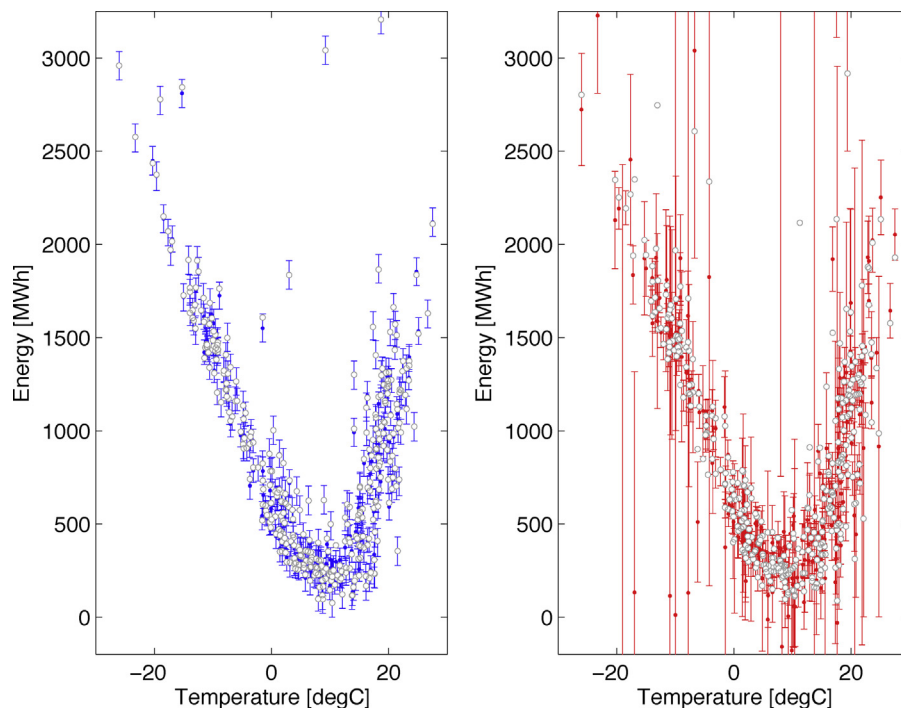


Fig. 3. Ideal-GP case for high occupancy variation. Fit of training set (left) and prediction at validation set (right).

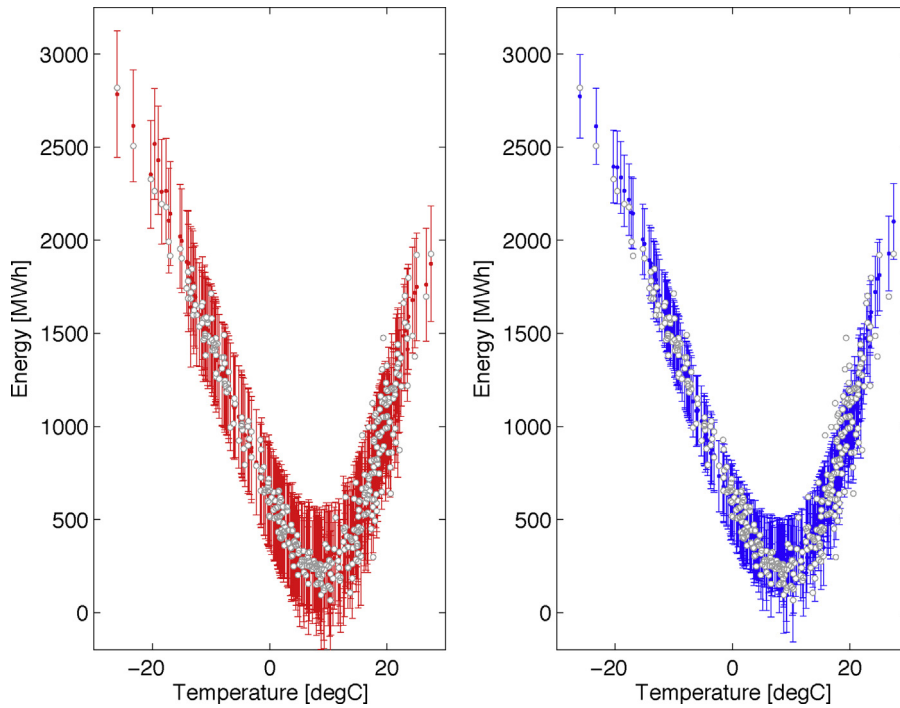


Fig. 4. Fit of standard-GP (left) and GP-MCEM (right) approaches for mild occupancy variation (training set).

a practical point of view because it implies that significant uncertainty will be observed if the M&V baseline model is used to predict energy performance (say, next year) at days with similar weather conditions but slightly different occupancy conditions. The modeler will then be tempted to add those days to the training set to mitigate uncertainty. Continuous data collection will proceed until the predictions stabilize; but they will always be vulnerable to occupancy levels outside the collected range, and large amounts of data will be required. We can thus conclude that the use of “perfect”

information does yield a highly accurate training model but this comes at the expense of significant vulnerability to even slight data perturbations.

In Fig. 4 we compare the training performance of the GP-MCEM approach with that of the standard-GP approach. For the MCEM approach, we used a total of 1000 samples in the E-step. As can be seen, the confidence levels of the standard-GP approach are significantly larger than those of the MCEM approach. The reason is that the standard approach treats the variability in the outputs as noise

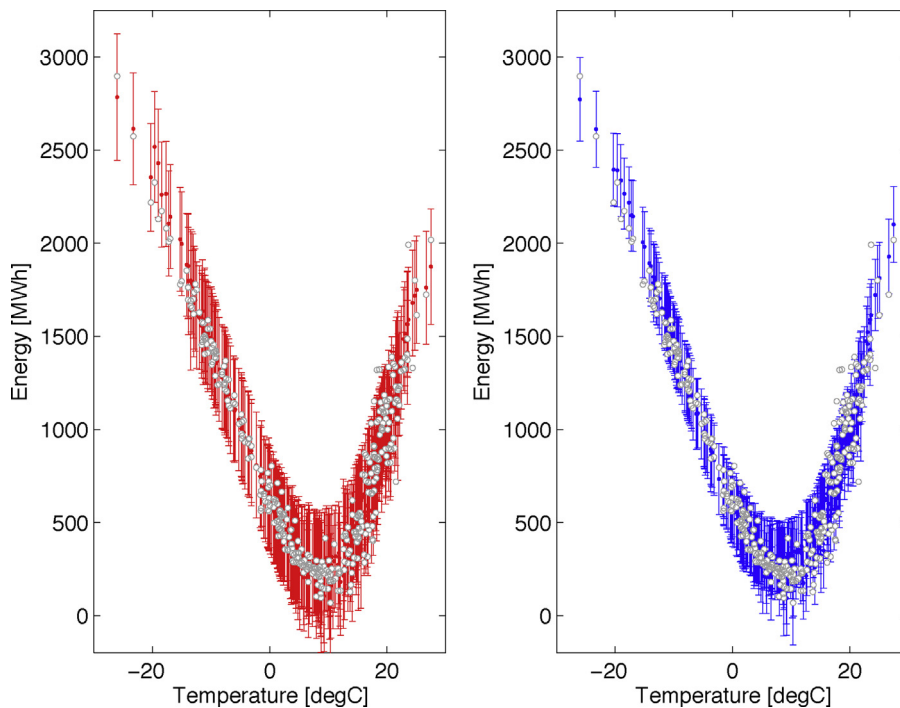


Fig. 5. Prediction of standard-GP (left) and GP-MCEM (right) approaches for mild occupancy variation (validation set).

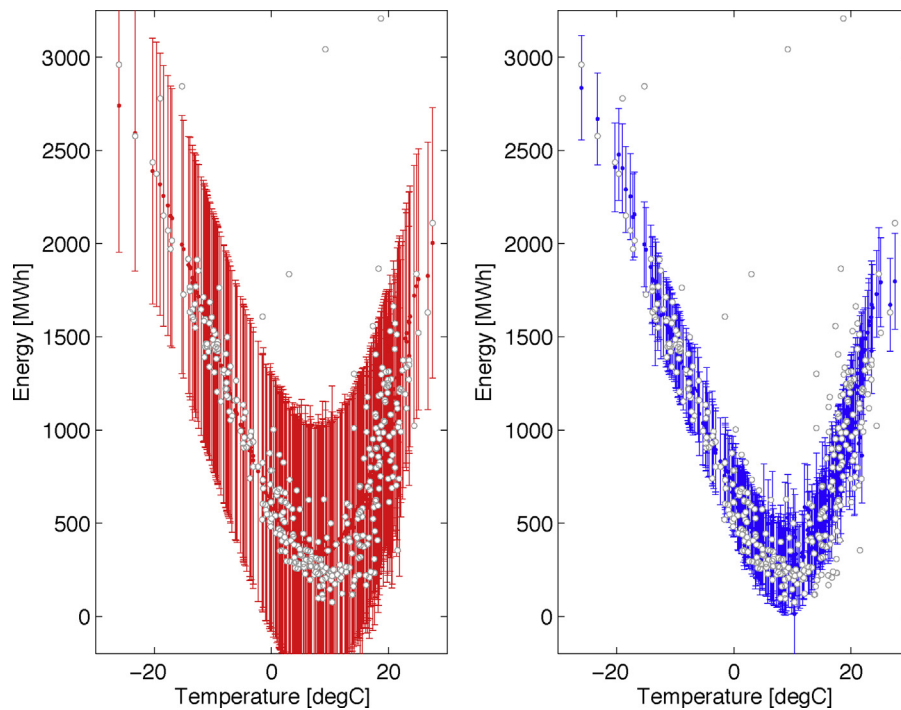


Fig. 6. Fit of standard-GP (left) and GP-MCEM (right) approaches for high occupancy variation (training set).

when in fact a hidden input variable is responsible for the variation. In Fig. 5 we compare the performance of both approaches in the validation set. Figs. 6 and 7 compare performance of standard-GP and GP-MCEM for the case of high occupancy variation. We can see that the confidence intervals of the standard-GP approach reach impractical levels. *The confidence intervals and predictions of the MCEM approach, however, remain robust even in the presence of high variability.*

The high volatility of the confidence levels observed in the ideal-GP approach is eliminated with the GP-MCEM approach. For the mild variation case, we see that neglecting occupancy information (standard-GP approach) yields even less volatility than does the ideal-GP approach. The performance of standard-GP, however, is not competitive as we increase variability. These results raise an interesting property from a data collection standpoint. The GP-MCEM can effectively reduce data needs by capturing variations in

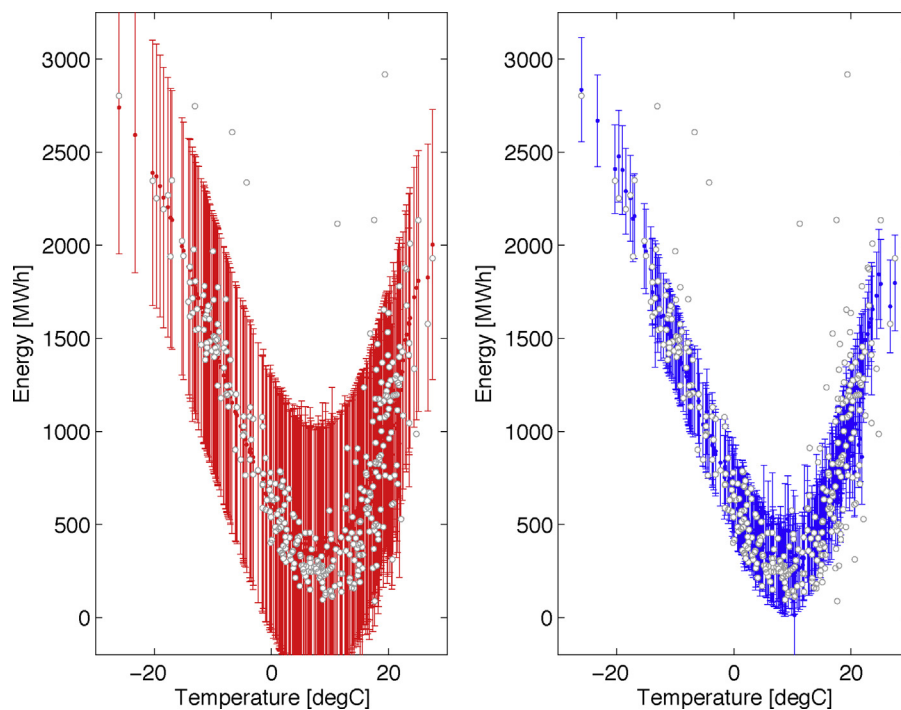


Fig. 7. Prediction of standard-GP (left) and GP-MCEM (right) approaches for high occupancy variation (validation set).

Table 1
Numerical behavior of GP approaches.

Approach	N	Iter	$\log \theta_0$	$\log \theta_1$	$\log \theta_2$
Ideal-GP	–	–	3.206	7.453	3.290
Standard-GP	–	–	2.899	7.245	4.906
GP-MCEM	40	56	3.563	7.925	4.531
	40	4	3.284	7.480	4.452
	40	9	3.447	7.653	4.461
GP-MCEM	200	40	3.658	8.127	4.263
	200	12	3.795	8.480	4.313
	200	12	3.725	8.309	4.340
GP-MCEM	300	9	3.881	8.592	4.272
	300	11	3.838	8.440	4.312
	300	10	3.613	8.016	4.220
GP-MCEM	1000	10	3.872	8.576	4.212
	1000	9	3.898	8.644	4.089
	1000	9	3.863	8.535	4.193

input data because this effectively expands the variable range used for training. Consequently, even if occupancy data were available, we can use the proposed approach to robustify training. In addition, the GP-MCEM approach provides a systematic framework to assess the effect of input data variations for different variables on prediction stability.

We highlight that the proposed GP-MCEM approach can also be used to construct models when limited and/or sparse measurement data are available. For instance, consider the case in which a sensor fails frequently or is simply too expensive to install. In this case, we can construct a GP model by providing a typical range for such measurement (e.g., a uniform distribution). Consequently, the proposed approach can be seen as a mechanism to reduce data collection and sensor costs of M&V processes.

4.2. Numerical behavior

We compare the performance of the GP-MCEM algorithm as a function of the number of samples used in the E-step. We performed three replications for every number of samples tried. The results are presented in Table 1. As can be seen, the number of iterations taken by the GP-MCEM algorithm stabilizes as the number of samples increases. The variability of the hyperparameter estimates tends to decrease as well. Some persistent error exists, however, even in the case with 1000 samples. Note that this is a general deficiency of stochastic algorithms including the one implemented here. For this class of algorithms, convergence can be guaranteed only asymptotically, or one must resort to empirical tests of convergence. Such tests are difficult to implement in the EM context because one cannot easily construct estimators for the log marginal evidence shown in Eq. (2.6). In practice, however, we have not observed significant performance differences in terms of training fit as we vary the number of samples. Consequently, one might consider terminating the algorithm once the fit of the training set stabilizes for a given number of replicates. We note, however, that this approach is ad hoc and thus more research is needed on appropriate termination tests.

From Table 1 we can also observe that the output noise parameter θ_2 is significantly overestimated in the standard-GP approach compared with the other approaches. For the standard approach the logarithmic value is 4.906 while for the GP-MCEM approach the value is around 4.29. At real scale, the former corresponds to 135.1 and the latter to 73.1. This is a difference of nearly 100%, manifested in much wider confidence levels.

5. Conclusions and potential extensions

Accounting for input data uncertainty permits a much more realistic and robust approach to Gaussian process modeling. In

particular, we have demonstrated that significant volatility in predictions can result even in the ideal case where perfect input data is assumed for training. Because the standard GP method ignores uncertainty in input data, the accuracy of energy savings estimates relies more strongly on the quality of data collected for M&V (e.g., number of measured variables, frequency of measurement, accuracy of sensors), which can substantially increase M&V expenses. To overcome this limitation, we proposed a GP framework coupled to a Monte Carlo expectation maximization algorithm (GP-MCEM) to train the GP model under input data uncertainty. We have demonstrated that the approach can significantly increase robustness and mitigate volatility in the predictions. The proposed approach can also be used to construct models when limited and sparse measurement data is available (e.g., by replacing sensor readings with crude estimates and providing ranges). This can in turn reduce data collection and sensor deployment costs of M&V processes.

Several opportunities for future research promise to build upon the demonstrated successes, particularly in the prediction phase in which we explicitly account for input uncertainty [24,27]. This paper focused on demonstrating the benefits of the GP-MCEM framework through an illustrative case study in which only occupancy uncertainty was considered. From a pragmatic standpoint, the current GP-MCEM framework is capable of accounting for uncertainty in multiple input variables as long as they are uncorrelated. For variables exhibiting correlations such as weather variables (outdoor temperature, solar radiation, relative humidity), the current framework will tend to overestimate uncertainty. Consequently, extensions are needed to deal with correlated variables; in particular, more efficient MCMC procedures are needed that are capable of sampling from significantly more complex input distributions in a computationally efficient manner. From an algorithmic standpoint, an ascent-based MCEM method capable of recovering the ascent property of EM with high probability can significantly aid numerical performance [18,20,28,29]. In addition, methods are needed that exploit parallelism and enable matrix-free settings, in order to accelerate computational performance [30].

Acknowledgment

This work was supported by the US Department of Energy, under Contract No. DE-AC02-06CH11357.

References

- [1] C.E. Rasmussen, C.K. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, 2006.

- [2] Y. Heo, V.M. Zavala, Gaussian process modeling for measurement and verification, *Energy and Buildings* 53 (2012) 7–18.
- [3] Y. Heo, D. Graziano, V.M. Zavala, P. Dickinson, M. Kamrath, M. Kirshenbaum, Cost-effective measurement and verification method for determining energy savings under uncertainty, in: *Proceedings of ASHRAE Annual Conference*, 2013.
- [4] Y. Zhang, Z. O'Neil, T. Wagner, G. Augenbroe, An inverse model with uncertainty quantification to estimate the energy performance of an office building, in: *Proceedings of 13th International Building Performance Simulation Association Conference*, 2013.
- [5] B. Yan, A.M. Malkawi, A Bayesian approach for predicting building cooling and heating consumption, in: *Proceedings of 13th International Building Performance Simulation Association Conference*, 2013.
- [6] Y. Heo, R. Choudhary, G.A. Augenbroe, Calibration of building energy models for retrofit analysis under uncertainty, *Energy and Buildings* 47 (2012) 550–560.
- [7] J.E. Yan, Y.J. Kim, K.U. Ahn, C.S. Park, Gaussian process emulator for optimal operation of a high rise office building, in: *Proceedings of 13th International Building Performance Simulation Association Conference*, 2013.
- [8] Y.J. Kim, K.U. Ahn, C.S. Park, I.H. Kim, Gaussian emulator for stochastic optimal design of a double glazing system, in: *Proceedings of 13th International Building Performance Simulation Association Conference*, 2013.
- [9] IPMVP, International Performance Measurement and Verification Protocol: Concepts and Options for Determining Energy and Water Savings, Efficiency Valuation Organization, 2010.
- [10] ASHRAE, Ashrae Guideline 14: Measurement of Energy and Demand Savings, American Society of Heating, Refrigerating, and Air-Conditioning Engineers Inc., Atlanta, GA, 2002.
- [11] Y. Sun, G. Augenbroe, Modeling the urban heat island effect and assessing the impact on building energy consumption, in: *Proceedings of the 2nd International Conference on Building Energy and Environment*, Boulder, 2012.
- [12] Y. Sun, Y. Heo, H. Xie, M. Tan, J. Wu, G. Augenbroe, Uncertainty quantification of microclimate variables in building energy simulation, *Journal of Building Performance Simulation* 7 (1) (2014) 17–32.
- [13] V.M. Zavala, Inference of building occupancy signals using moving horizon estimation and Fourier regularization, *J. Process Control* (2013) (in press).
- [14] I.P. Knight, G.N. Dunn, Evaluation of Heat Gains in UK Office Environments, Worldwide CIBSE/ASHRAE Gathering of the Building Services Industry, Edinburgh, Scotland, 2003.
- [15] P. Hoes, J.L.M. Hensen, M.G.L.C. Loomans, B. de Vries, D. Bourgeois, User behaviour in whole building simulation, *Energy and Buildings* 41 (3) (2009) 295–302.
- [16] J. Quiñero-Candela, Learning with Uncertainty – Gaussian Processes and Relevance Vector Machines, Technical University of Denmark, 2004 (Ph.D. Thesis).
- [17] G.C.G. Wei, M.A. Tanner, A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms, *Journal of the American Statistical Association* 85 (411) (1990) 699–704.
- [18] G. Fort, E. Moulines, Convergence of the Monte Carlo expectation maximization for curved exponential families, *Annals of Statistics* 31 (4) (2003) 1220–1259.
- [19] J.G. Booth, J.P. Hobert, Maximizing general linear mixed model likelihoods with an automated Monte Carlo EM algorithm, *Journal of the Royal Statistical Society, Series B: Statistical Methodology* 61 (1) (1999) 265–285.
- [20] B. Delyon, M. Lavielle, E. Moulines, Convergence of a stochastic approximation version of the EM algorithm, *Annals of Statistics* 27 (1) (1999) 235–251.
- [21] Y. Zhang, W.E. Leithead, Exploiting Hessian matrix and trust-region algorithm in hyperparameters estimation of Gaussian process, *Applied Mathematics and Computation* 171 (2) (2005) 1264–1281.
- [22] Matlab, Optimization Toolbox Users Guide R2013b, The MathWorks Inc., 2013.
- [23] J. Quiñero-Candela, A. Girard, J. Larsen, C.E. Rasmussen, Propagation of uncertainty in Bayesian kernels models – application to multiple-step ahead forecasting, in: *International Conference on Acoustics, Speech and Signal Processing*, 2003.
- [24] V. Zavala M., Real-time resolution of conflicting objectives in building energy management: an utopia-tracking approach, in: *Proceedings of SimBuil*, 2012.
- [25] V.M. Zavala, Real-time optimization strategies for building systems, *Industrial and Engineering Chemistry Research* 52 (9) (2012) 3137–3150.
- [26] A. Girard, C.E. Rasmussen, R. Murray-Smith, Gaussian Process Priors with Uncertain Inputs: Multiple-Step Ahead Prediction, Technical Report TR-2002-119, Department of Computing Science, Glasgow University, Glasgow, Scotland, 2002.
- [27] B.S. Caffo, W. Jank, G.L. Jones, Ascent-based Monte Carlo expectation-maximization, *Journal of the Royal Statistical Society, Series B: Statistical Methodology* 67 (2) (2005) 235–251.
- [28] W. Jank, Implementing and diagnosing the stochastic approximation EM algorithm, *Journal of Computational and Graphical Statistics* 15 (4) (2006) 803–829.
- [29] A. Mihai, J. Chen, L. Wang, A matrix-free approach for solving the parametric Gaussian process maximum likelihood problem, *SIAM Journal of Scientific Computing* 34 (1) (2012) A240–A262.

Further reading

- [17] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society, Series B: Statistical Methodology* 39 (1) (1999) 1–38.