Bayesian Calibration of Building Energy Models for Large Datasets

Zhun Min Adrian Chong

PH.D. THESIS

May 2017

School of Architecture, Carnegie Mellon University

Pittsburgh, PA 15213

Doctoral Committee:

Khee Poh Lam, Chair

Professor, Ph.D, FRIBA

School of Architecture, Carnegie Mellon University

Matteo Pozzi

Assistant Professor, Ph.D

Civil and Environmental Engineering, Carnegie Mellon University

Godfried Augenbroe

Professor, Ph.D, IBPSA-Fellow

School of Architecture, Georgia Institute of Technology

Nyuk Hien Wong

Professor, Ph.D

School of Design and Environment, National University of Singapore

A dissertation submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy in Building Performance and Diagnostics
at the School of Architecture, Carnegie Mellon University

Copyright Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Carnegie Mellon University, Pittsburgh, PA, USA to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I authorize Carnegie Mellon University, Pittsburgh, PA, USA to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

This page is intentionally left blank.

To my supportive parents

and my beloved Shoko

Acknowledgments

Firstly, I express my sincere gratitude to my advisor Professor Khee Poh Lam for his patience, motivation, and immense guidance. He has been supportive since the day I started my Ph.D. study. Besides guidance in academic rigor, his passion and diligence has also influenced me in more ways than one. I have learned a lot and continue to learn from his experience. Khee Poh is truly my advisor, mentor and friend.

I thank my committee member Professor Matteo Pozzi for his advice on the technical details of Bayesian probability. His attention to details and technical expertise has substantially improved the rigor of my research.

My special thanks go to my committee member Professor Godfried Augenbroe for his insightful comments and rigorous research attitude, which pushed me to widen my research from different perspectives. I am also grateful to him for introducing like-minded researchers in the same area of study from Georgia Institute of Technology and University of Cambridge.

I express my sincere gratitude to my committee member Professor Nyuk Hien Wong who has been a mentor to me even before I started my Ph.D. study. I have great respect for him and I very much appreciate his recommendations for applications of my research as well as his guidance on my academic career.

I thank the National University of Singapore for providing me the opportunity to pursue my Ph.D. through the overseas graduate scholarship program. My sincere thanks also goes to Dr. Minuro Yonezawa, Dr. Mikito Iwamasa and Aisu Hideyuki from Toshiba Research and Development Center for their continuous support.

Last but not least, I thank my fellow lab mates for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last five years. They are Weili Xu, Chao Ding, Omer T. Karaguzel, Yunjeong Mo, Shalini Ramesh, Betrand Lasternas, Haopeng Wang, Zhiang Zhang, Jihyun Park, Chenlu Zhang and all my friends at the intelligent workplace and the School of Architecture.

Abstract

Building energy models are increasingly used for the analysis and prediction of a building's energy consumption, to evaluate various energy conservation measures (ECMs), and for measurement and verification (M&V). To ensure their reliability, model calibration has been recognized as an integral component of the overall analysis. In particular, there has been increasing interest in the application of Kennedy and O'Hagen's Bayesian calibration framework to building energy models because of it's ability to naturally incorporate uncertainties. This includes three aspects: 1) uncertainties in calibration parameters; 2) model inadequacy that can be revealed by any discrepancies between model predictions and observed values; as well as 3) observation errors. However, despite several successful applications of Bayesian calibration to building energy models, it has been limited to monthly aggregated data because current methods are computationally prohibitive with hourly or daily calibration data. Current methods also consider a model to be calibrated when its coefficient of variation of the root mean square error (CVRMSE) or normalized mean bias threshold (NMBE) falls below a prescribed threshold set by standards and guidelines such as ASHRAE Guideline 14 (ASHRAE, 2002) and IPMVP (EVO, 2012). However, CVRMSE and NMBE do not check for convergence. If the Markov Chain Monte Carlo (MCMC) algorithm has not proceeded long enough, the generated samples may be grossly unrepresentative of the posterior distribution, and may make interpretation of the posterior distribution for the calibration parameters misleading (Gelman et al., 2014).

In this thesis, a Bayesian calibration method that is computationally acceptable with higher dimension data and large sample sizes is proposed, therefore extending its application to daily and hourly calibration data. This is achieved by: 1) sampling a representative subset of the entire dataset and using the sampled subset for the calibration; and 2) using a more effective MCMC algorithm, the No-U-Turn-Sampler

(NUTS) (Hoffman and Gelman, 2014) to explore the high dimensional posterior distribution. For greater rigor in assessing the calibrated model, we evaluate the model for both accuracy (agreement between observed values and calibrated predictions on test data) and convergence (multiple MCMC chains have converged to a common stationary distribution).

The application of the proposed method is demonstrated using three case studies. In all three case studies, the CVRMSE and NMBE computed with test data were below 15% and 5% respectively. Trace plots of multiple independent chains and Gelman-Rubin statistics $\hat{R}$ (Gelman et al., 2014) also suggests convergence to a common stationary distribution. Through the case studies, the influence of the discrepancy term $\delta(x)$ was also investigated. Results from the case studies show that $\delta(x)$ was able to reduce overall model bias, resulting in a better match between calibrated predictions and observations. Lastly, in the comparison of three MCMC algorithms (NUTS, random-walk Metropolis and Gibbs sampling), NUTS was found to be more effective in generating samples from the posterior distribution.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Building energy modeling or building energy simulation is the use of computer-based simulations to predict and assess a building's energy consumption. Originally intended for use during the design phase, energy models are increasingly used throughout a building's life-cycle (design, commissioning, operations and controls). To ensure reliability and accuracy of the energy model, model calibration has been recognized as an integral component to the overall analysis. Calibration is the process of adjusting model parameters so that the simulation predictions matches the measured data reasonably well.

Over the past decade, there has been growing interest in the calibration of building energy models. A calibrated model can be used to support an energy auditor's recommendations for cost-effective energy conservation measures (ECMs) (Reddy, 2006). Calibrated simulation is also appropriate for measurement and verification (M&V) (ASHRAE, 2002; EVO, 2012) where: 1) pre or post-retrofit data are unavailable or unreliable, but needed to determine the energy savings due to ECMs; 2) there is complex interactions between ECMs and other building systems, making it impractical or impossible to isolate and monitor each subsystem; 3) performance of each ECM needs to be estimated but the cost to isolate and monitor each ECM is prohibitively expensive; 4) only whole-building energy use data is available but savings from each ECM need to be quantified; 5) adjustments to baseline energy use need to be made in order to account for future changes in the building's energy use and demand (e.g., changes in hours of operation, weather conditions, space usage, etc.). Simulation can also be coupled with a building's control system, adjusting model parameters through continuous calibration and using the calibrated model to find an optimal control and response strategy (Augenbroe, 2002).

However, despite the uses and potential benefits of a calibrated model, challenges remain in widespread adoption. Models are only as accurate as the inputs provided. Errors in measured data, the choice of calibration method and model errors make calibration non-trivial. This is magnified by the large number of model parameters as simulation programs become increasingly sophisticated in a trend to include more sub-systems in the model. Consequently, calibrating these models with limited data can often lead to overparameterization and equifinality (i.e., the model parameters are not uniquely identifiable) (Beven, 2006). Since different parameter combination sets can result in similar and reasonably good agreement with measured data, it is important that sources of uncertainty are identified and quantified so that risks are considered when using these models in the decision-making process. This was demonstrated by Heo et al. (2012), where it was shown that by incorporating uncertainty, different ECMs might be preferred depending on the decision-makers' willingness to take risks. For instance, a risk-conscious decision-maker would prefer an ECM that yields a higher probability of guaranteed savings while a risk-taking decision-maker would prefer an ECM that yields the highest expected value.

Due to its ability to naturally incorporate uncertainties, Bayesian calibration has been increasingly employed in the calibration of building energy models. In particular, a Bayesian approach that follows that of Kennedy and O'Hagan (2001) has been increasingly used for the calibration of building energy models (Heo et al., 2012; Riddle and Muehleisen, 2014; Heo et al., 2015a,b; Li et al., 2016). This is because the formulation proposed by Kennedy and O'Hagan (2001) explicitly quantifies uncertainties in the calibration parameters, uncertainty due to discrepancy between the model and the actual physical system, as well as observation errors. Bayesian approaches also make use of prior probabilities, which can be used to represent expert knowledge or the current state of knowledge. However, the specification of prior probabilities requires a lot of thought and analysis. This is because if the priors have not been properly specified, the resulting posterior distribution of the calibration parameters can be misleading. This is especially true if small amounts of data are used since the posterior would be largely influenced by the prior.

With the emergence of Internet of Things (IoT) and as increasing number of sensors get deployed in buildings, there is an opportunity to investigate methods that effectively use this data within a calibration framework. This would include continuous calibration for M&V, operations and controls, adding another motivation towards a Bayesian approach for calibration. Such an approach provides a flexible framework for dynamically updating the calibrated model. As new data arrive, the old data is not discarded but instead assimilated to the new data through the use of priors. In other words, the previous posterior density acts the prior for the current calibration, thus providing a very systematic framework for the continuous calibration or updating of the energy model.

However, despite several successful applications of Bayesian calibration to building energy models, challenges remain in its application to building energy models. First, Bayesian calibration is typically carried out using random-walk Metropolis or Gibbs sampling (Heo et al., 2015b; Li et al., 2016). An inherent inefficiency of these algorithms can be attributed to their random walk behavior as the Markov Chain Monte Carlo (MCMC) simulation can take a long time zig-zagging while moving through the target distribution (Gelman et al., 2014).

Second, current application of Kennedy and O'Hagan (2001) uses a Gaussian process (GP) model to emulate the building energy model (Riddle and Muehleisen, 2014; Heo et al., 2015a,b). Although accurate, training GP models with large datasets is computationally prohibitive and thus has been limited to monthly calibration data. An alternative would be to apply the formulation to the original simulation model if the simulation can be made to run sufficiently fast, circumventing the use of an emulator as demonstrated in Higdon et al. (2004). Although this could help remove any uncertainty arising from using an emulator, it is cumbersome to automate the entire process. This is because an intermediary platform would need to be developed to enable data exchange between the simulation tool and the calibration process. For instance, to calibrate an energy model using a Bayesian approach, Chong and Lam (2015) reprogrammed two EnergyPlus objects in Python. Given the large variety of simulation tools currently available and the complexity of some tools, the platform can be difficult or impossible to develop. Furthermore, it would be hard to maintain such a platform since it would need to be constantly revised to keep up with any version updates of different tools. On the contrary, using a GP model provides the flexibility for users to specify the inputs and output to a calibration framework without the need of an intermediate platform between the simulation tool and the calibration process. Therefore, the use of an emulator or a surrogate model to represent the original simulation model is an important step for an automated calibration framework.

Third, model validation procedures do not place enough emphasis on evaluating model fitness. Currently, the quantification of uncertainties in input parameters is considered correct when the model's output meets the error criteria set out by ASHRAE Guideline 14 (ASHRAE, 2002). However, if the MCMC algorithm has not proceeded long enough, the generated samples may not be representative of the target distribution (Gelman et al., 2014). Evaluating model convergence statistics provides more confidence that the samples generated are representative of the posterior distribution of the calibrated parameters.

1.2 Building energy simulation tools

To date, a large number of building energy simulation tools have been developed. These tools are used by engineers, architects and researchers to model energy consumption in buildings and usually require large number of inputs from its users. Different tools provide users with varying levels of flexibility and thus need a different perspective on calibration. Generally, building energy simulation tools can be categorized as being either open or closed tools. Open tools are softwares that are usually open source and expose all of their internal parameters and calculation methods. As a result, these tools are more suitable for calibration because users typically have access to the source code and thus are able to better understand the underlying interaction between different model parameters. This is particularly important if the objective of the calibration is not only for accurate predictions but also to gain a better understanding of the calibration parameters. On the contrary, closed tools only expose certain parameters and thus the calibration is typically handicapped. Here, tools such as EnergyPlus is categorized as open. This is because although some parameters are hidden from users and despite the fact that it is cumbersome to make changes to these hidden parameters, it is not impossible given that they are open source. Therefore, examples of open tools are:

- **DOE-2** (Winkelmann et al., 1993): a building energy simulation tool that predicts the hourly energy use of a building given hourly weather information, and a description of the building's geometry and HVAC system. Funded by the U.S. Department of Energy, DOE-2 has been extensively used in the past with more than 20 interfaces created to make it easier to use (Crawley et al., 2008).
- **EnergyPlus** (Crawley et al., 2001; LBNL, 2016b): a whole building energy modeling tool developed based on the best features and capabilities of DOE-2 and Building Loads Analysis and System Thermodynamics (BLAST) (Hittle, 1979). EnergyPlus provides an integrated solution where the building loads and the HVAC systems are tightly coupled. It also provides sub-hourly, user-definable time steps for interactions between the thermal

zones and the HVAC systems. Other features include the ability to model multi-zone airflow and a wide variety of building and HVAC design options.

- **TRNSYS** (Klein et al., 2012): a transient systems simulation program designed with a modular structure. Energy calculations were solved by breaking them into a series of individual components. Components may vary from simple systems such as parts of a HVAC system (a pump or chiller) to more complex systems such as multi-zone buildings.
- **ESP-r** (ESRU, 1974): a building performance simulation tool equipped with the capability to model heat, air, moisture, light and electrical power flows at spatial and temporal resolutions specified by the user. It has been in development since 1974 with the intention to enable an integrated assessment of all aspects of building performance.

Examples of closed tools are:

- **IESVE** (Integrated Environment Solutions, 2015): The IES virtual environment is an integrated suite of applications linked by a common user interface and a single integrated data model. With a single model, the tool provides a platform for an analysis of a building's integrated performance.
- **Sefaira** (Trimble Buildings, 2017): Sefaira is designed so that project teams can quickly explore design options and understand their impact on building performance (energy and daylight outputs). It uses a proprietary Scala based Fulcrum engine to calculate the hourly energy use.

1.3 Uncertainty in building energy simulation

Buildings are complex systems composed of many components interacting with one another (Figure 1.1). Additionally building energy models are representations of the actual physical systems. Therefore, no single model is beyond dispute. To model these complex interaction, building energy models are increasing in complexity, requiring more parameters to be defined as inputs to the model. These inputs include but is not limited to a detailed description of the building's geometry, it's associated HVAC system, the quantification of various internal loads (occupancy, lighting, equipment loads, etc.), as well as weather conditions. However, in many cases, direct measurement of many parameters is impractical or impossible. Thus, in most practical situations, calibrating an energy model is an inverse problem that is ill-posed because the data that is available is typically insufficient for identifying a unique solution. As a result, calibrating these models with limited data might lead to identifiability issues (different parameter sets might give reasonable matching results with measured data) and a wide variety of model uncertainties despite the model having been calibrated. One possible approach is to recognize that the available data is not enough to determine a single solution and then generate equally possible candidates to represent the actual solution.

According to (De Wit and Augenbroe, 2002) uncertainty in building energy models can be classified as:

- Specification uncertainty: arising from incomplete or inaccurate specification of the building or systems being modeled.
- Modeling uncertainty: arising from simplifications and assumptions of complex physical processes. These simplifications and assumptions could be explicit to the modeler (such as thermal zoning) or hidden within the tool (calculation algorithms)
- Numerical uncertainty: arising from errors introduced in the discretization and simulation of the model
- Scenario uncertainty: arising from external conditions imposed on the building. Examples

include outdoor weather conditions and occupant behavior.

Several studies have been carried out focusing on uncertainty quantification. Sun et al. (2014) provided a procedure to quantify uncertainties in the microclimate variables used by building energy models. Macdonald and Strachan (2001) reviewed uncertainties in the thermophysical properties of construction materials and incorporated them into the building energy simulation tool ESP-r using Monte Carlo Analysis. Eisenhower et al. (2012a) modeled 1009 EnergyPlus input parameters as uncertain by varying them $\pm 20\%$ of their nominal value. Using sensitivity analysis, they provided insights to how uncertainty in input parameters may affect model output. To make the evaluation of retrofits more reliable, IPMVP has also published a document that provides guidance on uncertainty quantification (EVO, 2014).

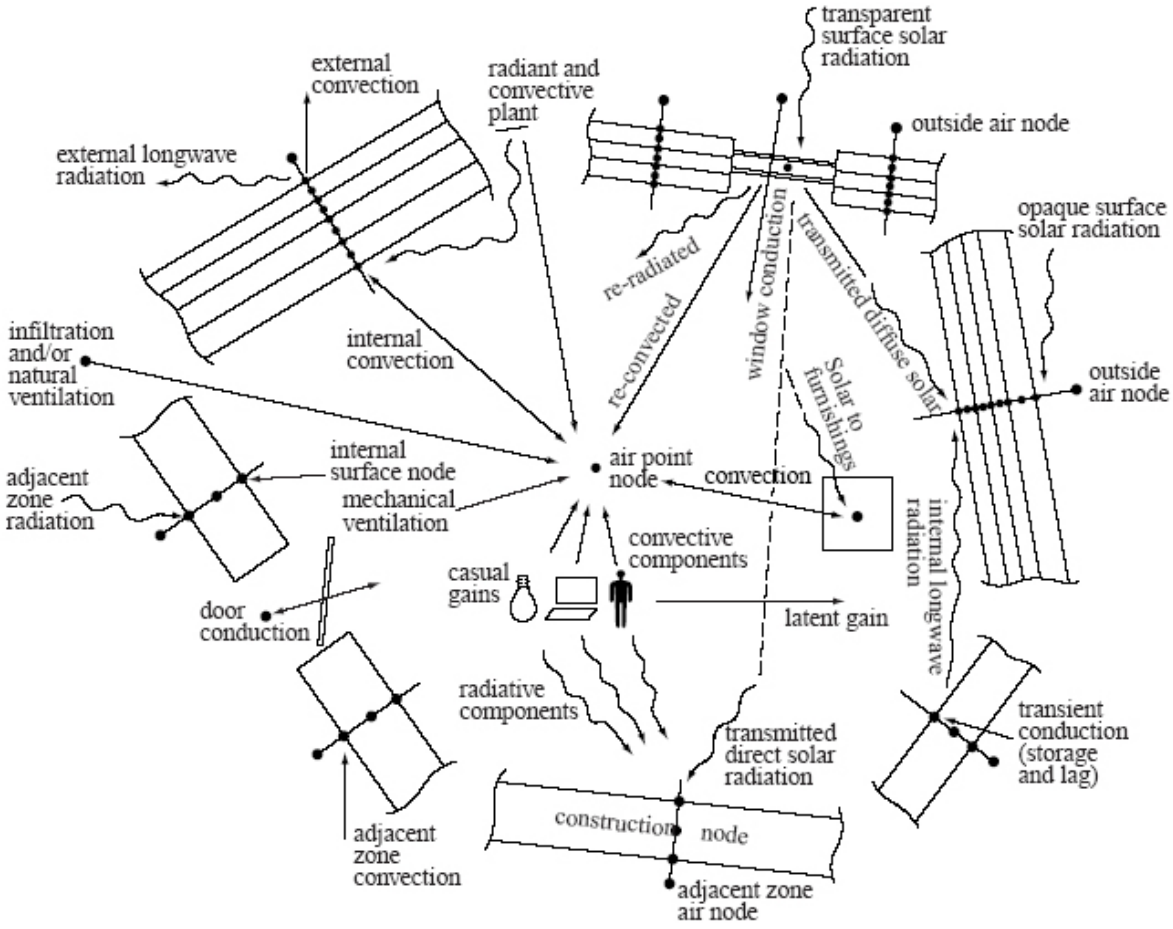

Figure 1.1: Building energy flowpaths, taken from (Clarke, 2001)

1.4 Current calibration approaches

Many approaches for calibrating building energy models have been proposed, requiring various degrees of automation, manual tuning and expert judgment. According to Coakley et al. (2014), calibration approaches for building energy simulation can be broadly defined as either manual or automated.

1.4.1 Manual calibration approaches

Calibration approaches that fall under this category typically requires the energy modeler to perform iterative manual tuning (Reddy, 2006; Coakley et al., 2014) and to have in-depth knowledge of the building and its operation. Therefore, manual calibration approaches usually involve either 1) detailed energy audits to gain a better understanding of the building systems and their operations (Ian Shapiro, 2009; Pedrini et al., 2002; Yoon et al., 2003); 2) intrusive tests where groups of end-use loads are turned on and off in a controlled sequence to provide information on their end-use impact (Soebarto, 1997); 3) collection of high-resolution and high-quality data for empirical validation (Clarke et al., 1993); or 4) a protocol of short-term end-use monitoring to gather data that would help explain the differences between measured and simulated data (Subbarao, 1988; Manke et al., 1996).

Manual tuning processes are usually facilitated by analytical tools such as graphical plots. Examples include time-series plots, box and whisker plots and scatter plots (Reddy, 2006). Graphical plots of calibration signature and characteristic signature have also been proposed to guide the parameter tuning process (Liu and Liu, 2011). Due to its iterative nature, the use of version control to keep track of model changes and the reasons for the change has been identified as an important process in order to improve the reproducibility of manual calibration methods (Raftery et al., 2011).

Although shown to be successful in several case studies, manual approaches suffer from several drawbacks. First, it is time consuming and labor intensive to calibrate a model based on trial and error, iterating between tuning model parameters and checking the accuracy of its predictions against measured data. Second, the calibration process relies largely on the expertise and skills of the modeler, making its reproducibility questionable, and thus restricting its widespread adoption. Third, they do not consider model uncertainties.

1.4.2 Automated calibration approaches

To overcome the drawbacks of manual calibration approaches, there has been increasing research towards the development of modern algorithms to automate or partially automate the calibration process. Compared to manual approaches that require the modeler to adjust model parameters in a heuristic and iterative manner, analytical or mathematical methods automatically select the parameters to tune and the amount they are adjusted by (Reddy, 2006). These approaches usually involve one or more of the following techniques: 1) meta-modeling; 2) optimization with an objective or penalty function; and 3) Bayesian calibration.

Meta-modeling involves the use of data-fit models as surrogates for the computationally complex building energy models such as those mentioned in Chapter 1.2. The use of meta-models as surrogates was aimed at reducing the cost of forward simulations. This is because advance calibration methods such as those that employ optimization algorithms and Bayesian calibration frameworks are typically iterative processes that require a large number of simulation runs (Reddy, 2006). Although building energy simulations have become less computationally intensive, it is still time consuming to run large number of simulations, especially if these simulations are to be run sequentially (output of current simulation is required to determine the inputs for the next simulation would be varied). Therefore, a key element of many optimization and Bayesian calibration approaches is the use of meta-models to carry out the inference during the calibration process, mapping the energy model's input parameters to the model's output. Examples of meta-modeling techniques that have been used to model building energy models include multiple linear regression (Li et al., 2016), support vector regression (Dong et al., 2005; Eisenhower et al., 2012b), neural networks (Neto and Fiorelli, 2008) and Gaussian Processes (Heo et al., 2012; Manfren et al., 2013).

Optimization can be used to automate the calibration process by defining an objective function that minimizes the monthly or hourly mean squared difference between simulation predictions and measured data. However, there may be an issue of identifiability, where many solution sets meet the defined objective function. To address this issue, penalty functions (functions that penalize solutions that differ significantly from its preferred value) can be defined to prevent unreasonable parameter values during the calibration process (Carroll and Hitchcock, 1993). The use of optimization programs such as GenOpt (LBNL, 2011) developed to couple optimization algorithms (e.g., genetic algorithms and particle swarm optimization) with building energy simulation have also been used to aid calibration efforts (Taheri et al., 2012).

More recently, there have been increasing efforts in a Bayesian approach for the calibration of building energy models. This is because of its ability to quantify model uncertainties while at the same time reducing discrepancies between simulation output and physical measurements. In reality, detailed information are seldom available. Availability of high quality data might also be limited where the installation of large number of sensors are prohibitively expensive or impractical. Arguably, uncertainty quantification becomes an important process in the use of building energy models. Consequently, issues related to prediction accuracy and prediction uncertainty would be of particular interest when using these models to make decisions. Probabilistic predictions also offer decision-makers greater confidence when using the model for decision-making. In the next section, we provides a review of the current state of the art with respect to the Bayesian calibration of building energy models.

1.4.3 Bayesian approach to calibration

Bayesian calibration was first applied to building energy models by (Heo et al., 2012) using the formulation proposed by Kennedy and O'Hagan (2001). The formulation represents the relationship between the observations y and the output of the simulator $\eta(.,.)$ by explicitly modeling uncertainty in the calibration parameters as well as uncertainty due to discrepancy between the simulator and actual physical system, and observation errors (Equation 1.1).

$$y(x) = \eta(x,t) + \delta(x) + \epsilon(x) \tag{1.1}$$

where,

$y(x)$ is the observed value,

$\eta(x, t)$ is the output from the building energy model given the input vector (x, t),

x is the known input factors,

t is the unknown calibration parameters,

$\delta(x)$ is the discrepancy between the simulation output and the observed output,

$\epsilon(x)$ is the observation error.

The application of Kennedy and O'Hagan (2001) formulation to building energy models was first seen in Heo et al. (2012), where a reduced order quasi-steady-state energy model was calibrated against monthly measured gas energy consumption. The method proposed by Heo et al. (2012) can be summarized by the following steps: 1) define prior probability distributions of uncertain parameters; 2) screen parameters to reduce the number of calibration parameters; 3) utilize a Gaussian process (GP) model to carry out the inference; and 4) explore the posterior distributions using Markov Chain Monte Carlo (MCMC). The same approach was also applied to a large portfolio of buildings (Heo et al., 2015a). To extend the method proposed by Heo et al. (2012) to more complex dynamic building energy models, Li et al. (2016) utilized a multiple linear regression emulator instead of a GP emulator, reducing computation cost with a slight loss in accuracy. Through this simplification, an EnergyPlus model was calibrated against monthly observations.

Another formulation that has been applied to building energy models does not separate different sources of uncertainty but instead combines them under a single additive model error $\epsilon(x)$ in the form:

$$y(x) = \eta(x, t) + \epsilon(x) \tag{1.2}$$

Using this formulation, Manfren et al. (2013) calibrated a Gaussian process meta-model that has been trained on simulation data. The single model error formulation was also used to apply a

Bayesian approach in the calibration of an EnergyPlus boiler and chiller model (Chong and Lam, 2015). More recently, Equation 1.2 has been used to calibrate a set of energy models in an urban context (Sokol et al., 2017). It is important to note that the formulation proposed by Kennedy and O'Hagan (2001) (Equation 1.1) improves on the single additive model error formulation (Equation 1.2) by accounting for the discrepancy between the model predictions and the actual observations.

1.4.4 Assessing calibration performance

Presently, a building energy simulation model is considered "calibrated" when its predetermined statistical index falls below a specified threshold. Table 1.1 lists the thresholds set out by different standards and guidelines. Commonly used statistical indices used to evaluate the performance of an energy model include the normalized mean bias error (NMBE) (Equation 1.3), the root mean squared error (RMSE) (Equation 1.4), and the coefficient of variation of the root mean squared error (CVRMSE) (Equation 1.5).

NMBE also known as mean bias error (MBE) is computed as the sum of differences between measured and simulated data. The sum is then normalized by the sum of measured data (Equation 1.3). NMBE serves as a good indicator of overall bias in the simulated data, providing an indication as to whether the predicted values tend to overestimate or understimate the actual values (EVO, 2012; ASHRAE, 2002). However, positive bias may be compensated by negative bias resulting in a cancellation of bias (Coakley et al., 2014). Hence, another indicator of model error should be used in conjunction with NMBE.

RMSE provides a measure of the variability in the data, or the amount of spread existing in the data (ASHRAE, 2002). It is computed as the sum of squared differences between measured and simulated data (Equation 1.4). The sum is then normalized by the number of data points and a square root of the result gives the RMSE.

CVRMSE is computed by dividing RMSE by the mean of the measured data (Equation 1.5). The CVRMSE provides a measure of how well the simulated data fits the actual values. Unlike NMBE, both RMSE and CVRMSE do not suffer from the cancellation effect since the errors are squared before summing. According to ASHRAE (2002), it is easier to achieve low NMBE than CVRMSE. NMBEs are typically reported to be in the range of $\pm 5\%$ to $\pm 10\%$. On the contrary, the best empirical models of building energy use performance were only capable of producing CVRMSE in the range of 10% to 20%.

$$NMBE(\%) = 100 \times \frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)}{(n-1) \times \bar{y}} \tag{1.3}$$

$$RMSE(\%) = 100 \times \sqrt{\frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n-1}} \tag{1.4}$$

$$CVRMSE(\%) = 100 \times \frac{\sqrt{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \,/\, (n-1)}}{\bar{y}} \tag{1.5}$$

where,

n is the number of observations,

y_i is the i^{ith} observation,

$\hat{y}_i$ is the i^{ith} prediction,

$\bar{y}$ is the mean of the observations.

Table 1.1: Error criteria for model to be deemed calibrated

Standard/Guideline	Monthly Criteria $(\%)$		Hourly Criteria $(\%)$	
	NMBE	CVRMSE	NMBE	CVRMSE
ASHRAE Guideline 14 (ASHRAE, 2002)	5	15	10	30
IPMVP (EVO, 2012)	-	-	5	20
FEMP (DOE, 2008)	5	15	10	30

However, one caveat is that the “calibrated” model may not be representative of the actual building performance since various combinations of inputs can still produce reasonable agreement between measured and simulated data, i.e., there can be several models that can be considered as “calibrated” (Coakley et al., 2014). Furthermore, since these calibration criteria are based solely on energy consumption and do not account for uncertainties and inaccuracies of the input parameters, this makes their reliability questionable. Calls have also now been made to use hourly measured data for calibration purposes for the reason that the model would better represent the building's actual performance, since calibration with monthly data could easily miss significant errors at a daily or hourly resolution (Reddy, 2006; Raftery et al., 2011).

1.5 Objective and hypothesis

The objectives of this thesis are to:

- Propose a Bayesian calibration method that is computationally acceptable for calibrating building energy models against daily or hourly calibration data, with a focus on improvements to the implementation of Bayesian calibration to building energy models.
- Improve the reproducibility and repeatability of the Bayesian calibration of building energy models by providing guidelines on calibration procedure and the evaluation of model fitness

The hypotheses of this thesis are:

- Using a representative subset D_{sub} of the entire dataset D, determined based on information divergence (Kullback and Leibler, 1951) between D_{sub} and D, in the Bayesian calibration of building energy models can provide sufficient accuracy with lower computation cost, according to the thresholds of CVRMSE and NMBE specified by ASHRAE Guideline 14 (ASHRAE, 2002).
- Using the No-U-Turn sampler (NUTS), an extension of Hamiltonian Monte Carlo (HMC) would improve computation efficiency of the Bayesian calibration of building energy models because (compared to random-walk Metropolis and Gibbs sampling) NUTS is able to achieve faster convergence in high-dimensional problems, based on convergence tests using the Gelman-Rubin statistics $\hat{R}$ (Gelman et al., 2014) and trace plots of multiple MCMC chains.

1.6 Organization of thesis

This thesis is outlined as follow:

- Chapter 1 describes the motivation behind a Bayesian approach to calibration and a Bayesian calibration method that is not computationally prohibitive when applied to hourly or daily calibration data or when applied to commonly used dynamic building energy simulation tools.
- Chapter 2 describes the proposed method and the additions to the current Bayesian calibration method so that computationally it is scalable to larger datasets, i.e. daily or hourly calibration data; and high-dimensional problems, i.e. greater number of calibration parameters of a complex building energy model.
- Chapter 3 demonstrates the application of the proposed Bayesian calibration approach to three case studies. Using the three case studies, we show that the proposed method is sufficiently accurate but at the same time its computation cost does not prohibit calibration with daily or hourly data.
- Chapter 4 empirically compares the effectiveness of three MCMC algorithms (NUTS, RWM and Gibbs sampling) within the proposed Bayesian calibration method described in Chapter 2. The comparison is done using the three case studies described in Chapter 3 .
- Chapter 5 summarizes the main findings of the thesis, draw conclusions and provides suggestions for future research.

Chapter 2

Proposed Bayesian Calibration Method

2.1 Overview

Figure 2.1 shows an overview of the proposed method. The proposed method is an extension of the Bayesian calibration method first applied to building energy models by Heo et al. (2012) and is based on Kennedy and O'Hagan (2001) Bayesian approach for calibrating computer models. The additions (highlighted in red in Figure 2.1) include: 1) sampling a representative subset from the entire dataset and using the sampled subset for the calibration; 2) using the No-U-Turn Sampler (NUTS), an extension to Hamiltonian Monte Carlo (HMC), to explore the posterior distribution; and 3) evaluating the calibrated model for both accuracy (agreement between observations and calibrated predictions on test data) and convergence (multiple MCMC chains have converged to a common stationary distribution).

The proposed Bayesian calibration method can be summarized as follows:

1. Create an energy model based on information that provides a preliminary understanding of the building, which includes construction drawings, design specifications, measured data, site visits, operation documents, etc.
2. Conduct sensitivity analysis to discern the influential calibration parameters t from the set of uncertain parameters θ that has been identified.

3. Run a fixed number of simulations m at the same observable input factors x. Using the simulation predictions $\eta(x,t)$ and the observed values $y(x)$, a field dataset $D^F = \begin{bmatrix} y(x) & x \end{bmatrix}$ and a simulation dataset $D^S = \begin{bmatrix} \eta(x,t) & x & t \end{bmatrix}$ is defined.

4. Use information theory to select a representative subset of the field data D^F_{sub} and a representative subset of the simulation data D^S_{sub}.

5. Combine D^F_{sub} and D^S_{sub} in a Gaussian process (GP) emulator using the approach proposed by Higdon et al. (2004).

6. Use NUTS (Hoffman and Gelman, 2014) to explore the posterior distribution of the calibration parameters t, correlation hyperparameters of the GP model β^η and β^δ, and variance hyperparameters of the GP model λ_η, λ_δ and λ_ϵ.

7. Evaluate performance of the calibrated model using two performance metric, which includes: 1) goodness of fit between model predictions and observations on a test data (CVRMSE and NMBE); and 2) convergence of the generated samples (Gelman-Rubin Statistics $\hat{R}$ and trace plots of multiple chains).

A detailed explanation of the proposed method is provided in Chapters 2.2 and 2.4.

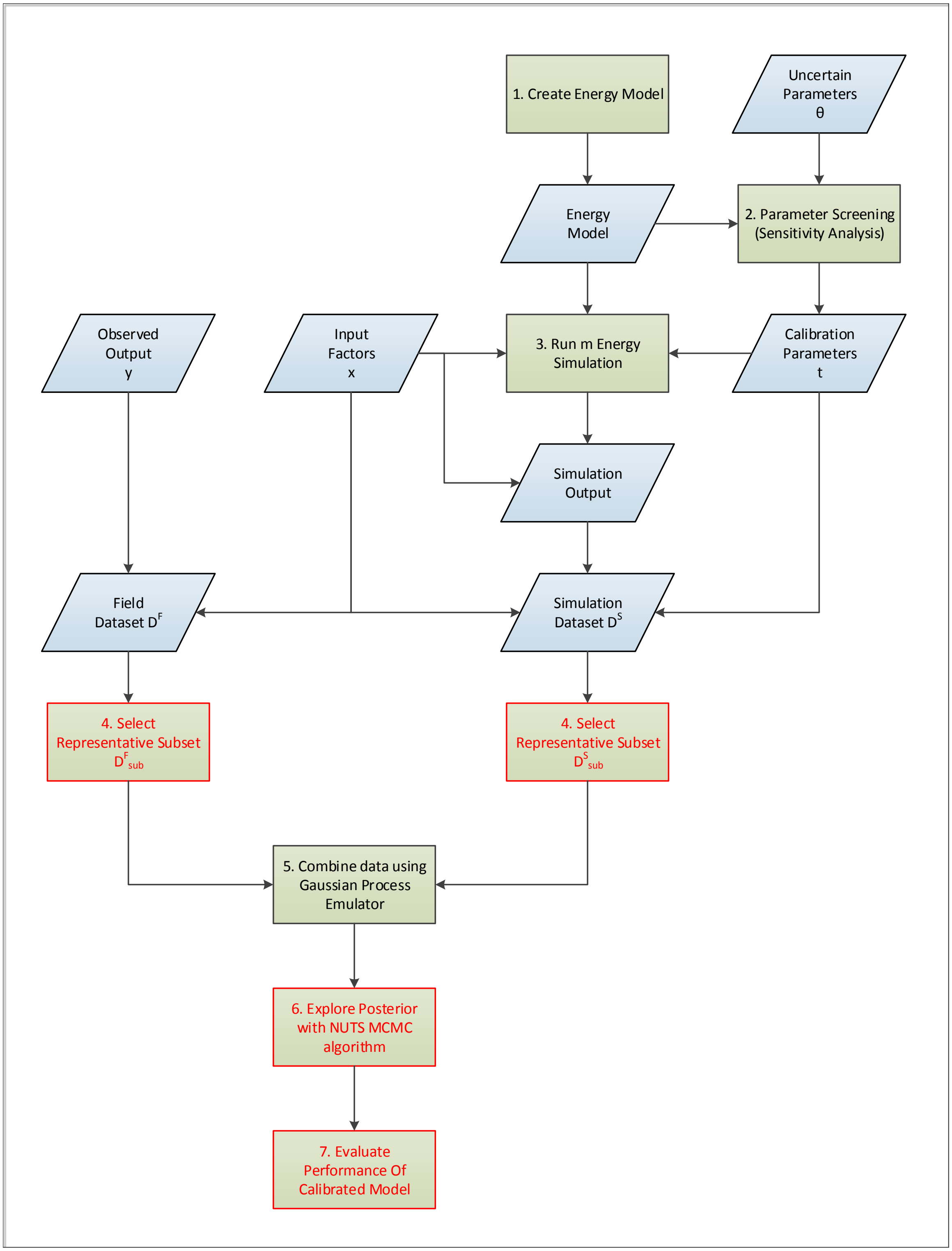
1. Create Energy Model
Uncertain Parameters θ
Energy Model
2. Parameter Screening (Sensitivity Analysis)
Observed Output y
Input Factors x
3. Run m Energy Simulation
Calibration Parameters t
Simulation Output
Field Dataset D^F
Simulation Dataset D^S
4. Select Representative Subset D^F_{sub}
4. Select Representative Subset D^S_{sub}
5. Combine data using Gaussian Process Emulator
6. Explore Posterior with NUTS MCMC algorithm
7. Evaluate Performance Of Calibrated Model

Figure 2.1: Overview of proposed approach.

2.2 Current Bayesian calibration method

2.2.1 Sensitivity analysis (parameter screening)

Building energy models require a large number of parameters as inputs, many of which are impractical or impossible to directly measure. As a result, these models often contain a large number of uncertain parameters. Calibrating such a model with limited data can often lead to overparameterization and issues of identifiability, i.e., different combinations of calibration parameters could result in the same model output. Additionally, it is also computationally costly to calibrate a model with a large number of uncertain parameters. A solution is to use sensitivity analysis to identify and screen non-sensitive parameters from the set of uncertain parameters θ. The resulting influential parameters which we denote with t are then subsequently use for the Bayesian calibration of the model.

Sensitivity analysis was carried out using the Morris method (Morris, 1991). This was executed using R sensitivity package (Pujol et al., 2016). The method of Morris belongs to the class of One-factor-At-a-Time (OAT) design (only one parameter changes values between consecutive simulations) and is suitable when the number of input factors are so large that other variance-based approaches are computationally prohibitive (Saltelli et al., 2008). Therefore, it is a common technique for carrying out sensitivity analysis in building energy models (Heo et al., 2012; De Wit and Augenbroe, 2002; Tian, 2013; Menberg et al., 2016; Kristensen and Petersen, 2016). The main advantage of the Morris method is its relatively lower computation cost as compared to other global sensitivity analysis methods, making it particularly well-suited for use with building energy models where the number of uncertain parameters is high.

The Morris method is based on calculating the elementary effects of each uncertain parameter θ. Then, the overall effect and interaction effect of each parameter is computed to determine each parameter's sensitivity. Suppose there are k parameters and each parameter space is divided into h levels. In other words, the parameter space is a k-dimensional h-level orthogonal grid. For each given value of parameter θ, the elementary effect of the i^{th} input is defined as:

$$d_i(\theta) = \left(\frac{f(\theta_1, \theta_2, ..., \theta_{i-1}, \theta_i + \Delta, \theta_{i+1}, ..., \theta_n) - f(\theta)}{\Delta} \right) \tag{2.1}$$

where,

$d_i(\theta)$ is the i^{th} elementary effect,

n is the number of parameters,

θ_i is the i^{th} parameter,

Δ is a predetermined multiple of $1/(h-1)$ and as suggested by Morris (1991), h is even and Δ is set equal to $h/[2(h-1)]$.

For sampling with the Morris method, r points are first randomly generated from the k-dimensional h-level grid. Each of the r points are then perturbed one dimension at a time with a step size of $h/[2(h-1)]$ until all k parameters have been varied once (Figure 2.2). The cost of the experiment is thus $r \times (k+1)$.

Using the r trajectories, the commonly used sensitive measures that includes the modified mean μ^* (Campolongo et al., 2007) and the standard deviation σ of each of the i^{th} parameter can then be computed by:

$$\mu_i^* = \sum_{j=1}^{r} |d_{ij}(\theta)|/r \tag{2.2}$$

$$\sigma_i = \sqrt{\sum_{j=1}^{r} (d_{ij}(\theta) - \mu_i)^2/r} \tag{2.3}$$

where,

r is the number of trajectories,

μ_i^* is the modified mean of the i^{th} elementary effect $d_i(\theta)$,

μ_i is the mean of the i^{th} elementary effect $d_i(\theta)$,

σ_i is the standard deviation of the i^{th} elementary effect $d_i(\theta)$.

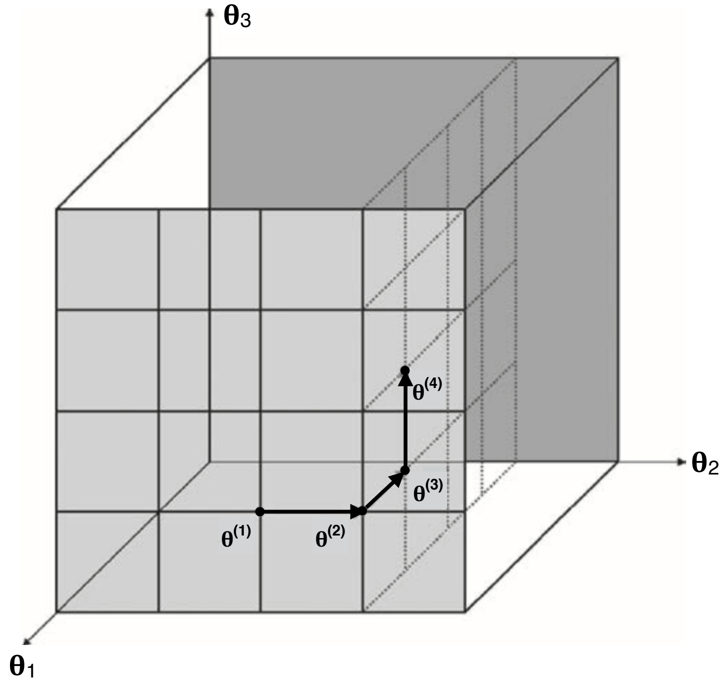

Figure 2.2: Example of a trajectory with $h = 5$ levels and $k = 3$ parameters/factors $(\theta_1, \theta_2, \theta_3)$, adapted from Saltelli et al. (2008).

In order to better interpret the sensitivity measures μ^* and σ, a graphical plot of μ^* against σ will be used (Saltelli et al., 2008). Parameters with low μ^* and σ are deemed non-influential and not used in the calibration process. Campolongo et al. (2007) defines μ^* as the mean of the distribution of the absolute values of the elementary effects. In other words, μ^* evaluates the overall influence of the input factor on the output. On the other hand, σ evaluates the factor's effect that is due to either interactions with other factors or/and curvature (Saltelli et al., 2008). In other words, a high σ indicates that the factor's elementary effect is strongly affected by the values of other factors and a low σ indicates that the factor's elementary effect is independent of the values of other factors.

2.2.2 Bayesian calibration statistical formulation

This study is based on Kennedy and O'Hagan's (2001) Bayesian approach for calibrating computer models. In this approach, the statistical formulation explicitly models uncertainty in the calibration parameters, the discrepancy between the simulator and the actual behavior of the building, as well as observation errors (Equation 2.4 below).

$$y(x) = \zeta(x) + \epsilon(x) = \eta(x, t^*) + \delta(x) + \epsilon(x) \tag{2.4}$$

where,
$y(x)$ is the observed value,
$\zeta(x)$ is the true behavior of the building,
$\eta(x, t)$ is the output from the building energy model given the input vector (x, t),
x is the observable input factors,
t is the unknown calibration parameters,
$\delta(x)$ is the discrepancy between the simulation output and the observed output,
$\epsilon(x)$ is the observation error.

In Equation 2.4, t^* is used to represent the true but unknown values of the calibration parameters t. This suggests that even in an ideal situation where $t = t^*$, the building energy model $\eta(x, t^*)$ would still be a biased representation of the true behavior of the building. Since the energy model is an approximation of reality, $\delta(x)$ is used to account for any model inadequacy that could be revealed by the discrepancy between the model predictions $\eta(x, t)$ and the true behavior of the building $\zeta(x)$. To learn about the calibration parameters t, m simulations with different values of t were run, where each simulation is run at the same observable input factors x. Maximin latin hypercube sampling (Stein, 1987) was used to generate different values of t for each simulation. This sampling approach tries to cover as much parameter space as possible by maximizing the minimum distance between design points.

Since the energy model could be computationally expensive to evaluate during the calibration process, a key element of this method is the use of a Gaussian process (GP) model as an emulator, mapping the input parameters of the model to the output of interest. To specify the GP model, a mean function $\mu(x, t)$ and a covariance function $Cov((x, t), (x', t'))$ needs to be defined. First, the GP model for the simulator $\eta(x,t)$ is defined with a mean function that returns the zero vector and a covariance function of the form (Higdon et al., 2004):

$$\begin{aligned}&Cov((x,t),(x',t')) = \\ &\frac{1}{\lambda_\eta} exp\bigg\{-\sum_{j=1}^{p}\beta_j^\eta|x_{ij}-x'_{ij}|^2 - \sum_{k=1}^{q}\beta_{p+k}^\eta|t_{ik}-t'_{ik}|^2\bigg\}\end{aligned} \tag{2.5}$$

where,

λ_η is the variance hyperparameter of this GP model,

$\beta_1^\eta, ...\beta_{p+q}^\eta$ are the correlation hyperparameters of this GP model,

p is the number of input factors x,

q is the number of calibration parameters t.

The discrepancy term $\delta(x)$ is also modeled using a GP model. This GP model is defined with a mean function that returns the zero vector and a covariance function of the form (Higdon et al., 2004):

$$Cov(x,x') = \frac{1}{\lambda_\delta} exp\bigg\{-\sum_{j=1}^{p}\beta_j^\delta|x_{ij}-x'_{ij}|^2\bigg\} \tag{2.6}$$

where,

λ_δ is the variance hyperparameter of this GP model,

$\beta_1^\delta, ..., \beta_p^\delta$ are the correlation hyperparameters of this GP model,

p is the number of input factors x.

Field data and simulation data is then combined using additive decomposition as proposed by Higdon et al. (2004). In other words, given that there are n observations and m simulation runs, the observed output is combined with the simulation output in a single $n + nm$ vector $z = \big[y(x_1), ..., y(x_n), \eta(x_1, t_1), ..., \eta(x_{nm}, t_{nm})\big]$. Note that because each simulation is run at the same observed input factors $x_1, ..., x_n$, running m simulations would produce a simulation dataset containing $nm = n \times m$ samples. The corresponding likelihood function is then given by:

$$\mathcal{L}(z \mid t, \beta^{\eta}, \lambda_{\eta}, \beta^{\delta}, \lambda_{\delta}, \lambda_{\epsilon}) \propto |\Sigma_z|^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2}(z-\mu)^T \Sigma_z^{-1}(z-\mu)\right\} \tag{2.7}$$

$$\Sigma_z = \Sigma_{\eta} + \begin{bmatrix} \Sigma_{\delta} + \Sigma_y & 0 \\ 0 & 0 \end{bmatrix} \tag{2.8}$$

where,
Σ_{η} is a $(n + nm) \times (n + nm)$ matrix computed based on Equation 2.5,
Σ_{δ} is a $n \times n$ matrix computed based on Equation 2.6,
Σ_y is the $n \times n$ covariance matrix used to account for observation errors and is given by I_n/λ_{ϵ}.

MCMC is then used to estimate the calibration parameters t, correlation hyperparameters (β^{η} and β^{δ}), and variance hyperparameters (λ_{η}, λ_{δ} and λ_{ϵ}).

2.2.3 Markov Chain Monte Carlo (MCMC) Algorithms

In any Bayesian calibration approach, MCMC is typically used to explore and generate samples from the posterior distribution. Its widespread use can be attributed to its ease of use in a wide variety of problems because it can readily handle large dimensions in the posterior. Two MCMC algorithms that have been used for the Bayesian calibration of building energy models are the random-walk Metropolis (RWM) (Metropolis et al., 1953) and the Gibbs sampling (Geman and Geman, 1984).

The RWM algorithm can be summarized as follows (Metropolis et al., 1953):

1. Arbitrarily select a valid initial starting point ψ^0.
2. Suppose $\psi^0, \psi^1, ..., \psi^i$ have been generated. Generate a candidate value ψ^{cand} from a symmetric proposal distribution given ψ^i.
3. Calculate the Metropolis acceptance probability α, i.e., the probability of transitioning to

the new candidate value

$$\alpha = min\left\{\frac{P(\psi^{cand}|y)}{P(\psi^{i}|y)}, 1\right\} \tag{2.9}$$

4. Accept and set ψ^{i+1} to the new candidate value with probability α or stay at the same point with probability $1-\alpha$.

$$\psi^{i+1} = \begin{cases} \psi^{cand} \text{ with probability } \alpha \\ \psi^{i} \text{ with probability } 1-\alpha \end{cases} \tag{2.10}$$

Another popular MCMC algorithm commonly used within the Bayesian calibration framework is Gibbs sampling (Geman and Geman, 1984). The algorithm proceeds by sampling each parameter from its conditional distribution while holding the remaining parameters fixed at their current values. To illustrate, suppose there are d parameters $\psi_1, \psi_2, ..., \psi_d$. At each iteration i, Gibbs sampling cycles through each parameter ψ_j, and samples it from its conditional distribution given the current value of the other parameters. This can be expressed by the following equation:

$$\psi_j^i \sim P(\psi_j|\psi_1^i, ..., \psi_{j-1}^i, \psi_{j+1}^{i-1}, ..., \psi_d^{i-1}) \tag{2.11}$$

where $\psi_1^i, ..., \psi_{j-1}^i, \psi_{j+1}^{i-1}, ..., \psi_d^{i-1}$ represents all other parameters at their current values except ψ_j.

Despite their simple implementation, one drawback of random walk Metropolis and Gibbs sampling algorithm is that they suffer from the "curse of dimensionality", i.e. in complicated problems, these algorithms may require an unacceptable large number of iterations (too time consuming) to converge to the posterior distribution.

2.3 Limitations of current bayesian calibration method

Although Bayesian calibration has been successfully applied to building energy models, challenges remain in its application to commonly used building energy simulation tools (section 1.2), daily or hourly calibration data, and a large number of calibration parameters. This is because the current Bayesian calibration method does not scale well to datasets with high dimensions and large sample sizes. This can be attributed to the following reasons:

- Gaussian process (GP) models have a runtime complexity of $\mathcal{O}(N^3)$ where N is the sample size of the data used to train the GP model. In other words, if twice as many samples were used, it would take eight times longer to train the GP model. To put into context, the sample size of hourly and daily data is approximately 730 and 30 times larger than monthly aggregated data collected over the same time period. Given n observations and m simulation runs, running each simulation at the same observable input factors $x_1, ..., x_n$ would produce a simulation dataset of size nm and a combined field and simulation dataset of size $n + nm$. Therefore, the number of samples used to train the GP model is $N = n + nm$. Correspondingly, it would take approximately $(730 + 730m)^3$ and $(30 + 30m)^3$ times longer to train the GP model with hourly and daily calibration data respectively. A common approach to overcome this would be to use heuristics to select representative subsets of the data. However, such an approach requires significant data analytics and expert knowledge, making the process difficult to replicate in an autonomous framework.
- The random-walk Metropolis (RWM) and the Gibbs sampling algorithms are routinely used for Bayesian calibration because of their simple implementation. However, these algorithms suffer from the "curse of dimensionality" and may take an unacceptable large number of iterations (longer runtimes) to achieve convergence for high-dimensional posterior distribution. In addition, fitting a large dataset to a GP model can be computationally challenging per iteration of RWM or Gibbs sampler. It is important to note that for the Bayesian calibration framework, the posterior distribution is typically high-dimensional and can easily involve more than 10 dimensions because of the GP correlation and variance hyperparameters (Equation 2.7).

Besides challenges associated with computation cost and the ability for the calibration process to be automated, current assessment of the calibrated model do not place enough emphasis on the evaluation of model performance. Currently, a model is considered calibrated when the coefficient of variation of the root mean squared error (CVRMSE) or the normalized mean bias error (NMBE) falls below the error criteria set by various guidelines or standards such as those listed in Table 1.1. To begin with, CVRMSE or NMBE is typically calculated using data that was used to calibrate the model. However, such a performance evaluation protocol is biased and may overfit the model, producing a calibrated model that is biased and performs poorly on unseen data. Moreover, CVRMSE and NMBE do not check for convergence of the iterative MCMC simulations. If a MCMC simulation has not proceeded long enough, the generated samples may be grossly unrepresentative of the target distributions (Gelman et al., 2014). Figure 2.3 shows an example of convergence issues. In the left plot, both chains appear stable and when looked at separately, convergence is seemingly achieved. However looking at both chains together shows a clear lack of convergence. The plot on the right shows that the two chains appear to cover a common distribution. However, to achieve convergence, each individual chain must achieve stationarity.

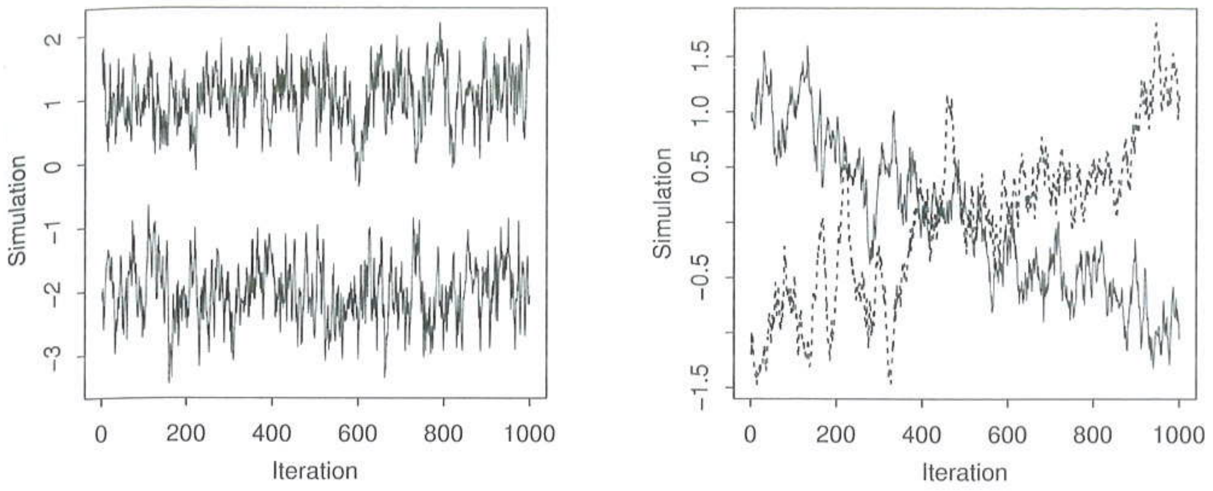

Figure 2.3: Examples of two challenges in assessing convergence of samples generated from MCMC methods, taken from Gelman et al. (2014)

2.4 Additions to current Bayesian calibration method

2.4.1 Overview

2 strategies were employed to overcome the computation challenge of applying Bayesian calibration to high-dimensional datasets and large samples sizes. As mentioned in Section 2.3, large sample sizes occur when calibrating against daily or hourly data. High dimensionality typically occurs from fitting the calibration parameters and the hyperparameters that define a GP emulator. The 2 approaches employed include:

1. Reducing the sample size by sampling a representative subset of the data from the entire dataset. The sampled subset is then be used for the calibration of the building energy model instead of the full dataset.

2. Using a more effective MCMC algorithm, the No-U-Turn Sampler (NUTS) (Hoffman and Gelman, 2014), which is an extension of Hamiltonian Monte Carlo (HMC) (Gelman et al., 2014). NUTS requires no manual tuning, is more efficient and converges more quickly in high-dimensional problems.

In addition, the following were also included to improve the rigor in assessing the performance of the calibrated model:

1. Assess accuracy (agreement between measured and predicted) using a hold-out test dataset that was not used in the calibration process.

2. Assess convergence of iterative MCMC simulations (i.e., multiple MCMC chains have converged to a common stationary distribution).

2.4.2 Sampling a representative subset

The traditional design of experiments involves specifying inputs through which the corresponding output can then be observed. However, unlike the traditional design of experiments, energy modelers are not provided with the flexibility to configure the input factors x that may affect the observed output y. Instead, energy modelers are typically provided with historical data containing measured values of y and x for the calibration of the energy model. Therefore, the design space is the set of x values at which the building has been operated and the experimental design corresponding to the field observations is a dataset $D^F = \begin{bmatrix} y & x \end{bmatrix}$ that is made up the observed values $y_1, ..., y_n \in \mathbb{R}$ and the corresponding observed input factors $x \in \mathbb{R}^{n \times p}$.

To learn about the calibration parameters t, m simulations are run at different combinations of (x, t). Therefore, the corresponding design of experiments is a simulation dataset $D^S = \begin{bmatrix} \eta & x & t \end{bmatrix}$ containing the simulator output $\eta_1, ..., \eta_{nm} \in \mathbb{R}$ and the corresponding input factors $x \in \mathbb{R}^{nm \times p}$ and the calibration parameters $t \in \mathbb{R}^{nm \times q}$. [1]

Having access to massive volumes of data does not imply that the calibration algorithm should be applied to the entire dataset. Since buildings are typically operated the same way throughout the year, there is significant redundancy in building data. This implies that a small sample of the large dataset would provide sufficient accuracy with significantly lower computation cost. Traditionally, the approach has been to manually select representative parts of the data (e.g., one week of summer and winter data) for the calibration and the analysis. However, such a process typically requires expert knowledge and is subjective, making it harder to be replicated and hence automated. Therefore, the proposed approach uses random samples from the dataset. However, determining the correct sample size is often not intuitive. To overcome this, a statistical approach that is based on information theory is used. Such an approach is more intuitive, repeatable and less prone to lost of available information. The approach uses Kullback-Leibler divergence (Kullback and Leibler, 1951) to measured the "distance" or "divergence" of the selected subset from the whole dataset. More specifically, it uses a metric known as sample quality Q (Gu et al., 2001). Suppose there is a dataset D with R attributes and the sample quality of its subset D_{sub} needs to be calculated. The sample quality of D_{sub} is used to measure how similar

[1] Since each simulation is run at the same observed input factors $x \in \mathbb{R}^{n \times p}$, running m simulations would result in a simulation dataset D^S containing $n \times m$ samples. As mentioned in Chapter 2.3, the resulting sample size $N = n + nm$ that the GP model is trained on can increase very quickly as more observations n is used. Together with its $\mathcal{O}(N^3)$ runtime complexity, this can make the calibration process computationally prohibitive with hourly or daily data.

D_{sub} is to D and is given by Equation 2.12.

$$Q(D_{sub}) = \exp(-J) \tag{2.12}$$

$$J = \frac{1}{R}\sum_{i=1}^{R} J_i(D_{sub}, D) \tag{2.13}$$

$$J_i(D_{sub}, D) = \sum_{j=1}^{c} \left(P_j^{D_{sub}} - P_j^{D}\right) \log \frac{P_j^{D_{sub}}}{P_j^{D}} \tag{2.14}$$

where,

D is the entire dataset,

D_{sub} is a subset of D,

J is the averaged information divergence,

$J_i(D_{sub}, D)$ is the Kullback-Leibler divergence (of the i^{th} attribute) between D_{sub} and D,

$P_j^{D_{sub}}$ is the probability of occurrence of the j^{th} value in D_{sub},

P_j^{D} is the probability of occurrence of the j^{th} value in D,

Q is the sample quality,

R is the number of attributes.

By definition, J is always larger than 0 (Kullback and Leibler, 1951; Gu et al., 2001). Therefore, $0 < Q \leq 1$ and $Q = 1$ indicates no divergence between the subset D_{sub} and the entire dataset D. The larger the information divergence J, the lower the sample quality Q and vice versa. In calculating $J_i(D_{sub}, D)$, each continuous attribute is first discretized. However, with the sampled dataset containing significantly less samples than the entire dataset, there may be zero entries in the sampled data D_{sub}. Kullback-Leibler divergence is only defined when all entries are non-zero. Therefore, a Bayesian prior is used to smooth the distribution. In this study, a Dirchlet prior was used to smooth the distribution. This results in a posterior distribution that is also Dirchlet (Equation 2.15) (Hausser and Strimmer, 2009). Table 2.1 shows the common choices for the flattening constant a_j when using a Dirichlet prior to smooth the sampled data.

Table 2.1: Common choices for a Dirichlet prior that can be used to smooth the sampled data (Equation 2.15), taken from Hausser and Strimmer (2009).

a_j	Cell frequency prior
0	no prior
$1/2$	(Jeffreys, 1946)
1	Bayes-Laplace uniform prior
$1/p$	Perks prior (Perks, 1947)
$\sqrt{n}/p$	minimax prior (Trybula, 1958)

$$P_j = \frac{a_j + b_j}{\sum_{j=1}^{c} a_j + \sum_{j=1}^{c} b_j} \tag{2.15}$$

where,

a_j is a flattening constant and acts as a pseudo-count,

b_j denotes the observed counts for each bin,

c is the number of bins or categories.

Then, using this definition of sample quality Q, a sampling schedule S is used to determine the sample size of the subset D_{sub}. In this study, a geometric sampling schedule is used (Equation 2.16) (Provost et al., 1999).

$$S = \{s_0, A \cdot s_0, A^2 \cdot s_0, A^3 \cdot s_0, ...\} \tag{2.16}$$

where,

S is the sampling schedule,

$s_0 > 0$ is the starting sample size,

$A > 1$ is the increment ratio.

Subsequently, a learning curve is used to depict the relationship between sample size and sample quality (Figure 2.4). The horizontal axis represents the number of random samples and the vertical axis represents the sample quality. From the case studies (Chapter 3), these curves typically increase very rapidly at the start and then gradually levels out to a plateau. This indicates that a small sample size would be sufficiently representative of the entire dataset and that when n_i is sufficiently large, adding more samples adds small accuracy improvements. Using the learning curve, an appropriate sample size n_{sub} is selected taking into consideration both sample quality and computation cost of the calibration algorithm.

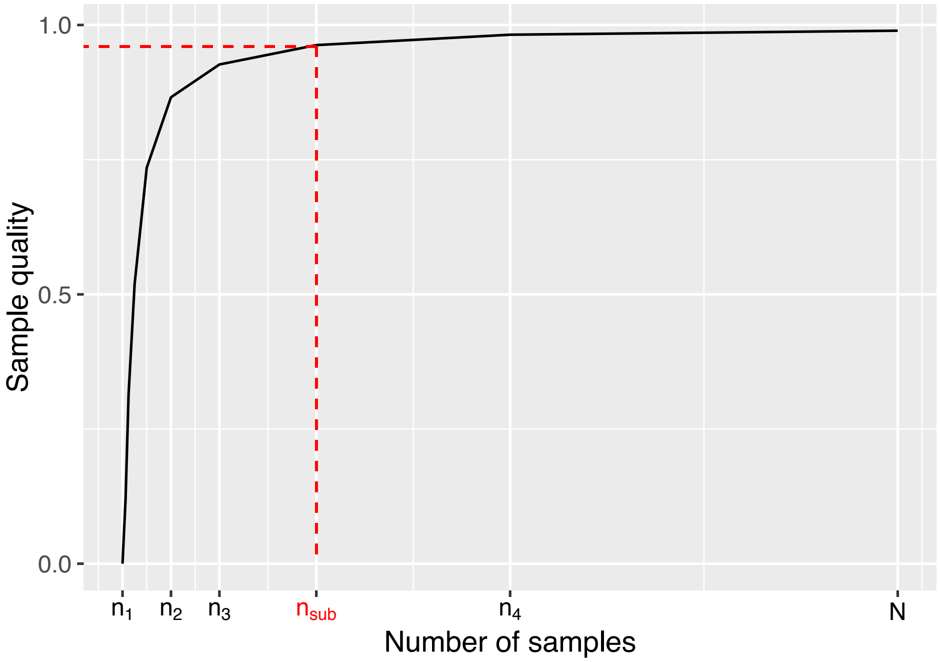

Figure 2.4: Curve depicting relationship between sample quality Q and sample size n

2.4.3 No-U-Turn Sampler (NUTS) MCMC algorithm

Instead of the commonly used random-walk Metropolis (RWM) (Metropolis et al., 1953) or Gibbs sampler (Geman and Geman, 1984), the No-U-Turn Sampler (NUTS) (Hoffman and Gelman, 2014), an extension of Hamiltonian Monte Carlo (HMC) algorithm is used to explore the posterior distribution during the MCMC sampling. HMC is a variant of the MCMC algorithm that avoids the random walk behavior and correlation between successive sampled states that plague many MCMC methods, allowing faster convergence to high-dimensional posterior distributions (Duane et al., 1987; Neal, 1993, 2011; Gelman et al., 2014). To avoid the random walk behavior, HMC borrows a concept from Hamiltonian dynamics, which describes an object's motion in terms of its position ψ and its momentum τ. In the context of applying HMC to Bayesian calibration as described in Chapter 2.2.2, the location variables ψ corresponds to the parameters of the posterior distribution, i.e., the calibration parameters t and the hyperparameters $\beta_1^\eta, ..., \beta_{p+q}^\eta$, $\beta_1^\delta, ..., \beta_p^\delta$, λ_η, λ_δ and λ_ϵ that define the GP emulator. To make the algorithm move faster in the parameter space, HMC introduces an auxiliary momentum variable τ_i for each variable ψ_i. The goal is to use Hamiltonian dynamics to find a more efficient proposal or jumping distribution. A basic implementation of HMC is shown in Listing 2.1.

As shown in Listing 2.1, the main part of HMC is the simultaneous update of (ψ,τ). This update is carried out during each of the N iterations and involves L leapfrog steps. In other words, the function `Leapfrog(`ψ`,`τ`,`ϵ`)` is called L times during each iteration. A useful insight behind the leapfrog algorithm is that, when the values of ψ is at a flat portion of the posterior and the log-posterior density is 0, the momentum τ will remain constant and the algorithm moves within the parameter space at constant velocity (Gelman et al., 2014). As the algorithm moves towards a region of lower probability density, the log-posterior density is negative and the algorithm slows down. On the contrary, if the algorithm moves towards a region of higher probability density, the log-posterior density is positive and the algorithm moves through the parameter space more rapidly. A downside is the longer computation time for each iteration of HMC because its gradient needs to be computed during each of the L leapfrog steps, which according to Hoffman and

Gelman (2014) is the most computationally intensive part of the algorithm.

Listing 2.1: Hamiltonian Monte Carlo (HMC), adapted from Hoffman and Gelman (2014)

```
Given $\psi^0$, $\epsilon$, $L$, $N$
for i = 1 to N
    Sample $\tau^0 \sim \mathcal{N}(0, I)$
    Set $\psi^i = \psi^{i-1}$, $\tilde{\psi} = \psi^{i-1}$, $\tilde{\tau} = \tau^0$
    for j = 1 to L
        $\tilde{\psi}$, $\tilde{\tau}$ = Leapfrog($\tilde{\psi}$, $\tilde{\tau}$, $\epsilon$)
    end
    $\alpha = min\left\{1, \frac{exp\{\mathcal{L}(\tilde{\psi}) - \frac{1}{2}\tilde{\tau}.\tilde{\tau}\}}{exp\{\mathcal{L}(\psi^{i-1}) - \frac{1}{2}\tau^0.\tau^0\}}\right\}$
    Set $\tau^i = \tilde{\tau}$, $\psi^i = -\tilde{\psi}$ with probability $\alpha$
end

function Leapfrog($\psi$, $\tau$, $\epsilon$)
    $\tilde{\tau} = \tau + (\epsilon/2)\nabla_q\mathcal{L}(\psi)$
    $\tilde{\psi} = \psi + \epsilon\tilde{\tau}$
    $\tilde{\tau} = \tilde{\tau} + (\epsilon/2)\nabla_\psi\mathcal{L}(\psi)$
    return $\tilde{\psi}$, $\tilde{\tau}$
```

From the pseudocode above, it can be seen that to run HMC, users need to provide values for 1) the scaling factor ϵ (leapfrog step size), and 2) the number of leapfrog steps per iteration L. Therefore, like most MCMC methods, the use of HMC requires time consuming initial runs to tune both ϵ and L. Poor choices of either parameter can result in an ineffective implementation of HMC (Hoffman and Gelman, 2014). To mitigate the challenges involved in tuning L, NUTS uses a recursive algorithm to automatically select the number of leapfrog steps L per iteration. It also automatically determines a value for the scaling factor ϵ through a dual averaging scheme, thus making it possible to run NUTS without requiring any user intervention. Details on its implementation can be found in Hoffman and Gelman (2014).

To summarize, NUTS is used for the following reasons:

- NUTS, an extension of HMC is more efficient in high-dimensional problems because the algorithm is guided by the log-posterior gradient, suppressing random-walk behavior that exists in most commonly used MCMC methods such as RWM and Gibbs sampling. This makes it suitable for the Bayesian calibration of building energy models, which usually involves exploring high-dimensional posterior distributions comprising of the calibration parameters t, correlation hyperparameters (β^{η} and β^{δ}), and variance hyperparameters (λ_{η}, λ_{δ} and λ_{ϵ}).
- NUTS automatically tunes the HMC parameters ϵ and L. Therefore NUTS can be used without any time consuming initial runs to tune the hyperparameters of HMC (i.e., the number of leapfrog steps L and the leapfrog stepsize ϵ).

Despite these advantages, neither NUTS or HMC has ever been utilized for the Bayesian calibration of building energy models. For this study, Bayesian inference with NUTS was implemented in R version 3.2.3 (R Core Team, 2015) with the package rstan (Stan Development Team, 2016). Appendix B provides details for implementing NUTS in a Bayesian calibration framework.

2.4.4 Model evaluation

Two categories of performance metrics were used to assess the performance of the calibrated model and they include: 1) assessing accuracy on a test dataset using standard metrics of agreement between model predictions and observed values; and 2) assessing convergence of multiple MCMC chains to a common stationary distribution

The coefficient of variation of the root mean squared error (CVRMSE) and the normalized mean bias error (NMBE) were used to assess the accuracy of the calibrated model. To prevent bias in the evaluation process, CVRMSE and NMBE is calculated on a hold-out test dataset that was not used in the calibration process. CVRMSE (Equation 1.5) provides a measure of how well the simulated data fits the actual values while NMBE (Equation 1.3) serves as a good indicator of overall bias in the simulated data, providing an indication as to whether the predicted values tend to overestimate or underestimate actual values. For hourly calibration data, a CVRMSE below 30% and NMBE below 10% is considered acceptable according to ASHRAE Guideline 14 (ASHRAE, 2002). With monthly calibration data, the thresholds are stricter at 15% (CVRMSE) and 5% (NMBE) respectively. See Table 1.1 for the acceptable limits of CVRMSE and NMBE set by different standards and guidelines.

Trace plots of multiple MCMC chains and Gelman-Rubin statistics ($\hat{R}$) (Gelman et al., 2014) were used to assess convergence. Looking at the trace plots allows us to determine if the chains are well-mixed and if different chains have converged to a common stationary distribution. Well-mixed chains indicate faster convergence to the stationary distribution and therefore faster computation. $\hat{R}$ is the ratio of between-chain variance to within-chain variance and is calculated as follows:

$$\hat{R} = \sqrt{\frac{v\hat{a}r(\psi|y)}{W}}, \quad \text{where} \quad v\hat{a}r(\psi|y) = \frac{N-1}{N}W + \frac{1}{N}B \tag{2.17}$$

$$B = \frac{N}{M-1}\sum_{j=1}^{M}(\overline{\psi}_{.j} - \overline{\psi}_{..})^2, \quad \text{where} \quad \overline{\psi}_{.j} = \frac{1}{N}\sum_{i=1}^{N}\psi_{ij}, \quad \overline{\psi}_{..} = \frac{1}{M}\sum_{j=1}^{M}\overline{\psi}_{.j} \tag{2.18}$$

$$W = \frac{1}{M}\sum_{j=1}^{M}\left[\frac{1}{N-1}\sum_{i=1}^{N}(\psi_{ij} - \overline{\psi}_{.j})^2\right] \tag{2.19}$$

where,
B is the between chain vairance,
W is the within chain variance,
M is the number of chains,
N is the number of iterations per chain,
ψ is the estimate.

$\hat{R}$ is based on the concept that if multiple chains have converged, there should be little variability between and within the chains. For convergence, $\hat{R}$ should be 1 ± 0.1. It is important to check for convergence because if the MCMC algorithm has not proceeded long enough, the generated samples may be grossly unrepresentative of the actual posterior distributions (Gelman et al., 2014). Therefore, assessing the convergence of all variables of the posterior distribution (t, β^{η}, β^{δ}, λ_{η}, λ_{δ} and λ_{ϵ}) is recommended. This would make the assessment of the calibrated model more rigorous and provide greater confidence that the generated samples are representative of the target posterior distribution.

Chapter 3

Case Studies

In this chapter, the effectiveness of the proposed method is evaluated using three different case studies. To demonstrate its flexibility, the case studies were selected to range from a single chiller component to a whole building energy model. The three case studies include: 1) a TRNSYS model of a water-cooled chiller component; 2) an EnergyPlus model of the cooling system of a ten story office building; and 3) a whole building EnergyPlus model of a mixed-use building.

To simplify the specification of the prior probability distributions, the outputs were standardized to have zero mean and unit variance and the inputs scaled to the range [0,1] (See Listing B.2 in Appendix B). Standardization also helps obtain better estimates of GP hyperparameters and ease maximum likelihood estimation. Priors for the following calibration parameters and GP hyperparameters were then specified as:

- Calibration paramaters $t_1, ..., t_q \sim \mathcal{U}(min = 0, max = 1)$. This is consistent with the earlier normalization that puts them in the range between 0 and 1.
- Correlation hyperparameters $\beta_1^\eta, ..., \beta_{p+q}^\eta$: These correlation hyperparameters were reparameterized using $\rho_i^\eta = \exp(-\beta_i^\eta/4)$, $i = 1, ..., p+q$ so that $0 < \rho_i^\eta < 1$ since $\beta_i^\eta > 0$ (Higdon et al., 2008; Guillas et al., 2009). Independent $Beta(a = 1, b = 0.5)$ priors were then specified for each ρ_i^η. Setting $a = 1$ and $0 < b < 1$ places most of the prior support near 1, indicating an expectation that only a subset of the inputs have an effect on the sim-

ulation output. Smaller values of b indicate an expectation that the output depends on an even smaller number of inputs.

- Correlation hyperparameters $\beta_1^\delta, ..., \beta_p^\delta$: Similarly, these correlation hyperparameters were reparameterized using $\rho_i^\eta = \exp(-\beta_i^\eta/4)$, $i = 1, ..., p$. A more conservative independent $Beta(a = 1, b = 0.4)$ prior was assigned to each ρ_i^η because an even smaller subset of the inputs was expected to have an effect on the discrepancy term $\delta(x)$.
- Variance hyperparameter λ_η: A $Gamma(a = 5, b = 5)$ prior was assigned to this hyperparameter. Here, a represents the shape and b is the rate. Since the outputs were standardized to have unit variance, λ_η is expected to be close to one. Therefore, a Gamma prior with $a = b = 5$ is suitable. In addition, a $Gamma(a = 5, b = 5)$ prior helps to stabilize the correlation hyperparameters (Higdon et al., 2008; Kern, 2000).
- Variance hyperparameters λ_δ and λ_ϵ: $Gamma(a = 1, b = 0.0001)$ priors were specified for both λ_δ and λ_ϵ. This results in a prior that is quite uninformative. Therefore, if the data is uninformative about these parameters, the posterior would be large, resulting in a small discrepancy and observation error respectively.

3.1 Case study 1 - mixed-use building in a college in Singapore

3.1.1 Case study and model description

In case study 1, the proposed method (Figure 2.1) was applied to a water-cooled chiller of a mixed-use building located in a college in Singapore. The chiller provides cooling to indoor spaces that include classrooms, offices, auditoriums, and mixed-use spaces. Singapore is located near the equator and has a tropical climate. Therefore the predominant energy consumption in buildings is for dehumidification and space cooling.

The water-cooled chiller was modeled using the TRNSYS “Type 666: Water Cooled Chiller” component. Due to its modular structure, TRNSYS is particularly suited for the modeling of a single chiller component. We use the following notations to define this chiller component.

The “Type 666: Water Cooled Chiller” component relies on catalog data to determine a chiller’s performance at different operating conditions (Thornton et al., 2014). Catalog data is provided in the form of lookup tables that include: 1) the chiller capacity ratio $[-]$ at varying chilled water setpoint temperatures $T_{chw,set}$ $[°C]$ and entering chilled water temperature $T_{chw,in}$ $[°C]$; 2) the chiller Coefficient of Performance (COP) ratio $[-]$ at varying chilled water setpoint temperatures $T_{chw,set}$ $[°C]$ and entering chilled water temperature $T_{chw,in}$ $[°C]$; and 3) the chiller fraction of full load power $FFLP$ at varying part load ratios PLR. Figure 3.1 shows a simplified representation of the cooling process.

Measurements used for this study include the chiller energy consumption, the entering chilled water temperature $T_{chw,in}$, the chilled water mass flow rate $\dot{m}_{chw}$ and the entering condenser water temperature $T_{cw,in}$. Data for the scheduled chilled water setpoint $T_{chw,set}$ was also obtained from the building management system. High accuracy temperature sensors ($\pm 0.03°C$) were used for the chilled and condenser water temperature measurements. For greater precision, thermowells were used to allow direct contact of the sensors with the water in the pipes. Chilled water flow rates were measured using full bore electromagnetic flow meters with an accuracy of $\pm 0.5\%$.

At each time step, the simulation takes as inputs, the chilled water set point temperature $T_{chw,set}$ and the entering condenser water temperature $T_{cw,in}$ to determine the current capacity ratio and COP ratio from the lookup tables. The chiller capacity and nominal COP can then be calculated using the rated capacity and the rated COP which we take as calibration parameters t (Equations 3.1 and 3.4).

$$COP_{nom} = COP_{rated} \cdot COP_{ratio} \tag{3.1}$$

where,

COP_{nom} is the chiller nominal COP at current conditions $[-]$,

COP_{rated} is the chiller rated COP at current conditions $[-]$,

COP_{ratio} is the chiller COP at current conditions divided by the rated COP $[-]$.

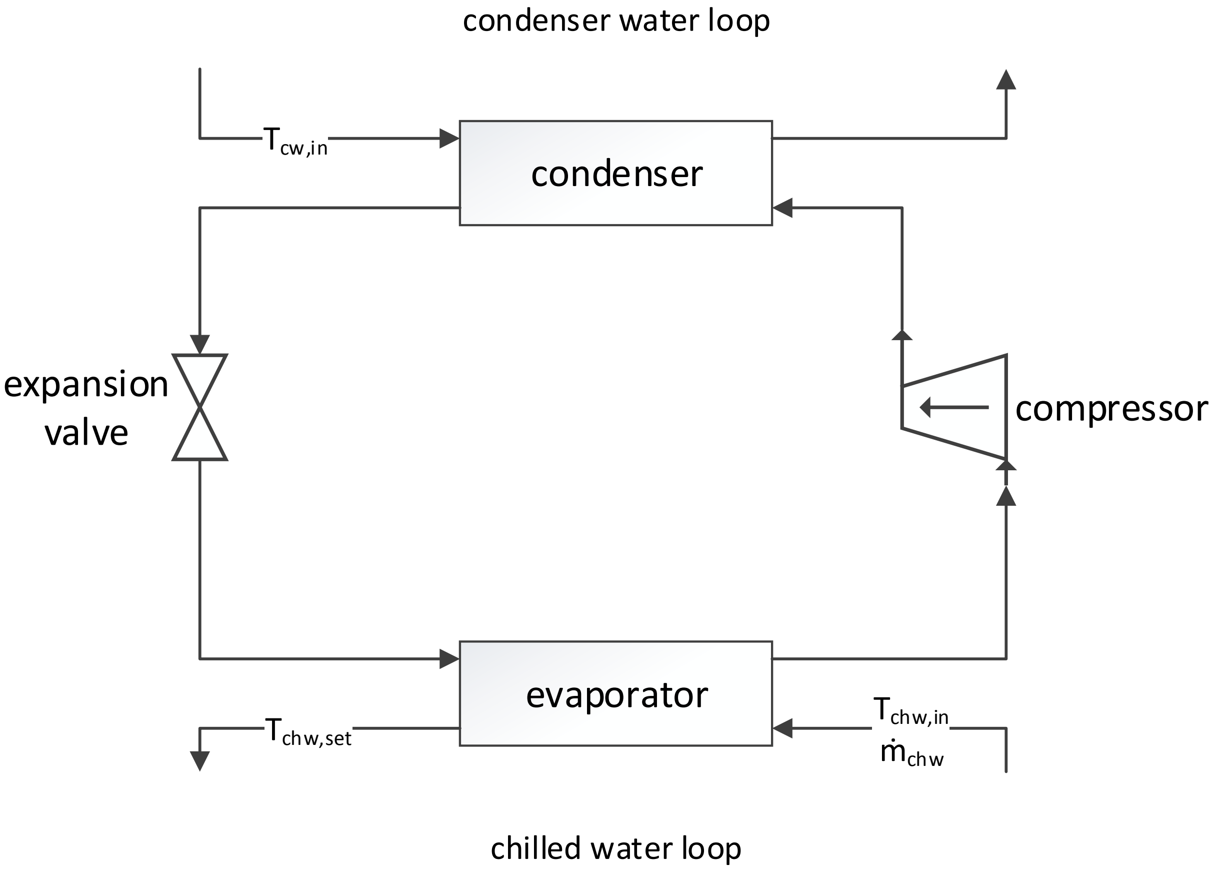

Figure 3.1: Case study 1: schematic showing cooling process of a water cooled chiller. Inputs at each time step include 1) entering chilled water temperature $T_{chw,in}$ [$°C$]; 2) chilled water mass flow rate $\dot{m}_{chw}$ [kg/h]; 3) chilled water setpoint temperature $T_{chw,set}$ [$°C$]; and 4) entering condenser water temperature $T_{cw,in}$ [$°C$].

$$Capacity = Capacity_{rated} \cdot Capacity_{ratio} \tag{3.2}$$

where,

$Capacity$ is the chiller capacity at current conditions [kJ/h],

$Capacity_{rated}$ is the chiller rated capacity [kJ/h],

$Capacity_{ratio}$ is the chiller capacity at current conditions divided by the rated capacity [$-$].

Then, the chiller load Q_{load} and the part load ratio PLR can be calculated using Equations 3.3 and 3.4 respectively.

$$\dot{Q}_{load} = \dot{m}_{chw} \cdot Cp_{chw} \cdot (T_{chw,in} - T_{chw,set}) \tag{3.3}$$

$$PLR = max(1, \frac{\dot{Q}_{load}}{Capacity}) \tag{3.4}$$

where,

$\dot{Q}_{load}$ is the current load on the chiller [kJ/h],

m_{chw} is the flow rate of fluid entering the chilled water stream [kg/hr],

Cp_{chw} is the specific heat of fluid entering the cooling water stream [$kJ/kg.K$],

$T_{chw,in}$ is the temperature of water entering the chilled water stream [$°C$],

$T_{chw,set}$ is the desired outlet temperature of water in the chilled water stream [$°C$],

PLR is the chiller Part Load Ratio (the ratio of the current load to the rated load) [$-$].

Although specific heat capacity Cp_{chw} is a parameter in the model, it is given a value of 4.19 $kJ/kg.K$ and not modeled as a random variable. This is because it is a well known property of water. Using catalogue data provided through a lookup table, the calculated PLR (Equation 3.4) is used to determine the fraction of full load power $FFLP$. Chiller power at a given time step can then be calculated using Equation 3.5.

$$P = \frac{Capacity}{COP_{nom}} FFLP \tag{3.5}$$

where,

P is the power drawn by the chiller at current conditions [kJ/h],

$Capacity$ is the chiller capacity at current conditions [kJ/h],

COP_{nom} is the chiller nominal COP at current conditions [$-$],

$FFLP$ is the fraction of full load power [$-$].

The energy consumption of the chiller is then calculated as power over time. In this example, sensitivity analysis was not carried out because only two parameters were identified as uncertain. Therefore, the inputs and output used for the calibration of the chiller model can be summarized as:

- Observed output $y(x)$: Measured energy consumption of water cooled chiller $[kJ]$.
- Simulation output $\eta(x, t)$: Predicted energy consumption of water cooled chiller $[kJ]$.
- Observed inputs x:
  (a) x_1: entering chilled water temperature $T_{chw,in}$ $[°C]$.
  (b) x_2: chilled water mass flow rate $\dot{m}_{chw}$ $[kg/h]$.
  (c) x_3: chilled water setpoint temperature $T_{chw,set}$ $[°C]$.
  (d) x_4: entering condenser water temperature $T_{cw,in}$ $[°C]$.
- Calibration parameters t:
  (a) t_1: Chiller rated capacity $[kJ/h]$.
  (b) t_2: Chiller rated COP $[-]$.

3.1.2 Bayesian calibration

Bayesian calibration as described in Chapter 2.2.2 was used for the calibration. Data collection took place between January 1, 2016 and April 30, 2016. After removing erroneous data and all instances for which the outcome or any of the inputs are missing, the dataset contained $n = 1130$ samples. Of the data collected, 30% (339 samples) were used as a test hold-out dataset and the remaining 70% (791 samples) were used for calibrating the model. Hourly data was used for the calibration.

To reduce computation cost, 2 strategies were employed and they include using information theory to reduce the number of samples (Chapter 2.4.2) and using NUTS (a more efficient MCMC algorithm) to explore the posterior distribution (Chapter 2.4.3). The observed input factors x is four dimensional with components corresponding to: 1) x_1: entering chilled water temperature $T_{chw,in}$; 2) x_2: chilled water mass flow rate $\dot{m}_{chw}$; 3) x_3: chilled water setpoint temperature $T_{chw,set}$; and 4) x_4: entering condenser water temperature $T_{cw,in}$. The observed output y is the measured energy consumption of the chiller. Therefore the experimental design corresponding to the field observations is a dataset $D^F = \begin{bmatrix} y & x_1 & x_2 & x_3 & x_4 \end{bmatrix}$ where $D^F \in \mathbb{R}^{791 \times 5}$ (there are 791 samples because only 70% of the data was used for the calibration).

To learn about the calibration parameters t, $m = 200$ TRNSYS simulations were run at different combinations of (x, t). Maximin LHS (Stein, 1987) was used to generate values for the calibration parameters, chiller rated capacity t_1 and chiller rated COP t_2. To be conservative, an upper and lower bound that is ± 20% of the initial values was assigned. Running each simulation at the same input factors x produced 791 × 200 $= 158200$ samples. Therefore, the experimental design for the simulations is a dataset $D^S = \begin{bmatrix} \eta(x,t) & x_1 & x_2 & x_3 & x_4 & t_1 & t_2 \end{bmatrix}$ where $D^S \in \mathbb{R}^{158200 \times 7}$.

To sample a representative subset D^F_{sub} from the field dataset D^F and a representative subset D^S_{sub} from the simulation dataset D^S, random samples of varying sizes were generated and their corresponding sample quality computed (Chapter 2.4.2). Figure 3.2 shows the relationship between sample size and sample quality. Taking into consideration both sample quality and computation cost, a sample size of 160 for D^F_{sub} and 640 for D^S_{sub} was used. Then, D^F_{sub} and D^S_{sub} was used for the calibration of the building energy model instead of the entire dataset D^F and D^S.

Using the formulation by Higdon et al. (2004), D^F_{sub} and D^S_{sub} were combined in a Gaussian process (GP) model. Following Chapter 2.2.2, a GP model for $\eta(.,.)$ was specified using a mean function that is set to return the zero vector and a covariance function of the form:

$$\begin{aligned} &Cov((x,t),(x',t')) = \\ &\frac{1}{\lambda_\eta} exp\Bigg\{ -\sum_{j=1}^{4} \beta_j^\eta |x_{ij} - x'_{ij}|^2 - \sum_{k=1}^{2} \beta_{p+k}^\eta |t_{ik} - t'_{ik}|^2 \Bigg\} \end{aligned} \tag{3.6}$$

The GP model for the discrepancy term $\delta(x)$ was also specified with a mean function that is set to return the zero vector and a covariance function of the form:

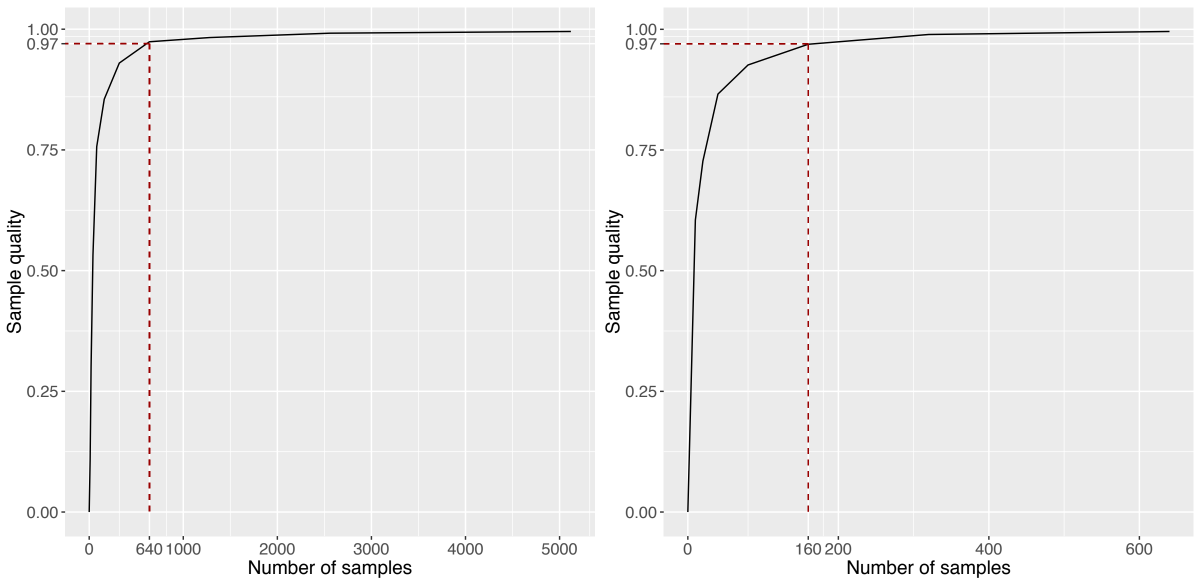

Figure 3.2: Case study 1: number of samples against sample quality for different subsets of simulation data D^S (left plot) and field data D^F (right plot). 640 random samples from simulation data D^S and 160 random samples from field data D^F is sufficient to represent the whole dataset.

$$Cov(x, x') = \frac{1}{\lambda_\delta} exp\left\{ -\sum_{j=1}^{4} \beta_k^\delta |x_{ij} - x'_{ij}|^2 \right\} \tag{3.7}$$

Given that there are 2 calibration parameters and 4 input factors, parameters of the posterior that need to be estimated include: 1) the calibration parameters t_1, t_2; 2) the correlation hyperparameters of the GP model $\beta_1^\eta, ..., \beta_6^\eta, \beta_1^\delta, ..., \beta_4^\delta$; and 3) the variance hyperparameters of the GP model λ_η, λ_δ and λ_ϵ. NUTS, an extension of HMC was used to explore the posterior distributions. Four independent chains of 500 iterations per chain were run. To be conservative, the first 250 iterations (50%) values were discarded as warmup/burn-in to reduce the influence of starting values.

Table 3.1 shows the prior probability distribution that was assigned to the calibration parameters as well as summarizes their posterior estimates. From Table 3.1, it can be seen that the chiller's rated capacity t_1 has a posterior 95% confidence interval of (2166341,2330755). This is narrower than the initial prior ($t_1 \sim \mathcal{U}(1795434, 2688586)$) that was assigned. The same observation is made for the chiller's COP t_2 where the posterior 95% confidence interval (7.1,8.3) is narrower than the initial prior ($t_2 \sim \mathcal{U}(5.6, 8.4)$) that was assigned. Figure 3.3 shows that the posterior estimates for t_1 and t_2 are normally distributed with mean 2166341 kJ/h and 7.7 respectively. Also, the 2 dimensional histogram in Figure 3.3 illustrates that t_1 and t_2 are positively correlated. This is not surprising since the chiller's power consumption is directly proportional to the ratio of its capacity and its COP (Equation 3.5).

Table 3.1: Case study 1: Prior distribution and summary statistics for posterior distribution of calibration parameters.

Parameter [units]	**Prior**	**Posterior**			
		mean	**2.5 Percentile**	**50 Percentile**	**97.5 Percentile**
t_1 $[kJ/h]$	$\mathcal{U}(1795434, 2688586)$	2166341	2009712	2168428	2330755
t_2 $[-]$	$\mathcal{U}(5.6, 8.4)$	7.7	7.1	7.6	8.3

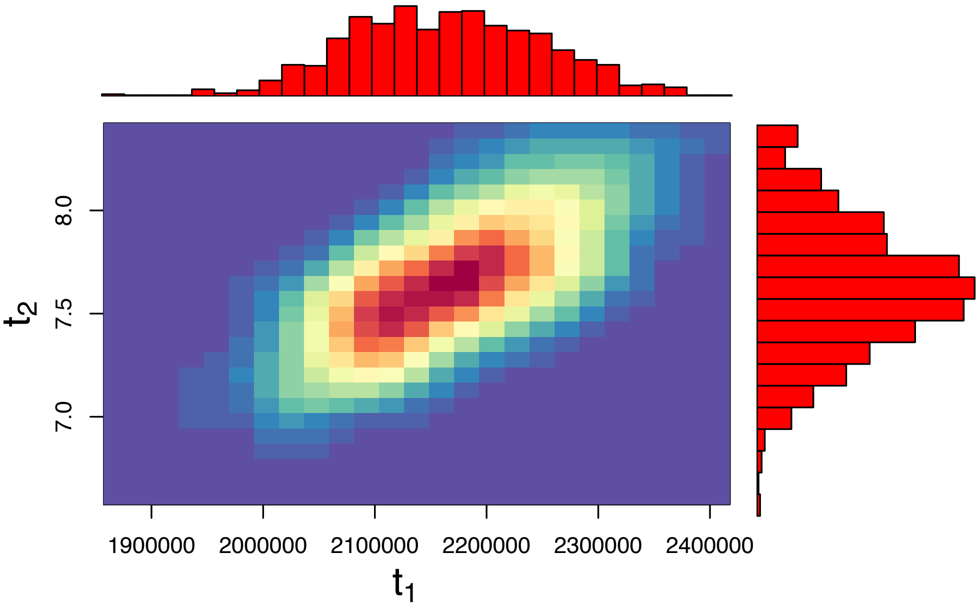

Figure 3.3: Case study 1: posterior distribution (excluding warmup) of calibration parameters: chiller rated capacity t_1 $[kJ/h]$, chiller rated COP t_2 $[-]$.

3.1.3 Model evaluation

Given the focus on prediction accuracy, it is important to evaluate the prediction accuracy of the calibrated model. To assess prediction accuracy, predictions by the calibrated model is compared to measurements in the hold-out test dataset. Figure 3.4 shows the 339 hold-out samples and how they compare with the 95% confidence interval of the predictions. These are predictions by the calibrated model at input settings that were not part of the data used to calibrate or train the model. It can be observed that there is a good match between the calibrated predictions and the field observations, with most measurements being within the 95% confidence interval of the predictions.

Table 3.2: Case study 1: CVRMSE and NMBE computed with posterior mean estimates with and without the discrepancy term $\delta(x)$.

Statistical Formulation	**CVRMSE (%)**	**NMBE (%)**
$\eta(x,t) + \epsilon(x)$	12.0	-0.8
$\eta(x,t) + \delta(x) + \epsilon(x)$	9.4	-0.7

Over a total of 339 hold-out samples, CVRMSE (9.4%) and NMBE (-0.7%) with the mean posterior predictions is also within the acceptable thresholds for hourly calibration data set by various standards and guidelines (See Table 1.1). This is also illustrated in Figure 3.5, which shows a histogram of residuals for the calibrated predictions standardized by the observed values and their standard deviation. The figure shows that most residuals are within $\pm 1\sigma_y$. However, there are a few predictions that lies further to the right of zero, suggesting that they overestimate their corresponding observed values. These overestimation occur for measurement points that have high entering chilled water temperature ($T_{chw,in}$) and low chilled water temperature setpoint ($T_{chw,set}$) (Figure 3.6). When the inlet chilled water temperature is high and the chilled water setpoint temperature is low, a high chiller energy consumption is expected. Figure 3.6 illustrates that this is true for most measurement points with the exception of the outliers (i.e., data that were overestimated by the calibrated model). A possible explanation is that the chiller is not operating as expected, and the chilled water supply temperature is significantly higher than the setpoint. Therefore, it might be useful to introduce operation of the chiller as a model parameter to determine if it is indeed due to the chiller's operation. Nonetheless, such operational data is currently unavailable and thus cannot be tested in the current study.

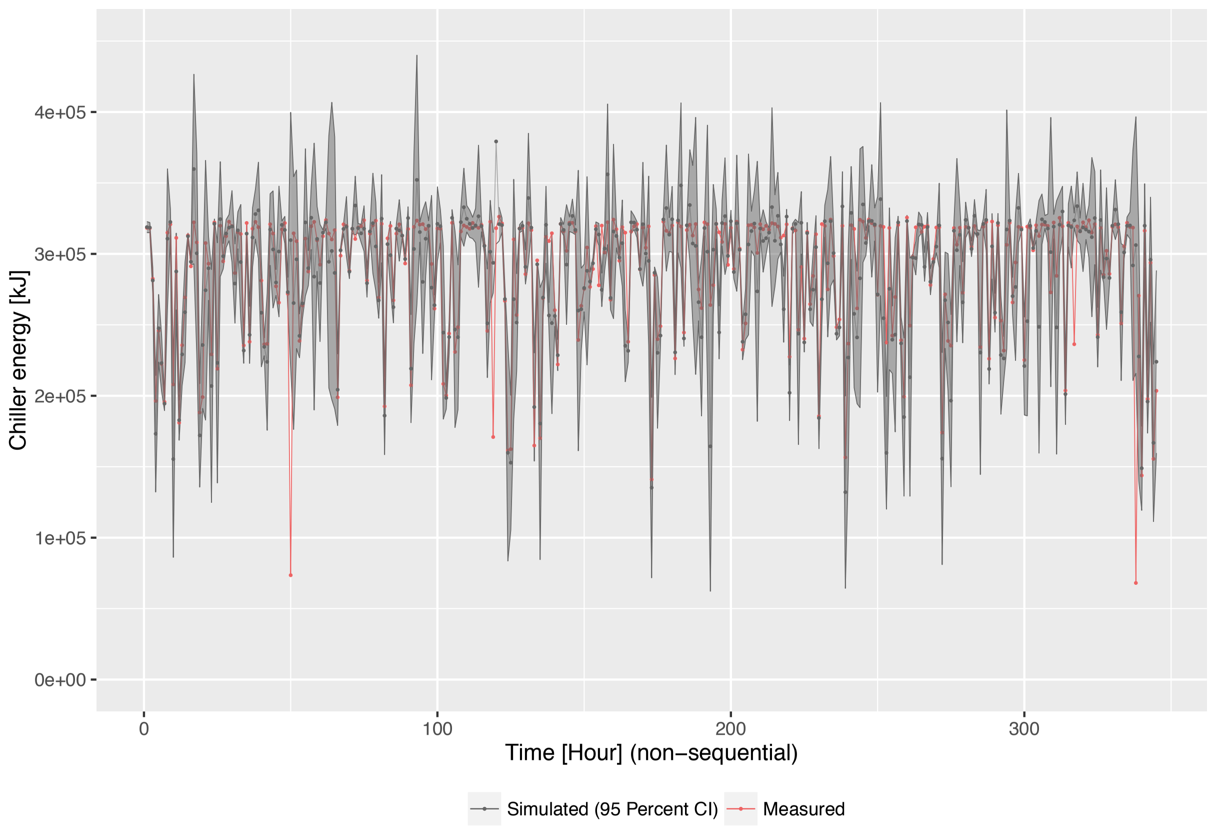

Figure 3.4: Case study 1: comparing 95% confidence interval calibrated predictions with field observations from a hold-out test dataset.

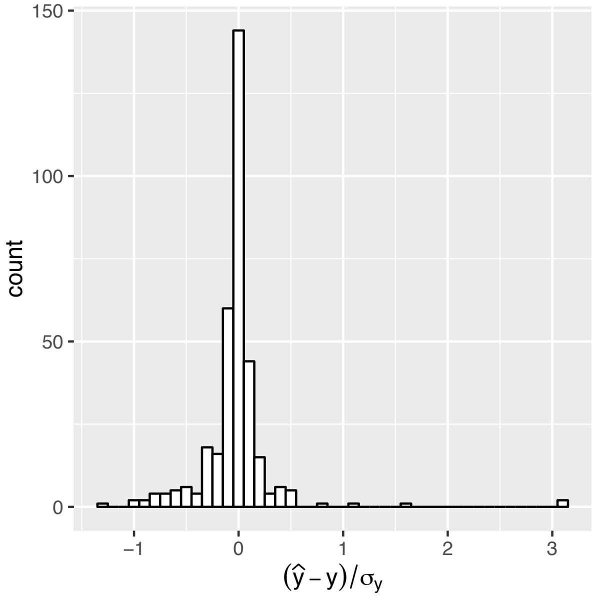

Figure 3.5: Case study 1: histogram of residuals (test data) standardized by standard deviation of observed output.

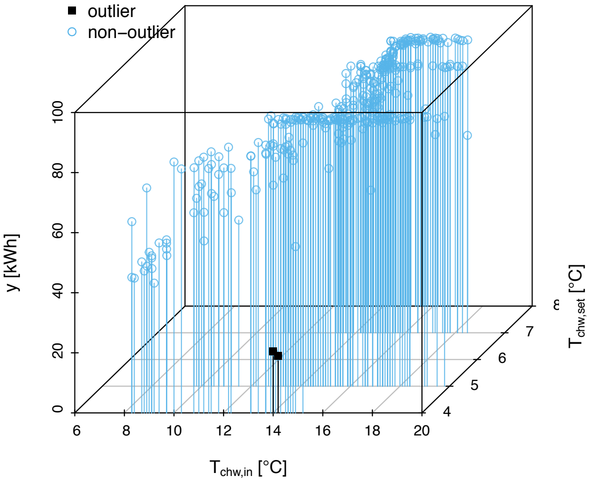

Figure 3.6: Case study 1: outliers in testing dataset.

Figure 3.7 shows the resulting mean posterior predictions[1] with and without the discrepancy term $\delta(x)$, as well as how they compare with the observed values $y(x)$ of the hold-out test dataset. To gain a better understanding of the simulator's prediction, Figure 3.7 shows how the predictions and discrepancies varies against different input factors, including entering chilled water temperature $T_{chw,in}$, chilled water mass flow rate $\dot{m}_{chw}$, and entering condenser water temperature $T_{cw,in}$. As mentioned in Chapter 2.2.2, the discrepancy term $\delta(x)$ is used to estimate how well the model predictions matches actual observed values. Overall, it can be seen that the simulator $\eta(x,t)$ underestimates (negative bias) the chiller's energy at low $T_{chw,in}$ and $\dot{m}_{chw}$, and overestimates (positive bias) the chiller's energy at high $T_{chw,in}$ and $\dot{m}_{chw}$. This compensation between positive and negative bias explains why the NMBE is relatively small at 3.7% (Table 3.2).

A closer look at Figure 3.7 shows that the discrepancy term is slightly positive when $T_{chw,in} < 12.5°C$ and $\dot{m}_{chw} < 250000\ kg/h$. However, the discrepancy term flips and becomes negative when $T_{chw,in} > 15°C$ and $\dot{m}_{chw} > 300000\ kg/h$. This suggests that the simulator predictions $\eta(x,t)$ tend to underestimate chiller energy consumption when $T_{chw,in} < 12.5°C$ and $\dot{m}_{chw} < 250000\ kg/h$ and overestimate chiller energy consumption when $T_{chw,in} > 15°C$ and $\dot{m}_{chw} > 300000\ kg/h$. Figure 3.7 also shows that including the discrepancy term $\delta(x)$ reduces the overall bias across different $T_{chw,in}$ and $\dot{m}_{chw}$. Additionally, a decrease in CVRMSE from 12% to 9.4% was also observed .

Trace plots and $\hat{R}$ of the calibration parameters (t_1, t_2), correlation hyperparameters ($\beta_1^\eta, ..., \beta_6^\eta$ and $\beta_1^\delta, ..., \beta_4^\delta$), and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) were used to assess convergence. See Chapter 2.4.4 for a description of both metrics. From Figures A.1 and A.2, it can be seen that all components of the posterior distribution are well mixed and have converged to a common stationary distribution. $\hat{R}$ is also within 1 ± 0.1 for all calibration parameters and hyperparameters of the GP model.

[1] The mean posterior predictions are computed by taking the mean of the predictions at input settings given by the hold-out test dataset

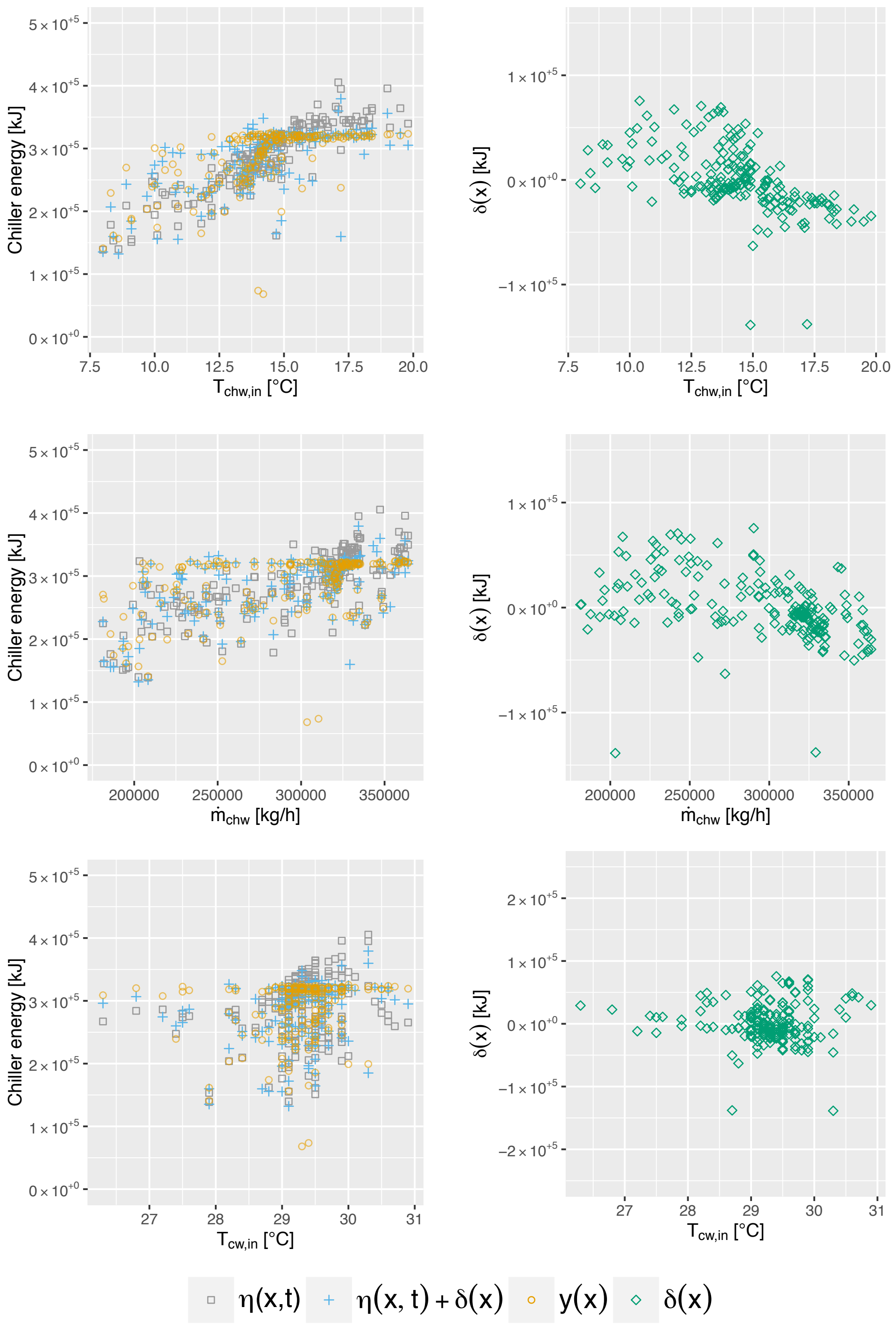

Figure 3.7: Case study 1: posterior mean estimates for the field observations $y(x)$ (chiller energy consumption), the calibrated simulator output $\eta(x,t)$, the discrepancy term $\delta(x)$ and the calibrated prediction with discrepancy term $\eta(x,t)+\delta(x)$.

3.2 Case study 2 - office building in Pennsylvania U.S.A

3.2.1 Case study and model description

In case study 2, the proposed method (Figure 2.1) was applied to the cooling system of a ten-story office building located in Pennsylvania U.S.A.

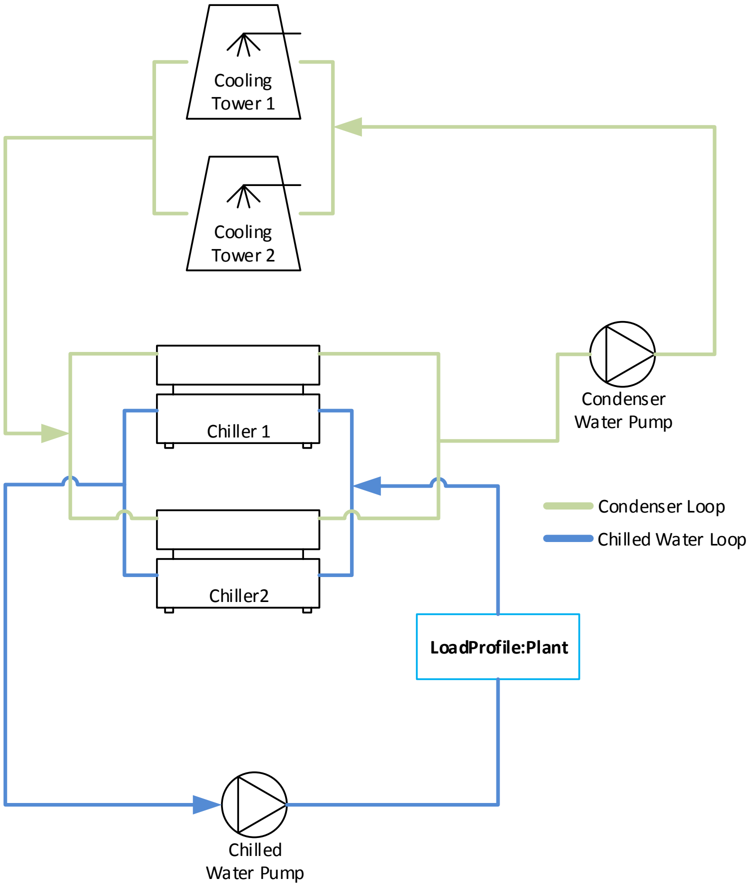

Figure 3.8: Case study 2: simplified representation of cooling system modeled using EnergyPlus.

The cooling system was modeled using EnergyPlus version 8.5 and consists of the following functional parts (Figure 3.8): (a) loads from a cooling coil that transfers heat from air to water; (b) two chillers connected in parallel that cools the water; (c) chilled water distribution pumps that send chilled water to the cooling coil; (d) condenser water pumps for circulation in the condenser loop; and (e) two cooling towers in parallel that reject heat from the chillers to the atmosphere. The following EnergyPlus objects were used to model this cooling system (LBNL, 2016c): (a) LoadProfile:Plant; (b) Chiller:Electric:EIR; (c) Pump:VariableSpeed; and (d) CoolingTower:SingleSpeed. The calibration was carried out using cooling electricity consumption as the output of interest, which was calculated as the sum of electricity consumption by the chillers, chilled water pumps, condenser water pumps and cooling towers. Table 3.3 shows a list of the data points used in this study as well as the sensors used for their measurements.

Table 3.3: Case study 2: List of measurements and sensors.

Description	**Metering instrument**
Cooling coil inlet water temperature	Onicon System 10 Btu Meter
Cooling coil outlet water temperature	Onicon System 10 Btu Meter
Chilled water flow rate	Onicon System 10 Btu Meter
Chiller 1 power	WNC-3D-480-MB
Chiller 2 power	WNC-3D-480-MB
Chilled water pump power	WNC-3D-480-MB
Condenser water pump power	WNC-3D-480-MB
Cooling tower 1 fan power	WNC-3D-480-MB
Cooling tower 2 fan power	WNC-3D-480-MB

The EnergyPlus LoadProfile:Plant object was used to simulate a scheduled demand profile when the coil loads are already known (LBNL, 2016c). This makes it possible to isolate and calibrate the HVAC system without any propagation of uncertainties due to calculation of building loads. Demanded loads were calculated based on measured water mass flow rate and temperature difference across the cooling coil according to Equation 3.8 (LBNL, 2016a). Given the coil load Q_{load} (Equation 3.8) and the fraction of peak flow rate (Equation 3.9) at a given timestep, the simulation then steps through the code of each HVAC component to calculate its power consumption.

$$Q_{load} = \rho V_{chw} C_p (T_{in} - T_{out}) \tag{3.8}$$

where,

Q_{load} is the scheduled coil load $[W]$,

$\rho = 1.225$ is the density of water $[kg/m^3]$,

V_{chw} is the volumetric flow rate $[m^3/s]$,

C_p is the specific heat of water $[J/kg°C]$,

T_{in} is the inlet water temperature $[°C]$,

T_{out} is the outlet water temperature $[°C]$.

$$V_{frac} = \frac{V_{chw}}{V_{chw,max}} \tag{3.9}$$

where,

V_{frac} is the fraction of peak flow rate $[-]$,

V_{chw} is the volumetric flow rate $[m^3/s]$,

$V_{chw,max}$ is the maximum chilled water flow rate that was measured $[m^3/s]$.

The EnergyPlus Chiller:Electric:EIR object uses performance information at reference conditions along with three performance curves to determine a chiller's performance at off-reference conditions (LBNL, 2016a). The three performance curves are: (1) Cooling Capacity Function of Temperature Curve (Equation 3.10); (2) Energy Input to Cooling Output Ratio Function of Temperature Curve (Equation 3.11); and (3) Energy Input to Cooling Output Ratio Function of Part Load Ratio Curve (Equation 3.12).

$$CapFT = a_1 + b_1(T_{cw,l}) + c_1(T_{cw,l})^2 + d_1(T_{cond,e}) + e_1(T_{cond,e})^2 + f_1(T_{cw,l})(T_{cond,e}) \tag{3.10}$$

where,

$CapFT$ is the cooling capacity factor, equal to 1 at reference conditions $[-]$,

$T_{cw,l}$ is the leaving chilled water temperature [$^{\circ}C$],

$T_{cond,e}$ is the entering condenser fluid temperature [$^{\circ}C$].

$$EIRFT = a_2 + b_2(T_{cw,l}) + c_2(T_{cw,l})^2 + d_2(T_{cond,e}) + e_2(T_{cond,e})^2 + f_2(T_{cw,l})(T_{cond,e}) \quad (3.11)$$

where,

$EIRFT$ is the energy input to cooling output factor, equal to 1 at reference conditions [$-$],

$T_{cw,l}$ is the leaving chilled water temperature [$^{\circ}C$],

$T_{cond,e}$ is the entering condenser fluid temperature [$^{\circ}C$].

$$EIRFPLR = a_3 + b_3(PLR) + c_3(PLR)^2 \quad (3.12)$$

where,

$EIRFPLR$ is the energy input to cooling output factor, equal to 1 at reference conditions [$-$],

$PLR = \frac{\text{cooling load}}{\text{chiller's available cooling capacity}}$ is the part load ratio [$-$].

Using the outputs from Equations 3.10 to 3.12, chiller power can then be calculated by Equation 3.13.

$$P_{chiller} = \frac{Q_{ref}}{COP_{ref}}(CapFT)(EIRFT) \quad (3.13)$$

where,

$P_{chiller}$ is the chiller power at a specific PLR [W],

Q_{ref} is the chiller capacity at reference conditions [W],

COP_{ref} is the chiller's coefficient of performance (COP) at reference conditions [$-$].

The parameters of this chiller model (Q_{ref}, COP_{ref}, regression coefficients of Equations 3.10, 3.11 and 3.12) were determined based on measured data using the reference-curve method that was proposed by Hydeman and Gillespie Jr (2002).

The EnergyPlus Pump:VariableSpeed object calculates the power consumption of a variable speed pump using a cubic curve (Equation 3.14) (LBNL, 2016c).

$$FFLP = a_5 + b_5(PLR) + c_5(PLR)^2 + d_5(PLR)^3 \tag{3.14}$$

where,
$FFLP$ is the fraction of full load power $[-]$,
$PLR = \frac{\text{Flow Rate}}{\text{Design Flow Rate}}$ is the part load ratio $[-]$.

Using the $FFLP$ calculated by Equation 3.14, pump power is then calculated by Equation 3.15. The value of P_{design} is set based on measurements of the pump's flow rate and power. A value of 1 was assigned to motor efficiency because pump motor inefficiencies are already accounted for in the measurements of flow and power. Least squares regression is used to compute the coefficients of Equation 3.14, with $FFLP$ and PLR calculated by Equations 3.16 and 3.17 respectively.

$$P_{pump} = (P_{design})(FFLP)(Eff_{motor}) \tag{3.15}$$

where,
P_{pump} is the pump power $[W]$,
P_{design} is the design pump power consumption $[W]$,
$FFLP$ is the fraction of full load power $[-]$,
Eff_{motor} is the pump motor efficiency $[-]$.

$$FFLP_i = \frac{power_i}{max(power_1, power_2, ..., power_n)} \tag{3.16}$$

where,

$FFLP_i$ is the fraction of full load power for the i^{th} measurement $[-]$,

$power_i$ is the i^{th} measured pump power $[W]$.

$$PLR_i = \frac{flow_i}{max(flow_1, flow_2, ..., flow_n)} \tag{3.17}$$

where,

PLR_i is the part load ratio for the i^{th} measurement $[-]$,

$flow_i$ is the i^{th} measured pump flow rate $[m^3/s]$.

Taken over time, the total energy consumption of the cooling system is then calculated as the sum of the energy consumption for each individual component.

3.2.2 Uncertainty quantification and sensitivity analysis

Before calibrating the model, sensitivity analysis was used to screen out non-influential parameters. The aim is to mitigate over-parameterization and reduce computational cost without affecting model accuracy. Table 3.4 shows the uncertain parameters θ as well as their respective initial, minimum and maximum values. 10 parameters were modeled as uncertain. Although the set of uncertain parameters θ are specific to this case study, they correspond to the set of parameters typically selected as random variables for a centralized cooling system.

Design fan power θ_9 and nominal capacity θ_{10} of cooling towers 1 and 2 were modeled as a single random variable because they have the same make and model and were installed at the same time. On the contrary, based on the measured data, chillers 1 and 2 have very different capacity (θ_1 and θ_3) and COP (θ_2 and θ_4) at reference conditions. Therefore, these parameters were modeled as separate random variables. The initial value for each parameter was assigned based on either 1) measured data (as described in the previous section); or 2) as-built architectural and mechanical drawings. Pump motor efficiency was assigned a wide range of 0.6 to 1.0 because no information on pump motor efficiency was available. To be conservative, the remaining 8 parameters were varied ±20% of their initial values.

Table 3.4: Case study 2: list of model parameters and their range.

Model parameter	**Symbol**	**Initial Value**	**Min**	**Max**
Chiller 1:				
Reference Capacity (W)	θ_1	653378	522702	784053
Reference COP $(-)$	θ_2	6.86	5.49	8.23
Chiller 2:				
Reference Capacity (W)	θ_3	243988	195190	292785
Reference COP $(-)$	θ_4	2.32	1.85	2.78
Chilled water pump:				
Design Power Consumption (W)	θ_5	18190	14552	21828
Motor Efficiency $(-)$	θ_6	1.0	0.6	1.0
Condenser water pump:				
Design Power Consumption (W)	θ_7	11592	9274	13911
Motor Efficiency $(-)$	θ_8	1.0	0.6	1.0
Cooling Tower 1 and 2:				
Design Fan Power (W)	θ_9	11592	9274	13911
Nominal Capacity (W)	θ_{10}	549657	439726	659589

The Morris method was used to carry out the sensitivity analysis. Detailed description of the method can be found in Chapter 2.2.1. Implementation was carried out using R sensitivity package (Pujol et al., 2016). All uncertain parameters θ were assigned a uniform distribution. The Morris method was applied with 10 parameters (θ_1 to θ_{10}), 20 trajectories and 16 levels. This led to an experimental design of $20 \times (10 + 1) = 220$ simulation runs. Using results from the simulation runs, we generate a graphical plot of μ^* against σ (Figure 3.9) to better interpret the sensitivity measures.

From Figure 3.9, it can be seen that parameters θ_5 to θ_9 have μ^* and σ close to zero, indicating that these parameters have negligible influence on the simulation output (total cooling energy consumption). Considering both μ^* and σ, we can conclude that parameters θ_1, θ_2, θ_3, θ_4, and θ_{10} are important with θ_2, θ_3 and θ_4 appearing to have effects that involve either curvature or interactions. This result is not surprising because parameters θ_1 to θ_4 are parameters of the chiller component (Table 3.4), and therefore is expected to have the greatest influence on cooling energy.

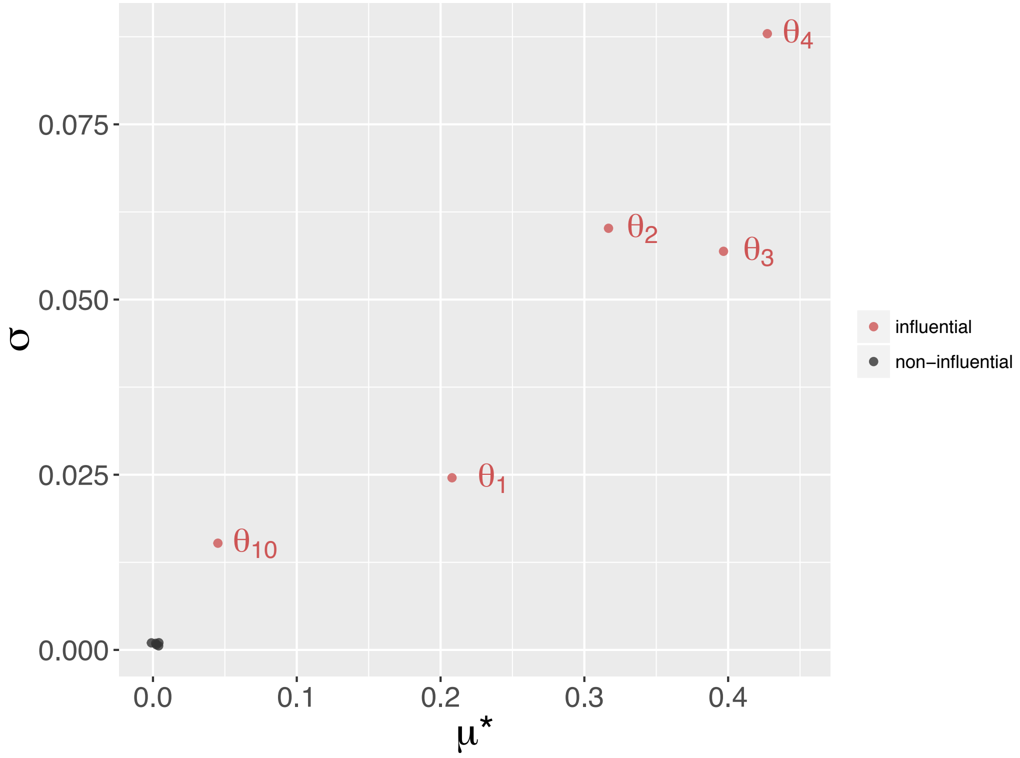

Figure 3.9: Case study 2: graphical plot of sensitive measures μ^* and σ for parameters θ_1 to θ_{10} (Table 3.4). The closer the parameters are to the upper right, the more sensitive the parameter. Parameters close to the bottom left are non-influential parameters.

Based on the sensitivity analysis, only parameters θ_1, θ_2, θ_3, θ_4, and θ_{10} would be used for the Bayesian calibration of the EnergyPlus model. Note that the notation t is used to denote the calibration parameters, i.e., the influential parameters that were selected from the set of uncertain parameters θ. Therefore, the inputs and output used for the Bayesian calibration of this EnergyPlus model can be summarized as:

- Observed output $y(x)$: Measured energy consumption of the cooling system [kWh].
- Simulation output $\eta(x, \theta)$: Predicted energy consumption of the cooling system [kWh].
- Observed inputs x:
  (a) x_1: Load of cooling coil Q_{load} [W].
  (b) x_2: Fraction of peak chilled water flow rate [$-$].
- Calibration parameters t:
  (a) $t_1 = \theta_1$: Chiller 1 reference capacity [W].
  (b) $t_2 = \theta_2$: Chiller 1 reference COP [$-$].
  (c) $t_3 = \theta_3$: Chiller 2 reference capacity [W].
  (d) $t_4 = \theta_4$: Chiller 2 reference COP [$-$].
  (e) $t_5 = \theta_{10}$: Nominal capacity of cooling towers [W].

3.2.3 Bayesian calibration

Bayesian calibration as described in Chapter 2.2.2 was used for the calibration. Data collection took place between June 1, 2014 and August 31, 2014. After removing erroneous data and all instances for which the outcome or any of the inputs are missing, the dataset contained 719 samples. Of the data collected, 30% (219 samples) were used as a test hold-out dataset and the remaining 70% (503 samples) were used for calibrating the model. Hourly data was used for the calibration.

To reduce computation cost, 2 strategies were employed that include using information theory to reduce the number of samples (Chapter 2.4.2) and using NUTS (a more efficient MCMC algorithm) to explore the posterior distribution (Chapter 2.4.3). The observed input factors x is two dimensional with components corresponding to: 1) x_1: cooling coil load; and 2) x_2: fraction of peak chilled water flow rate. The observed output $y(x)$ is the measured energy consumption of the cooling system. Therefore the experimental design corresponding to the field observations is a dataset $D^F = \begin{bmatrix} y & x_1 & x_2 \end{bmatrix}$ where $D^F \in \mathbb{R}^{503 \times 3}$ (there are 503 samples because only 70% of the data was used for the calibration).

To learn about the calibration parameters t, $m = 200$ EnergyPlus simulations were run at different combinations of (x, t). Maximin LHS (Stein, 1987) was used to generate values for the calibration parameters t_1, t_2, t_3, t_4, and t_5. Running each of the 200 simulations at the same input factors x generates $503 \times 200 = 100600$ samples. Therefore, the experimental design for the simulations is a dataset $D^S = \begin{bmatrix}\eta(x,t) & x_1 & x_2 & t_1 & t_2 & t_3 & t_4 & t_5\end{bmatrix}$ where $D^S \in \mathbb{R}^{100600 \times 8}$.

To sample a representative subset D^F_{sub} from the field dataset D^F and a representative subset D^S_{sub} from the simulation dataset D^S, random samples of varying sizes were generated and their corresponding sample quality computed (Chapter 2.4.2). Figure 3.10 shows the relationship between sample size and sample quality. Taking into consideration both sample quality and computation cost, a sample size of 80 for D^F_{sub} and 640 for D^S_{sub} was used. Then, D^F_{sub} and D^S_{sub} was used for the calibration of the building energy model instead of the entire dataset D^F and D^S.

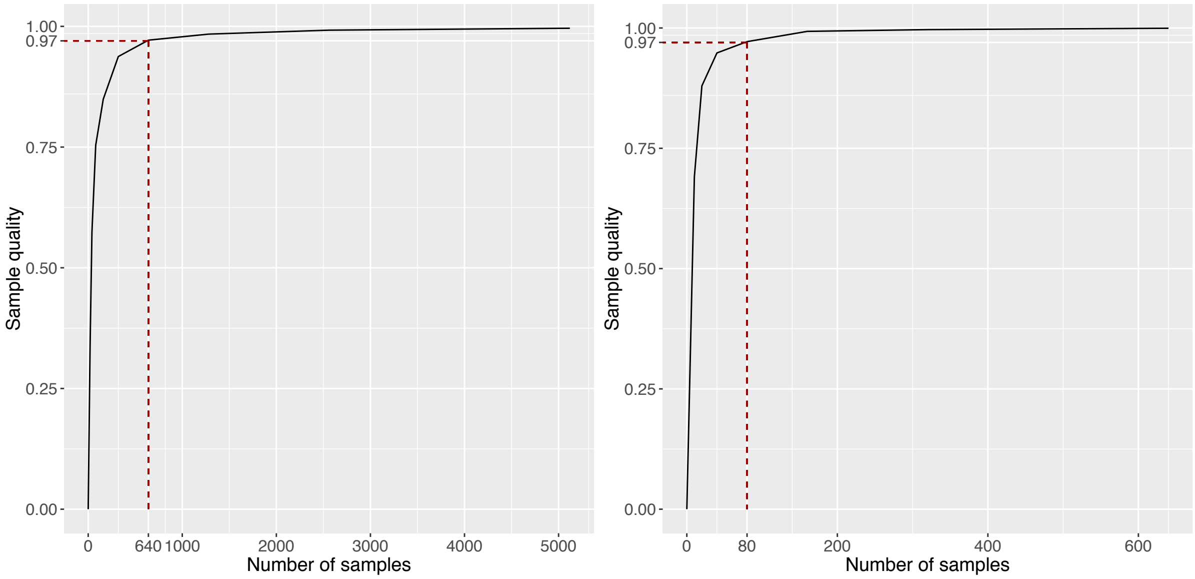

Figure 3.10: Case study 2: number of samples against sample quality for different subsets of simulation data D^S (left plot) and field data D^F (right plot). 640 random samples from simulation data D^S and 80 random samples from field data D^F is sufficient to represent the whole dataset.

Using the formulation by Higdon et al. (2004), D^F_{sub} and D^S_{sub} were combined in a Gaussian process model. Following Chapter 2.2.2, the GP model for $\eta(.,.)$ was specified using a mean function that returns the zero vector and a covariance function of the form:

$$\begin{aligned} &Cov((x,t),(x',t')) = \\ &\frac{1}{\lambda_\eta} exp\bigg\{ -\sum_{j=1}^{2} \beta^\eta_j |x_{ij} - x'_{ij}|^2 - \sum_{k=1}^{5} \beta^\eta_{p+k} |t_{ik} - t'_{ik}|^2 \bigg\} \end{aligned} \tag{3.18}$$

The GP model for the discrepancy term $\delta(x)$ was also specified with a mean function that returns the zero vector and a covariance function of the form:

$$Cov(x,x') = \frac{1}{\lambda_\delta} exp\bigg\{ -\sum_{j=1}^{2} \beta^\delta_k |x_{ij} - x'_{ij}|^2 \bigg\} \tag{3.19}$$

Given that there are 5 calibration parameters and 2 input factors, parameters of the posterior that need to be estimated include: 1) the calibration parameters $t_1, ..., t_5$; 2) the correlation hyperparameters of the GP model $\beta^\eta_1, ..., \beta^\eta_7, \beta^\delta_1, \beta^\delta_2$; and 3) the variance hyperparameters of the GP model λ_η, λ_δ and λ_ϵ. NUTS, an extension of HMC was used to explore the posterior distributions. Four independent chains of 500 iterations per chain were run. To be conservative, the first 250 iterations (50%) values were discarded as warmup/burn-in to reduce the influence of starting values.

Figure 3.11 shows the posterior distribution of the calibration parameters t. From the Figure, it can be observed that the posterior estimates suggest that chiller 1 and chiller 2 have a lower rated capacity (t_1 and t_3) as compared to its initial estiamte (Table 3.4). This is based on the observation that the histogram for the posterior estimates for t_1 and t_3 are right-skewed (Figure 3.11). Similarly, the posterior estimates for t_2 and t_4 are left-skewed, suggesting that chillers 1 and 2 have a higher rated COP than initially expected. On the contrary, the posterior distribution for the nominal capacity of the the cooling tower t_5 appears to be uniformly distributed. Also, the posterior distribution for t_3 and t_4 appears to be more skewed as compared to the posterior estimates for t_1 and t_2. In fact, the posterior distribution for t_1 and t_2 are as wide as their prior probabilities, suggesting little to no reduction in their uncertainties. This suggests that only t_3 and t_4 had an effect on the output and the remaining parameters (t_1, t_2 and t_5) adds little to the overall uncertainty.

Table 3.5: Case study 2: Prior distribution and summary statistics for posterior distribution of calibration parameters.

Parameter [units]	**Prior**	**Posterior**			
		mean	**2.5 Percentile**	**50 Percentile**	**97.5 Percentile**
t_1 $[W]$	$\mathcal{U}(522702, 784053)$	631815	525887	624418	766085
t_2 $[-]$	$\mathcal{U}(5.5, 8.2)$	7.1	5.7	7.2	8.2
t_3 $[W]$	$\mathcal{U}(195190, 292785)$	225232	196741	219819	278445
t_4 $[-]$	$\mathcal{U}(1.9, 2.8)$	2.5	2.0	2.6	2.8
t_5 $[W]$	$\mathcal{U}(439726, 659589)$	559549	447494	563538	653525

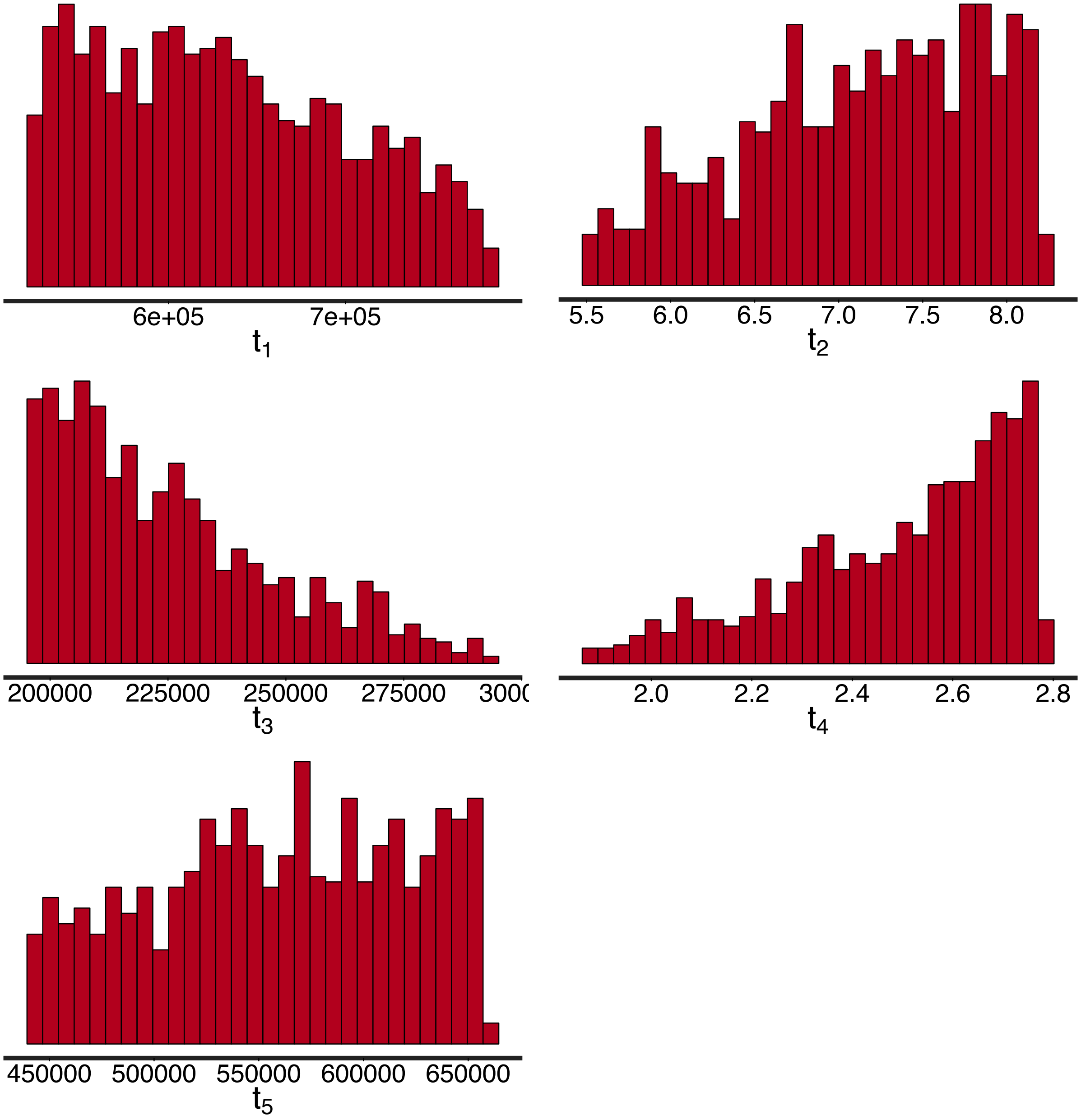

Figure 3.11: Case study 2: posterior distribution (excluding warmup) of calibration parameters: t_1 chiller 1 reference capacity $[W]$, t_2 chiller 1 reference COP $[-]$, t_3 chiller 2 reference capacity $[W]$, t_4 chiller 2reference COP $[-]$, and t_5 cooling tower nominal capacity $[W]$.

3.2.4 Model evaluation

First, prediction accuracy is assessed using standard metrics of agreement and a visual comparison of model predictions and observed values. To prevent bias, predictions by the calibrated model is compared to measurements in the hold-out test dataset. Of the 719 observations that were collected, 30% of the data (216 samples) was withheld and not used for the calibration. Figure 3.13 shows the hold-out samples and how they compare with the 95% confidence interval of the predictions. Overall, the figure shows good agreement between the calibrated predictions and the measurements in the hold-out dataset with some overestimation and underestimation, i.e., some observed values were slightly below or above the 95% confidence interval of the calibrated predictions. Over all 216 hold-out samples, CVRMSE and NMBE were 6.0% and -0.3% respectively, meeting the acceptable thresholds for hourly calibration data set by various standards and guidelines (See Table 1.1). Agreement between predictions and observations in the hold-out dataset is also illustrated in Figure 3.5, which shows a histogram of residuals for the calibrated predictions standardized by the observed values and its standard deviation. The histogram shows that the residuals are centered around zero with all residuals being within $\pm 1\sigma_y$.

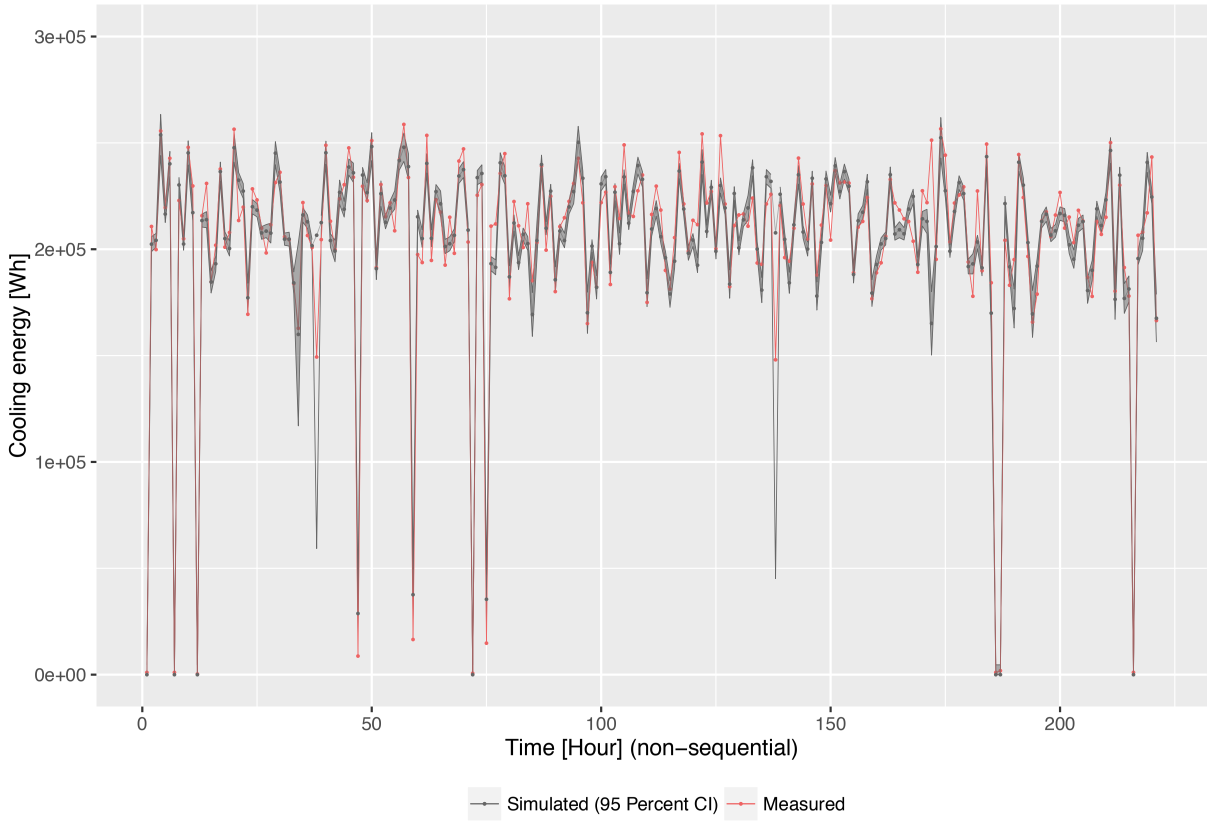

Figure 3.12: Case study 2: comparing 95% confidence interval calibrated predictions with field observations on the hold-out test dataset

Table 3.6: Case study 2: CVRMSE and NMBE computed with posterior mean estimates with and without the discrepancy term $\delta(x)$.

Statistical Formulation	**CVRMSE (%)**	**NMBE (%)**
$\eta(x, \theta) + \epsilon(x)$	15.2	11.9
$\eta(x, \theta) + \delta(x) + \epsilon(x)$	6.0	-0.3

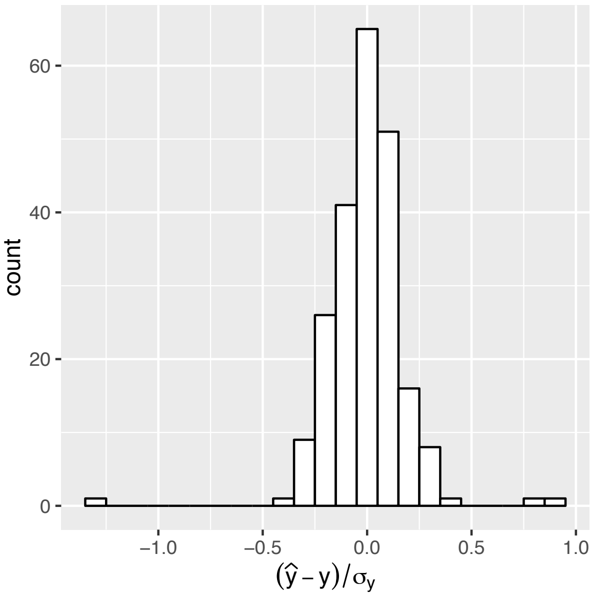

Figure 3.13: Case study 2: histogram of residuals (test data) standardized by standard deviation of observed output.

Figure 3.14 shows the resulting mean posterior predictions[2] with and without the discrepancy term $\delta(x)$, as well as how they compare with the observed values $y(x)$ of the hold-out test dataset. To gain a better understanding of the simulator's prediction, Figure 3.14 shows how the predictions and discrepancies varies against different input factors, which include, cooling coil load Q_{load} and fraction of peak chilled water flow rate V_{flow}. It is interesting to note that the range of V_{frac} is above 0.95 a majority of the time, i.e., the pumps are operating close to

[2] The mean posterior predictions are computed by taking the mean of the predictions at input settings given by the hold-out test dataset

maximum flow rates most of the time. Since the chilled water pumps installed are variable speed pumps, this indicates an opportunity for operating the cooling system more efficiently.

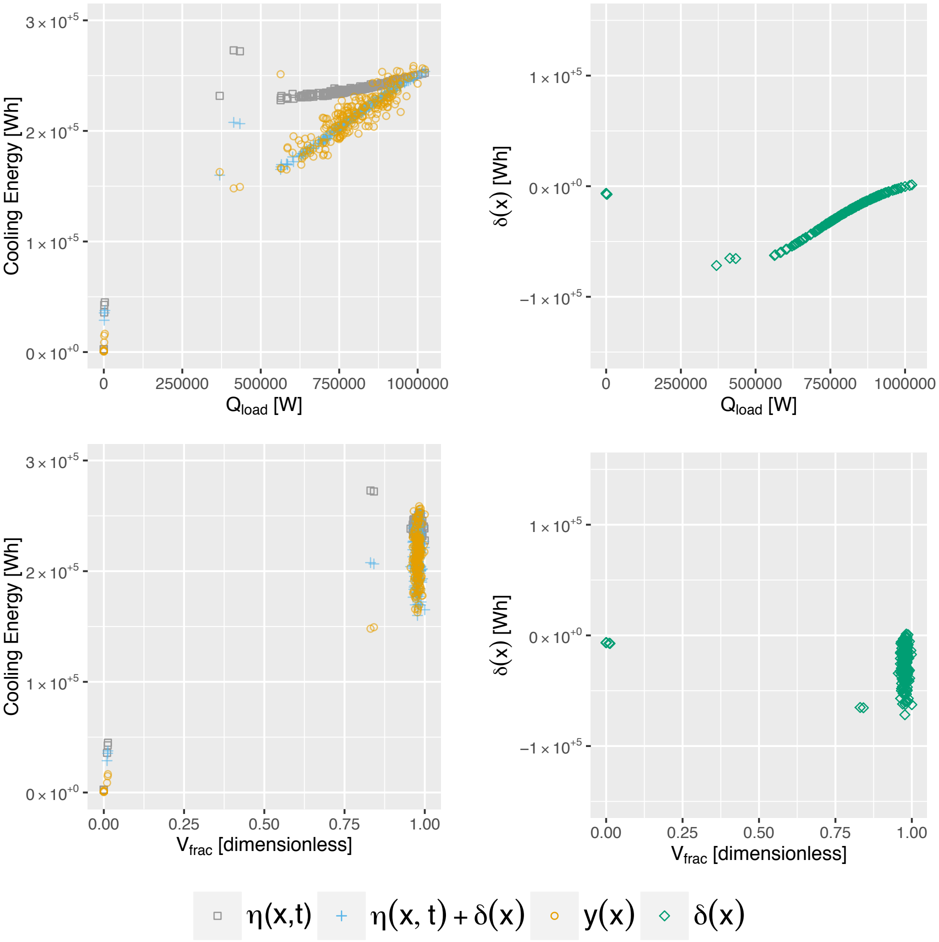

Figure 3.14: Case study 2: posterior mean estimates for the field observations $y(x)$ (cooling system energy consumption), the calibrated simulator output $\eta(x,t)$, the discrepancy term $\delta(x)$ and the calibrated prediction with discrepancy term $\eta(x,t) + \delta(x)$.

From Figure 3.14, it can be seen that the discrepancy term is negative across various values of Q_{load} and V_{frac}. This suggests that the simulator $\eta(x,t)$ tends to overestimate the energy consumption of the cooling system. This also explains a positive NMBE of 11.9%. A closer look at Figure 3.14 reveals that the discrepancy between model predictions and observed values are the largest when $2.5 \times 10^5\ W < Q_{load} < 5 \times 10^5\ W$ and gradually decreases towards zero as Q_{load} increases. On the contrary, discrepancy between model predictions and observed values are relatively constant across different V_{chw} values. From Figure 3.14, it can be seen that the varying bias across different values of Q_{load} is well adjusted by the discrepancy term $\delta(x)$. This is obvious from a downward shift of the simulation predictions (grey squares), resulting in calibrated predictions (blue crosses) that better match observed values (yellow circles). The decrease in overall bias can also be observed by a decrease in NMBE from 11.9% to -0.3% (Table 3.2). In addition, CVRMSE is also reduced by more than one-half from 15.2% to 6.0%.

Although the discrepancy term $\delta(x)$ is able to account for the differences between the calibrated simulation and the observed values y, care should be taken when interpreting the posterior estimates for the calibration parameters t. This is because the discrepancy term $\delta(x)$ is fairly large and does not stay constant over x. This large discrepancy leads to a high degree of uncertainty regarding the calibration parameters t. This suggests investigating aspects of the model that deal with the cooling load to ensure that the model is well calibrated and that the discrepancy term $\delta(x)$ is not being overfitted by the data.

Trace plots and $\hat{R}$ of the calibration parameters ($t_1, ..., t_5$), correlation hyperparameters ($\beta_1^\eta, ..., \beta_7^\eta$ and $\beta_1^\delta, \beta_2^\delta$), and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) were used to assess convergence. See Chapter 2.4.4 for a description of both metrics. From Figures A.3 and A.4, it can be seen that all components of the posterior distribution are well mixed and have converged to a common stationary distribution. $\hat{R}$ is also within 1 ± 0.1 for all calibration parameters and hyperparameters of the GP model.

3.3 Case Study 3 - mixed-use building on a university campus in Singapore

3.3.1 Case study and model description

In case study 3, the proposed method (Figure 2.1) was applied to a whole building energy model of a mixed-use building located on a university campus in Singapore. The building is a three story mixed-use building located at the National University of Singapore (Figure 3.15).

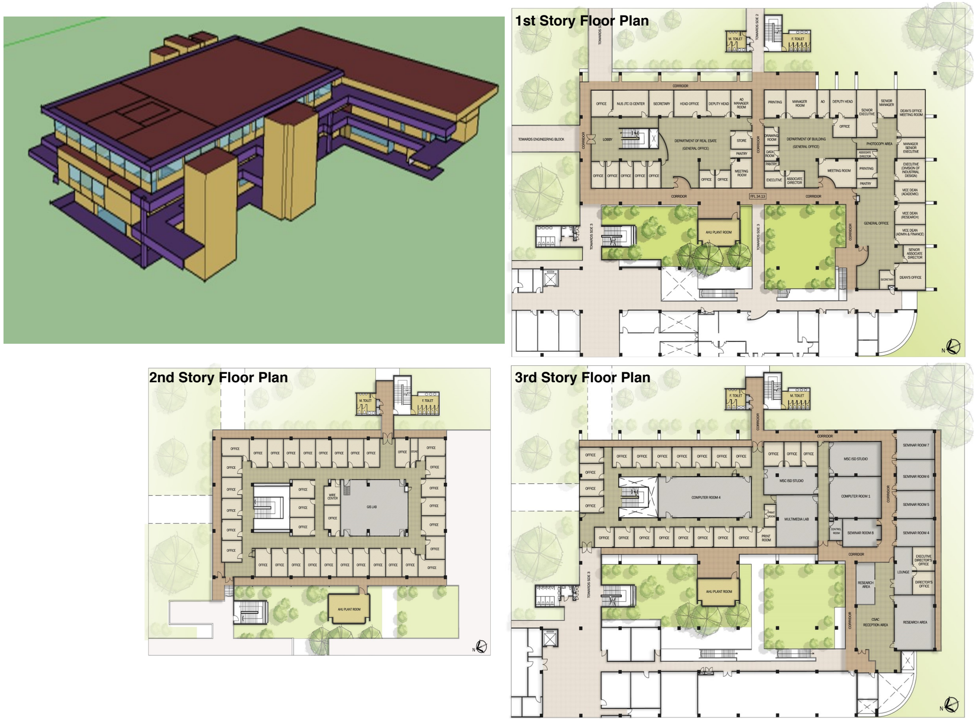

Figure 3.15: Case study 3: Model view and floor plans of mixed-use building located on a university campus in Singapore.

The model was created with EnergyPlus version 8.5 based on the following information to gain a preliminary understanding of the envelope properties, the spatial layout and the HVAC system: 1) as-built architectural drawings; 2) HVAC drawings and specifications; and 3) electrical line drawings. Figure 3.15 shows the floor plans of the building. Altogether, eleven zone types were identified (Table 3.7). Given weather information, a description of the building's geometry and its HVAC system, EnergyPlus is able to predict the energy consumption of the building. Measured data used for this study includes 1) the outdoor dry-bulb temperature, 2) the outdoor relative humidity and 3) the cooling energy consumption of the building. Measurements of the outdoor dry-bulb temperature and relative humidity were measured at a height of about 90 meters above sea level. The Vaisala CS500 temperature sensor ($\pm 0.5°C$) and the Vaisala CS500 relative humidity sensor ($\pm 2.5\%$) were used to measure the outdoor dry-bulb temperature and relative humidity respectively.

Table 3.7: Case study 3: Initial values for parameters of internal load components of EnergyPlus model.

Space Type	**Occupancy Density**	**Lighting Power Density**	**Equipment Power Density**
	$[m^2/person]$	$[W/m^2]$	$[W/m^2]$
Circulation	0	11.54	0
Computer room	5.8	16.65	23.47
General office	83.9	15.01	6.45
Geographical Information Systems lab	5.4	17.57	23.47
Library	26.2	7.96	23.47
Research lab	55.7	8.34	13.99
Architecture studio	21.4	13.03	18.44
Multimedia lab	10.8	6.03	23.47
Research room	11	5.94	13.99
Seminar room	10.8	12.28	0
Staff room	46	22.13	6.74

To better quantify the inputs to the EnergyPlus model, an energy audit was carried out. This energy audit included a site walkthrough amongst other data collection efforts. Given limited resources, it is impractical to collect data over a long period of time. Table 3.7 shows the peak occupancy density, peak lighting power density and the peak equipment power density for each zone type. Lighting loads were determined based on: 1) spot measurements of representative fixtures and the frequency of their usage over a period of one week; and 2) the fixture type and its wattage. In a similar way, equipment loads were determined by taking into consideration: 1) spot measurements of representative zone types and the frequency of their usage over a period of one week; and 2) electrical line drawings. Occupancy levels were determined by recording the number of people in a representative zone type across a period of two weeks. A high infiltration rate of 2 ACH was assumed given that the building is old and that there is a high occurrence of cracks and poorly sealed windows.

3.3.2 Uncertainty quantification and sensitivity analysis

Before calibrating the model, sensitivity analysis was used to screen out non-influential parameters (Section 2.2.1). The aim is to mitigate over-parameterization and reduce computational cost without affecting model accuracy. Table 3.8 lists the uncertain parameters θ as well as their respective initial, minimum and maximum values. 17 parameters were modeled as uncertain. Thermal properties of the opaque envelope θ_1 to θ_9 were varied $\pm$ 2 standard deviations σ, where $\sigma = 5\%$, 1% and 12.25% for conductivity, density, and specific heat respectively (Macdonald, 2002). Uncertainties for window U-value θ_{10} and window SHGC θ_{11} were determined taking into account that the glass installed is a 6mm clear glass. Metabolic rate θ_{17} was assigned based on the fact that the occupants were seated and carrying out light work (Macdonald, 2002). To be conservative, the remaining parameters were varied ±20% of their initial values. Since different zones have different occupancy density, equipment power density and lighting power density, they were varied by applying a multiplier to their initial values (Table 3.7).

The Morris method was used to carry out the sensitivity analysis. Detailed description of the method is provided in Chapter 2.2.1. Implementation was carried out using R sensitivity package (Pujol et al., 2016). All uncertain parameters θ were assigned a uniform distribution. The Morris method was applied with 17 parameters (θ_1 to θ_{17}), 30 trajectories and 16 levels. This led to an experimental design with $30 \times (17 + 1) = 540$ simulation runs. Using results from the simulation runs, a graphical plot of μ^* against σ (Figure 3.16) was used to better interpret the sensitivity measures.

Table 3.8: Case study 3: list of model parameters and their range.

Model parameter	**Symbol**	**Initial Value**	**Min**	**Max**
Opaque Envelope Thermal Properties:				
Gypsum conductivity $[W/m \cdot K]$	θ_1	0.16	0.144	0.176
Gypsum density $[kg/m]$	θ_2	785	769	801
Gypsum specific heat $[kJ/kg \cdot K]$	θ_3	830	623	1038
Brick conductivity $[W/m \cdot K]$	θ_4	0.675	0.608	0.743
Brick density $[kg/m]$	θ_5	1602	1570	1634
Brick specific heat $[kJ/kg \cdot K]$	θ_6	790	593	988
Gypsum board conductivity $[W/m \cdot K]$	θ_7	0.16	0.144	0.176
Gypsum board density $[kg/m]$	θ_8	800	784	816
Gypsum board specific heat $[kJ/kg \cdot K]$	θ_9	1090	818	1363
Glazing Thermal Properties:				
Window U-value $[W/m \cdot K]$	θ_{10}	5.778	5.49	6.067
Window SHGC $[-]$	θ_{11}	0.862	0.82	0.91
Ventilation:				
Infiltration rate $[ACH]$	θ_{12}	2.0	1.6	2.4
Internal loads:				
Occupancy density multiplier $[-]$	θ_{13}	1.0	0.8	1.2
People metabolic rate $[W]$	θ_{14}	150	130	250
Equipment power density multiplier $[-]$	θ_{15}	1.0	0.8	1.2
Lighting power density multiplier $[-]$	θ_{16}	1.0	0.8	1.2
Cooling System:				
Chiller COP $[-]$	θ_{17}	3.35	2.68	4.02

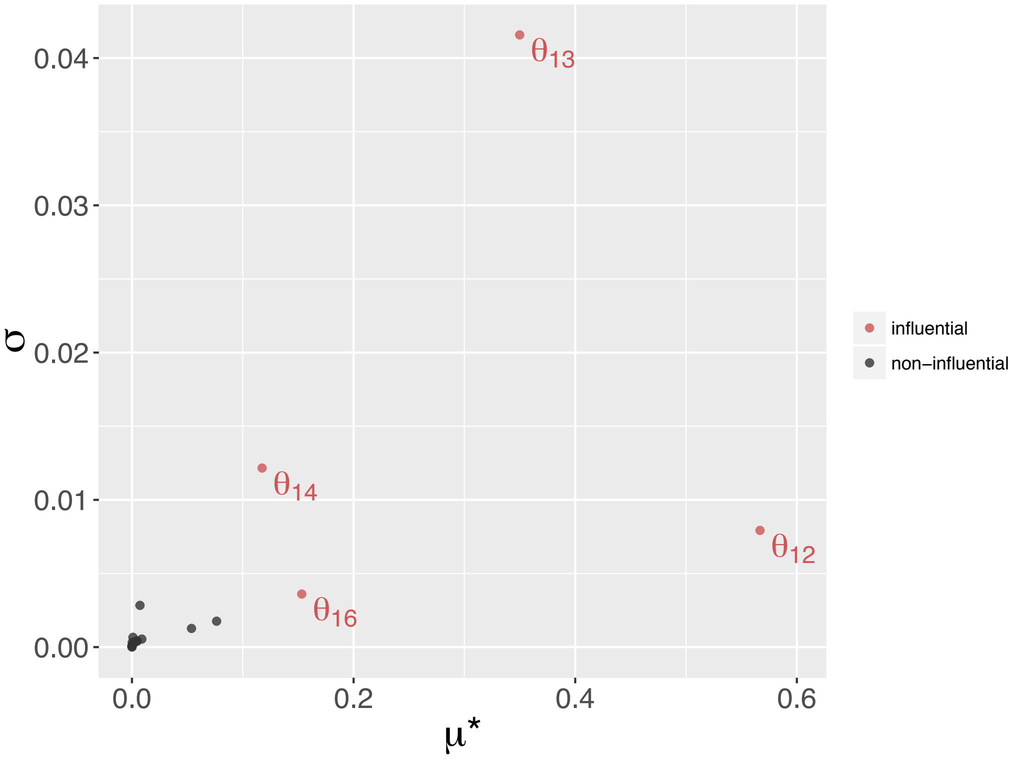

Figure 3.16: Case study 3: graphical plot of sensitive measures μ^* and σ for parameters θ_1 to θ_{17} (Table 3.8). The closer the parameters are to the upper right, the more sensitive the parameter. Parameters close to the bottom left are non-influential parameters.

From Figure 3.16, it can be seen that the parameters infiltration rate θ_{12}, occupancy density multiplier θ_{13}, people metabolic rate θ_{14} and lighting power density multiplier θ_{16} are separate from the remaining parameters that have μ^* and σ very close to zero. Parameters θ_{12}, θ_{14} and θ_{16} have σ that is close to zero. On the contrary, θ_{13} has μ^* that is smaller than that of θ_{12} but σ that is substantially larger than all the other parameters. Considering both μ^* and σ, we conclude that parameters θ_{12}, θ_{13}, θ_{14} and θ_{16} are important, and that θ_{12} have effects that involve either curvature or interactions with other parameters.

Based on the sensitivity analysis, only parameters θ_{12}, θ_{13}, θ_{14} and θ_{16} would be used for the Bayesian calibration of the EnergyPlus model. Note that we use the notation t to denote the calibration parameters, i.e., the influential parameters that were selected from the set of uncertain parameters θ. Therefore, the inputs and output used for the Bayesian calibration of this

EnergyPlus model can be summarized as:

- Observed output $y(x)$: Measured whole building cooling energy consumption [kWh].
- Simulation output $\eta(x, \theta)$: Predicted whole building cooling energy consumption [kWh].
- Observed inputs x from local weather station:
  (a) x_1: Outdoor dry-bulb air temperature [$^\circ C$].
  (b) x_2: Outdoor relative humidity [$\%$].
- Calibration parameters θ:
  (a) $t_1 = \theta_{12}$: Infiltration rate [ACH].
  (b) $t_2 = \theta_{13}$: Occupancy density multiplier [$-$].
  (c) $t_3 = \theta_{14}$: People metabolic rate [W].
  (d) $t_4 = \theta_{16}$: Lighting power density multiplier [$-$].

3.3.3 Bayesian calibration

Bayesian calibration as described in Chapter 2.2.2 was used for the calibration. Data collection took place between January 1, 2013 and December 31, 2013. After removing erroneous data and all instances for which the outcome or any of the inputs are missing, the dataset contained 242 samples. Of the data collected, 50% (121 samples) were used as a test hold-out dataset and the remaining 50% (121 samples) were used for calibrating the model. Daily data was used for the calibration.

To reduce computation cost, 2 strategies were employed that include using information theory to reduce the number of samples (Chapter 2.4.2) and using NUTS (a more efficient MCMC algorithm) to explore the posterior distribution (Chapter 2.4.3). The observed input factors x is two dimensional with components corresponding to: 1) x_1: outdoor dry-bulb air temperature; and 2) x_2: outdoor relative humidity. The observed output $y(x)$ is the measured whole building cooling energy consumption. Therefore the experimental design corresponding to the field observations is a dataset $D^F = \begin{bmatrix} y & x_1 & x_2 \end{bmatrix}$ where $D^F \in \mathbb{R}^{121 \times 3}$.

To learn about the calibration parameters t, $m = 200$ EnergyPlus simulations were run at different combinations of (x, t). Maximin LHS (Stein, 1987) was used to generate values for the calibration parameters t_1, t_2, t_3 and t_4. Running each of the 200 simulations at the same input factors x generates $121 \times 200 = 24200$ samples. Therefore, the experimental design for the simulations is a dataset $D^S = \begin{bmatrix} \eta(x,t) & x_1 & x_2 & t_1 & t_2 & t_3 & t_4 \end{bmatrix}$ where $D^S \in \mathbb{R}^{24200 \times 7}$.

To sample a representative subset D^F_{sub} from the field dataset D^F and a representative subset D^S_{sub} from the simulation dataset D^S, random samples of varying sizes were generated and their corresponding sample quality computed (Chapter 2.4.2). Figure 3.17 shows the relationship between sample size and sample quality. Taking into consideration both sample quality and computation cost, a sample size of 16 for D^F_{sub} and 640 for D^S_{sub} was used. Then, D^F_{sub} and D^S_{sub} was used for the calibration of the building energy model instead of the entire dataset D^F and D^S.

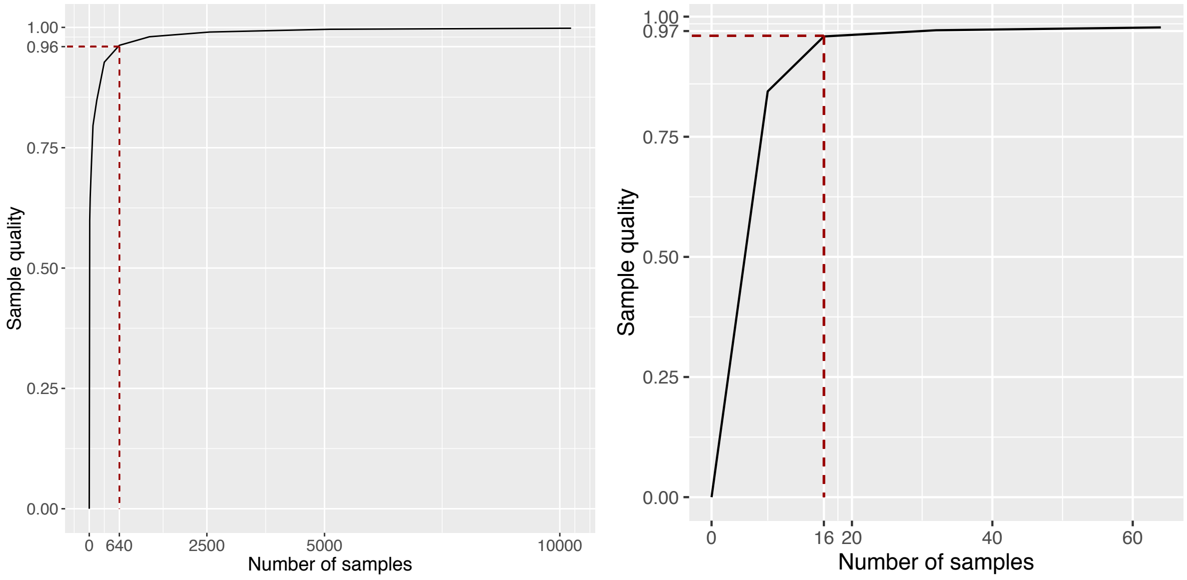

Figure 3.17: Case study 3: number of samples against sample quality for different subsets of simulation data D^S (left plot) and field data D^F (right plot). 640 random samples from simulation data D^S and 16 random samples from field data D^F is sufficient to represent the whole dataset.

Using the formulation by Higdon et al. (2004), D^F_{sub} and D^S_{sub} were combined in a Gaussian process model. Following Chapter 2.2.2, the GP model for $\eta(.,.)$ was specified using a mean function that returns the zero vector and a covariance function of the form:

$$\begin{aligned} &Cov((x,t),(x',t')) = \\ &\frac{1}{\lambda_\eta} exp\bigg\{ -\sum_{j=1}^{2} \beta^\eta_j |x_{ij} - x'_{ij}|^2 - \sum_{k=1}^{4} \beta^\eta_{p+k} |t_{ik} - t'_{ik}|^2 \bigg\} \end{aligned} \tag{3.20}$$

The GP model for the discrepancy term $\delta(x)$ was also specified with a mean function that returns the zero vector and a covariance function of the form:

$$Cov(x,x') = \frac{1}{\lambda_\delta} exp\bigg\{ -\sum_{j=1}^{2} \beta^\delta_k |x_{ij} - x'_{ij}|^2 \bigg\} \tag{3.21}$$

Given that there are 4 calibration parameters and 2 input factors, parameters of the posterior which need to be estimated is 17 dimensional and include: 1) the calibration parameters $t_1, ..., t_4$; 2) the correlation hyperparameters of the GP model $\beta^\eta_1, ..., \beta^\eta_6, \beta^\delta_1, \beta^\delta_2$; and 3) the variance hyperparameters of the GP model λ_η, λ_δ and λ_ϵ. NUTS, an extension of HMC was used to explore the posterior distributions. Four independent chains of 500 iterations per chain were run. To be conservative, the first 250 iterations (50%) values were discarded as warmup/burn-in to reduce the influence of starting values.

Figure 3.18 shows the posterior distribution of the calibration parameters t. The posterior distribution for the calibration parameters $t_1, ..., t_4$ appear to be uniformly distributed across the range defined by the prior probability distribution (Table 3.9), suggesting that the uncertainty due to the calibration parameters is still relatively large given the data. In addition, the 95% confidence intervals for the calibration parameters have lower and upper bounds that are very close to the minimum and maximum values specified for the uniform prior probability distributions.

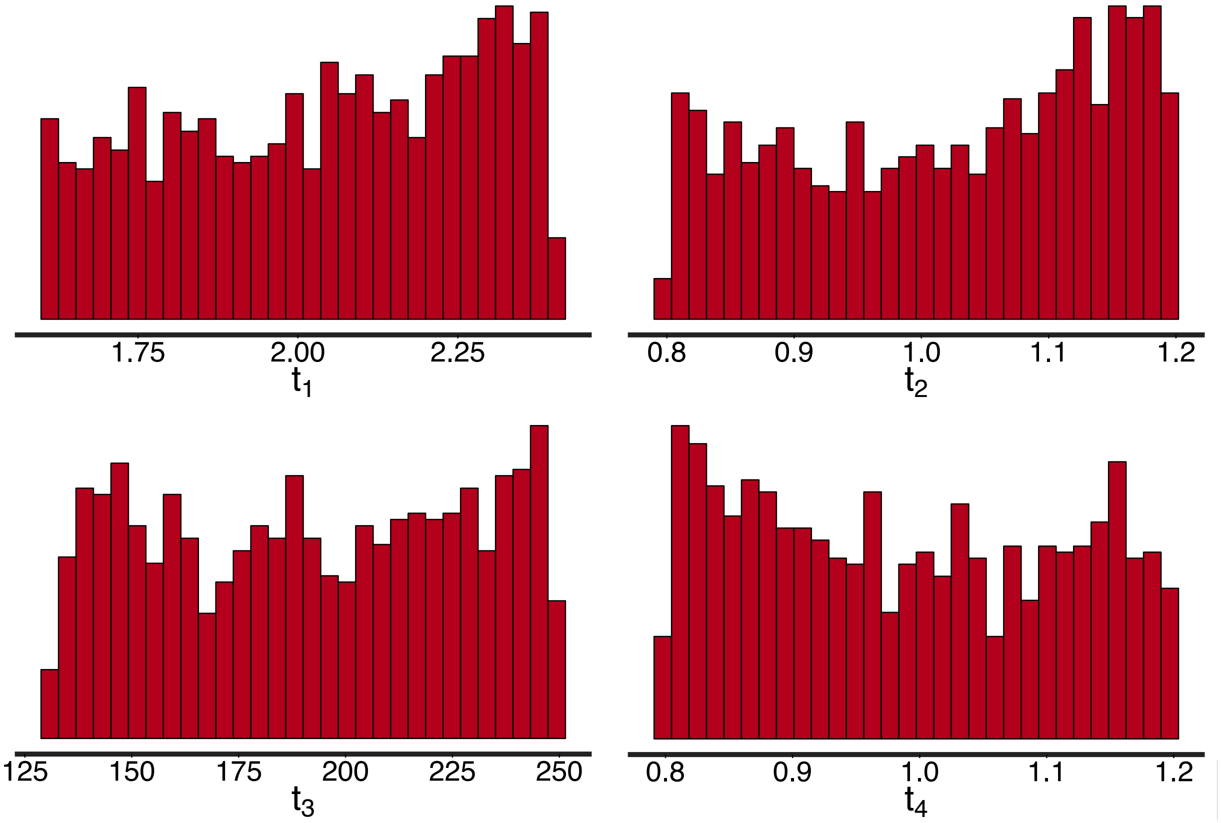

Figure 3.18: Case study 3: posterior distribution (excluding warmup) of calibration parameters: t_1 infiltration rate $[ACH]$, t_2 occupancy density multiplier $[-]$, t_3 people metabolic rate $[W]$, t_4 lighting power density multiplier $[-]$.

Table 3.9: Case study 3: Prior distribution and summary statistics for posterior distribution of calibration parameters.

Parameter (units)	Prior	Posterior			
		mean	**2.5 Percentile**	**50 Percentile**	**97.5 Percentile**
$t_1\ (-)$	$\mathcal{U}(1.60, 2.40)$	2.04	1.62	2.06	2.38
$t_2\ (-)$	$\mathcal{U}(0.80, 1.20)$	1.02	0.81	1.04	1.19
$t_3\ (W)$	$\mathcal{U}(130, 250)$	192	135	192	247
$t_4\ (-)$	$\mathcal{U}(0.80, 1.20)$	0.99	0.81	0.98	1.19

3.3.4 Model evaluation

Similar to case studies 1 and 2, prediction accuracy of the calibrated model was assessed using a visual comparison of the calibrated predictions against a hold-out test dataset (Figure 3.19). Of the 242 observations that were collected, 50% of the data (121 samples) was withheld and not used for the calibration. Figure 3.19 shows the hold-out samples and how they compare with the 95% confidence interval of the predictions. Overall, it can be observed that most of the observed values falls within the 95% confidence interval. However, the figure also shows that compared to case studies 1 (Figure 3.4) and 2 (Figure 3.12), predictions for case study 3 is not as precise and has a larger 95% confidence interval. A histogram of the residuals for case study 3 (Figure 3.20) also has a wider deviation as compared to case studies 1 and 2. Nonetheless, the histogram shows that the residuals are centered around zero and appears to be normally distributed, with most of the residuals within $\pm 2\sigma_y$. Over all 121 hold-out samples, CVRMSE and NMBE were 11.9% and 0.7% respectively, meeting the acceptable thresholds for hourly calibration data set by various standards and guidelines (See Table 1.1). Since the standards and guidelines do not provide any criteria for evaluating daily calibration data, the stricter monthly error criteria for assessing the calibrated model was used.

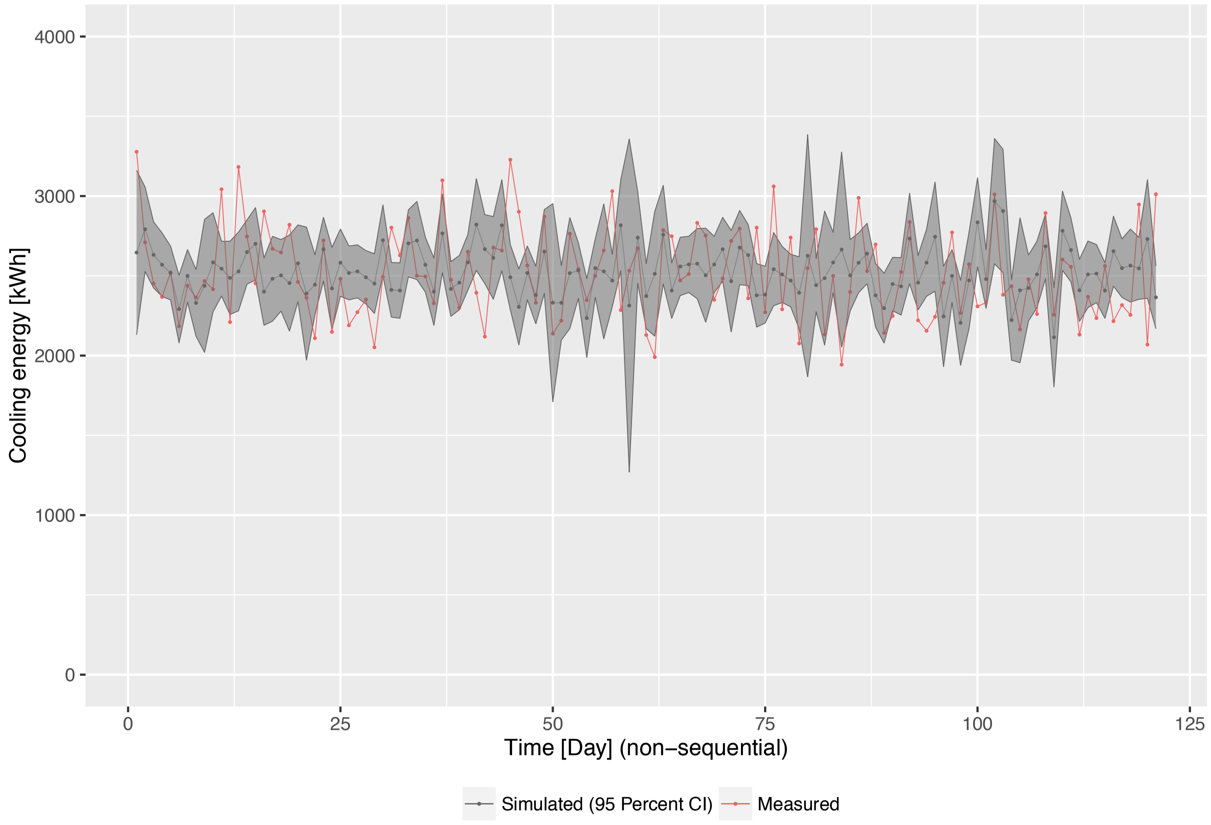

Figure 3.19: Case study 3: comparing 95% confidence interval calibrated predictions with field observations from the hold-out test dataset.

Table 3.10: Case study 3: CVRMSE and NMBE computed with posterior mean estimates with and without the discrepancy term $\delta(x)$.

Statistical Formulation	**CVRMSE (%)**	**NMBE (%)**
$\eta(x, \theta) + \epsilon(x)$	11.9	0.5
$\eta(x, \theta) + \delta(x) + \epsilon(x)$	11.9	0.7

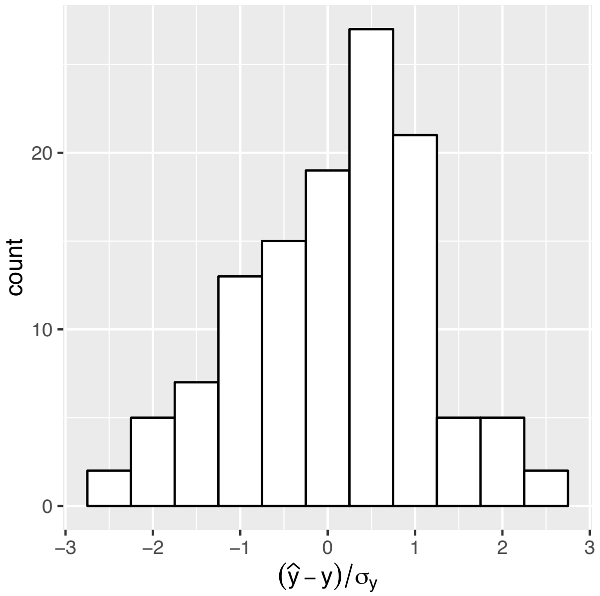

Figure 3.20: Case study 3: histogram of residuals (test data) standardized by standard deviation of observed output.

Figure 3.21 shows the resulting mean predictions[3] with and without the discrepancy term $\delta(x)$, as well as how they compare with the field observations $y(x)$. To gain a better understanding of the simulator's prediction, Figure 3.21 shows how the predictions and discrepancies varies against different input factors including outdoor dry-bulb temperature and outdoor relative humidity (RH). It can be seen that the discrepancy term $\delta(x)$ is close to zero across different values of dry-bulb temperature and relative humidity. Since the discrepancy is nearly zero, the

[3] The mean posterior predictions are computed by taking the mean of the predictions at input settings given by the hold-out test dataset

model predictions $\eta(x, t)$ are very similar to the calibrated predictions $\eta(x, t) + \delta(x)$. This is also illustrated by a CVRMSE and NMBE that is very similar (Table 3.19). Although the discrepancy term is very small (approximately zero), overall uncertainty of the calibration parameters remains about the same, with the prior probability distribution and the 95% confidence interval of the posterior distribution having approximately the same range (Table 3.9). This suggests that the data is non-informative about the calibration parameters and cautions users against interpreting a small discrepancy term as greater confidence in the posterior distribution of the calibration parameters.

Trace plots and $\hat{R}$ of the calibration parameters ($t_1, ..., t_4$), correlation hyperparameters ($\beta_1^\eta, ..., \beta_6^\eta$ and $\beta_1^\delta, \beta_2^\delta$), and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) were used to assess convergence. See Chapter 2.4.4 for a description of both metrics. From Figures A.5and A.6, it can be seen that all components of the posterior distribution are well mixed and have converged to a common stationary distribution. $\hat{R}$ is also within 1 ± 0.1 for all calibration parameters and hyperparameters of the GP model.

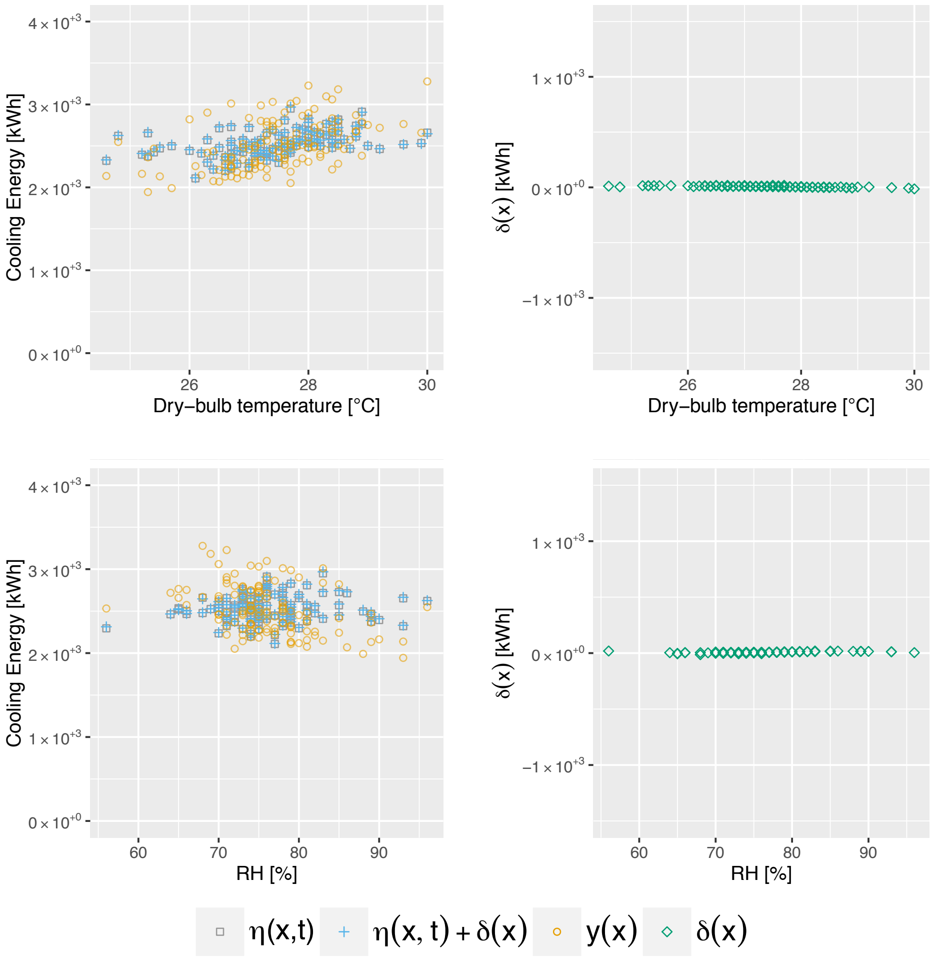

Figure 3.21: Case study 3: posterior mean estimates for the field observations $y(x)$, the calibrated simulator output $\eta(x,t)$, the discrepancy term $\delta(x)$ and the calibrated prediction with discrepancy term $\eta(x,t) + \delta(x)$.

Chapter 4

Comparison of MCMC Algorithms

4.1 Overview

In this chapter, three MCMC algorithms were empirically compared. This include comparing the effectiveness of NUTS (a variant of HMC) that was outlined in Chapter 2.4.3 with the more commonly used random-walk Metropolis (RWM) and Gibbs sampling (both described in Chapter 2.2.3). The comparison was carried out using the three case studies that was described in Chapter 3. The algorithms were compared using trace plots and the Gelman-Rubin statistics $\hat{R}$ (Gelman et al., 2014) to check for mixing and convergence to the posterior distribution.

In an earlier study by Chong and Lam (2017), using Gelman-Rubin statistics $\hat{R}$ to assess convergence, it was shown that for the same number of iterations, NUTS was more effective in generating samples from the posterior distribution as compared to RWM and Gibbs sampling. In addition, it was shown in the same study that NUTS was able to achieve adequate convergence with as little as 500 iterations. Therefore, for this study, NUTS was run with 500 iterations. To be conservative, the comparison is made based on the number of gradient evaluations instead of the number of iterations. In other words, if there are 10000 gradient evaluations when 500 iterations are run, RWM and Gibbs sampling would be run with 10000 iterations. This is because each iteration of NUTS contains L leapfrog steps and gradient is evaluated once during each leapfrog step (See Listing 2.1 in Chapter 2.4.3). According to Hoffman and Gelman (2014), the computation cost of an MCMC algorithm is dominated by the number of likelihood or gradient evaluations. Therefore, number of gradient evaluations is used as a normalized comparison of computation cost. Table 4.1 shows the number of iterations run for each algorithm with reference to case studies 1, 2 and 3 (see Chapter 3 for a detailed description of each case study).

Table 4.1: Number of iterations run for each algorithm for different case studies

	NUTS	RWM	Gibbs
Case Study 1	500	9000	9000
Case Study 2	500	10000	10000
Case Study 3	500	12000	12000

For all three case studies, NUTS was run for 500 iterations. This corresponds to about 9000, 10000, and 12000 gradient evaluations for case studies 1, 2 and 3 respectively. Consequently, RWM and Gibbs sampling was run for 9000, 10000, and 12000 iterations for case studies 1, 2 and 3 respectively. An additional tuning of the acceptance ratio is required for RWM. It is generally accepted that the optimal acceptance rate of the Metropolis algorithm is about 23% (Gelman et al., 1996). For RWM, a normal proposal or jumping distribution was used. Then, its variance is tuned until an acceptance rate of between 20% and 25% was achieved. Gibbs sampling was run for the same number of iterations as RWM although this typically required more time to run since each iteration of the Gibbs sampler cycles through each parameter and samples it from its conditional distribution while holding the other parameters fixed (Equation 2.11). Nonetheless, following Hoffman and Gelman (2014), the Gibbs sampler was run for the same number of iterations as RWM since there are many variants of Gibbs sampling where the alternating conditional sampling could be done more efficiently.

To assess convergence, four independent chains was run for each MCMC algorithm. As a conservative choice, the first 50% iterations were discarded to diminish the influence of the starting values. $\hat{R}$ and trace plots of the calibration parameters t, variance hyperparameters λ_η, λ_δ and λ_ϵ, and correlation hyperparameters β^η and β^δ is then used to assess convergence.

4.2 Assessing convergence

Tables 4.2, 4.3 and 4.4 show the comparison of $\hat{R}$ values with different MCMC algorithms for case studies 1, 2, and 3 respectively. From the tables, it can be seen that running 500 iterations using NUTS is sufficient to achieve adequate convergence, with $\hat{R}$ being within 1 ± 0.1 for all parameters of the posterior distribution. This observation is consistent across all three case studies. The low number of iterations can be attributed to rapid mixing as illustrated by the trace plots in Appendix A.1.

On the contrary, $\hat{R}$ values computed with samples generated by RWM indicates a lack of convergence. In all three case studies, $\hat{R}$ exceeds the threshold of 1.1 for all 15 parameters (Tables 4.2, 4.3 and 4.4). The trace plots (Appendix A.2) also show that the different MCMC chains are not traversing the same distribution and that more iterations is needed before convergence is achieved. The trace plots also show poor mixing, indicating that more tuning of the proposal distribution is necessary for better mixing and thus faster convergence.

Gibbs sampling performs substantially better than RWM but not as well as NUTS. Tables 4.2, 4.3 and 4.4 show that, compared to RWM, significantly less parameters have $\hat{R}$ exceeding the threshold of 1 ± 0.1. In case study 2, only one parameter (λ_ϵ) has $\hat{R}$ exceeding the specified limit (Table 4.3). This increases to four parameters (β_2^η, β_4^η, β_5^η, β_6^η, λ_η, and λ_ϵ) in case study 3 (Table 4.4). The trace plots for case study 3 (Figures A.17 and A.18) also show convergence issues. When looked at separately, all four chains appear stable and convergence is seemingly achieved. However, looking at all chains together show that one chain (yellow chain) is traversing a different distribution. This is particularly obvious in the trace plot for λ_η (Figure A.17). In addition, the trace plots for λ_ϵ (Figures A.13, A.15 and A.17) show that it is moving through the parameter space relatively slowly, suggesting poor mixing and that a more iterations is needed before adequate convergence can be achieved.

Table 4.2: $\hat{R}$ of calibration parameters and GP hyperparameters with different MCMC algorithms for case study 1. Values exceeding 1 ± 0.1 are highlighted in red.

Components of Posterior Distribution	**RWM**	**Gibbs Sampling**	**NUTS (HMC)**
Calibration Parameters			
t_1	20.9	1.00	1.00
t_2	142	1.01	1.00
Correlation Hyperparameters of GP model			
β_1^η	1.98	1.01	1.00
β_2^η	18.1	1.00	1.00
β_3^η	29.5	1.00	1.00
β_4^η	23.4	1.01	1.00
β_5^η	3.70	1.00	1.00
β_6^η	16.2	1.01	1.00
β_1^δ	14.5	1.00	1.00
β_2^δ	8.54	1.00	1.00
β_3^δ	13.9	1.00	1.00
β_4^δ	29.4	1.01	1.00
Variance Hyperparameters of GP model			
λ_η	399	1.00	1.00
λ_δ	1488	2.21	1.00
λ_ϵ	28225	40.6	1.00

Table 4.3: $\hat{R}$ of calibration parameters and GP hyperparameters with different MCMC algorithms for case study 2. Values exceeding 1 ± 0.1 are highlighted in red.

Components of Posterior Distribution	**RWM**	**Gibbs Sampling**	**NUTS (HMC)**
Calibration Parameters			
t_1	37.4	1.00	1.00
t_2	25.4	1.00	1.00
t_3	18.6	1.00	1.00
t_4	30.5	1.01	1.00
t_5	25.6	1.01	1.00
Correlation Hyperparameters of GP model			
β_1^η	1.21	1.03	1.00
β_2^η	9.31	1.00	1.00
β_3^η	6.98	1.04	1.00
β_4^η	10.5	1.03	1.00
β_5^η	20.4	1.05	1.00
β_6^η	5.41	1.02	1.00
β_7^η	10.5	1.02	1.00
β_1^δ	15.0	1.00	1.00
β_2^δ	12.4	1.00	1.00
Variance Hyperparameters of GP model			
λ_η	197	1.00	1.00
λ_δ	3509	1.02	1.00
λ_ϵ	16649	25.3	1.00

Table 4.4: $\hat{R}$ of calibration parameters and GP hyperparameters with different MCMC algorithms for case study 3. Values exceeding 1 ± 0.1 are highlighted in red.

Components of Posterior Distribution	**RWM**	**Gibbs Sampling**	**NUTS (HMC)**
Calibration Parameters			
t_1	10070	1.00	1.00
t_2	17313	1.00	1.00
t_3	4246	1.00	1.00
t_4	5341	1.00	1.00
Correlation Hyperparameters of GP model			
β_1^η	5329	1.06	1.00
β_2^η	1941	10.6	1.00
β_3^η	23.6	1.10	1.00
β_4^η	72.4	1.36	1.00
β_5^η	1573	1.13	1.00
β_6^η	10933	1.23	1.00
β_1^δ	6101	1.03	1.00
β_2^δ	2158	1.02	1.00
Variance Hyperparameters of GP model			
λ_η	66971	2.15	1.00
λ_δ	1176365	1.09	1.00
λ_ϵ	6748269	4.41	1.00

Chapter 5

Conclusion

5.1 Discussion and recommendations

The present study demonstrated a systematic framework (See Chapter 2) for the application of Kenndy and O'Hagen's (2001) formulation to building energy models. This study focuses on the improvement of the current implementation of Bayesian calibration to building energy models. This was achieved using 2 approaches, which include:

- Reducing the number of samples used for the calibration by selecting a representative subset of the entire dataset. This was carried out by considering the sample quality (Equation 2.12) of the subset.
- Using a more efficient MCMC algorithm, the No-U-Turn Sampler (NUTS) (Hoffman and Gelman, 2014) that is an extension to Hamiltonian Monte Carlo (HMC) to explore parameters of the posterior distribution.

In all three case studies, visual plots (Figures 3.4, 3.12 and 3.19) comparing the calibrated predictions and a hold-out test dataset showed good agreement between the predictions and the observations, with most observations being within the 95% confidence intervals of the predictions. CVRMSE and NMBE were also within the acceptable thresholds set by various standards and guidelines (Table 1.1).

This study also investigated the impact of including the discrepancy term $\delta(x)$ in the statistical formulation used for the Bayesian calibration of building energy models. Based on the three case studies, it was found that as intended, the discrepancy term $\delta(x)$ was able account for varying bias as a function of the input factors x. Results from case studies 1 and 2 showed that the discrepancy term can reduce overall bias, leading to a better agreement between the observed values and the calibrated predictions. At the same time, in case study 3 where there is little or no discrepancy between the model output and the observed output, the discrepancy term does not overcompensate and the resulting discrepancy is also approximately zero.

When the discrepancy term $\delta(x)$ is fairly large such as in case study 2, it makes interpretation of the posterior distribution of the calibration parameters difficult. This is because, as mentioned previously, the discrepancy term is an indication of how well the simulation model matches reality. Therefore, a large discrepancy provides caution when interpreting the posterior distribution of the calibration parameters. It is also important to note that a low CVRMSE and NMBE indicates that there is a good fit between the calibrated predictions and the observations, but does not justify that the resultant posterior estimates for the calibration parameters are a good approximation of its true value.

Although a large discrepancy is indicative of large model bias and suggests caution for the interpretation of the calibration parameters, the same is not true about its inverse. A small discrepancy does not suggest greater confidence in the posterior distribution of the calibration parameters. This was illustrated in case study 3. In case study 3, although the discrepancy term is approximately zero, the data is non-informative about the calibration parameters, with the prior and the posterior distribution of the calibration parameters having approximately the same range and distribution.

Using a statistical approach that is based on Kullback-Leibler divergence (Kullback and Leibler, 1951) to measure sample quality (Equation 2.12), it was found that a relatively small number of samples was sufficiently representative of the whole dataset. This suggests that a significantly smaller subset of the entire dataset is adequate for the calibration process, substantially reducing computation cost while still maintaining prediction accuracy. This is because buildings are usually operated the same way, resulting in large redundancies in building data. Figures 5.1 and 5.2 show the data for two different HVAC systems. From Figure 5.2, it can be observed that the chiller is only operating over a very small range of chilled water temperature ($T_{chw,in} \approx 15°C$) and chilled water mass flow rate ($\dot{m}_{chw} \approx 325000\ kg/h$). Similarly, from Figure 5.2 it can be seen that the cooling system only operates at close to maximum flow rate ($V_{frac} \approx 1$) and with a coil load Q_{load} of approximately 800000 W.

This study also compared the effectiveness of using different MCMC algorithms within the proposed method. In particular, NUTS was compared with the more commonly used random-walk Metropolis (RWM) and Gibbs sampling. Two metrics were used to assess convergence and they include the Gelman-Rubin statistics $\hat{R}$ and visual inspection of trace plots to ensure that multiple sequences have mixed and are traversing the same distribution. Compared to RWM and Gibbs sampling, NUTS was found to achieve better convergence for all parameters of the posterior distribution. RWM was unable to achieve adequate convergence for almost all parameters. Trace plots also reveal that a substantially larger number of iterations is required before adequate convergence can be achieved. Overall, Gibbs sampling showed adequate convergence in most parameters. However, in all three case studies, $\hat{R}$ exceeded 1 ± 0.1 for the variance parameters λ_δ and λ_ϵ. Trace plots for λ_ϵ further revealed that it is moving through the parameter space very slowly, indicating poor mixing and that more iterations is required before convergence. Therefore, based on this comparative study, we recommend using NUTS over RWM and Gibbs sampling. This conclusion is consistent with the results of an earlier study (Chong and Lam, 2017), which showed that for the same number of iterations, NUTS was more effective than RWM and Gibbs sampling.

Based on our findings, we recommend the following:

- Use a smaller representative subset (based on sample quality) of the entire dataset when calibrating against daily or hourly data.
- Use NUTS for the MCMC sampling.
- Use results with respect to the posteriors of the calibration parameters and the discrepancy

term to determine if interpreting the posterior distribution of the calibration parameters should be treated with caution.

5.2 Suggestions for future work

While this thesis has demonstrated the application of Bayesian calibration to large datasets, which can result when calibrating against hourly or daily data (large sample sizes) or when calibrating a large number of model parameters (high dimensions), many opportunities for extending the scope of this thesis remain. This section presents some of these opportunities, which include:

- Effectiveness of different emulators: In this thesis, we use a Gaussian process (GP) emulator to map the model's input parameters to it's output. Training of a GP model can be computationally expensive. We overcome this challenge but selecting a representative subset for the calibration. However, it would be useful to test and compare the outcomes with alternative emulators to determine the scenarios when a particular emulator would be appropriate.
- Specification of prior probabilities: Currently, the proposed method relies on the modeler to specify the prior probability distributions for the calibration parameters and hyperparameters that define the GP model. The use of appropriate prior probability distributions for the calibration parameters can help prevent unreasonable parameter values during the calibration process. However, there is currently no guidelines on the specification of prior probability distributions for the hyperparameters that defines the GP model. Therefore, there needs to be increased effort towards the development of guidelines to increase the consistency and transparency of applying Bayesian calibration to building energy models. An extensive study that investigates the effect of prior probabilities on the calibration result would also help guide the calibration process thus informing users regarding the consequence of assigning wrong priors.

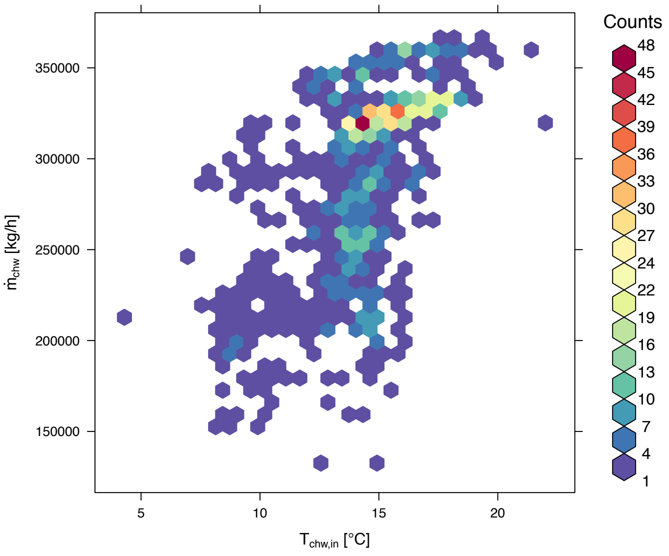

Figure 5.1: 2 dimensional histogram of the entering chilled water temperature $T_{chw,in}$ ($^{\circ}C$) and the chilled water mass flow rate $\dot{m}_{chw}$ (kg/h) of Chiller.

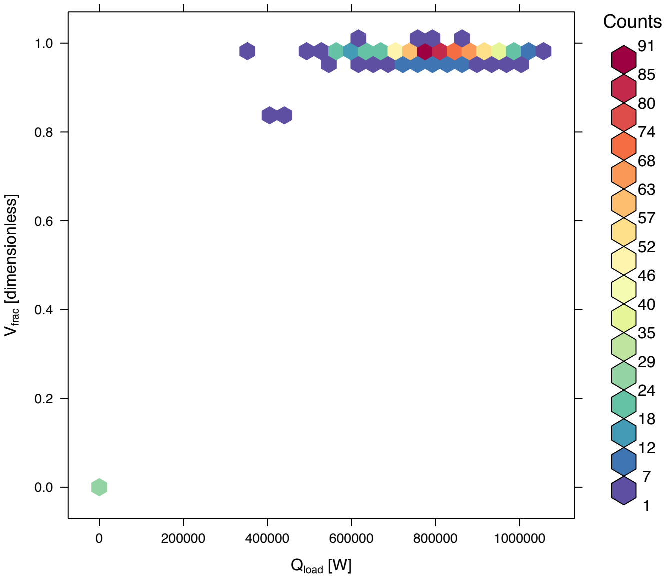

Figure 5.2: 2 dimensional histogram of the cooling coil load Q_{load} (W) and the raction of peak chilled water flow rate ($dimensionless$).

This page is intentionally left blank.

Appendix A

Trace Plots for Case Studies

A.1 No-U-Turn-Sampler (HMC)

A.1.1 Case study 1 - mixed-use building in a college in Singapore

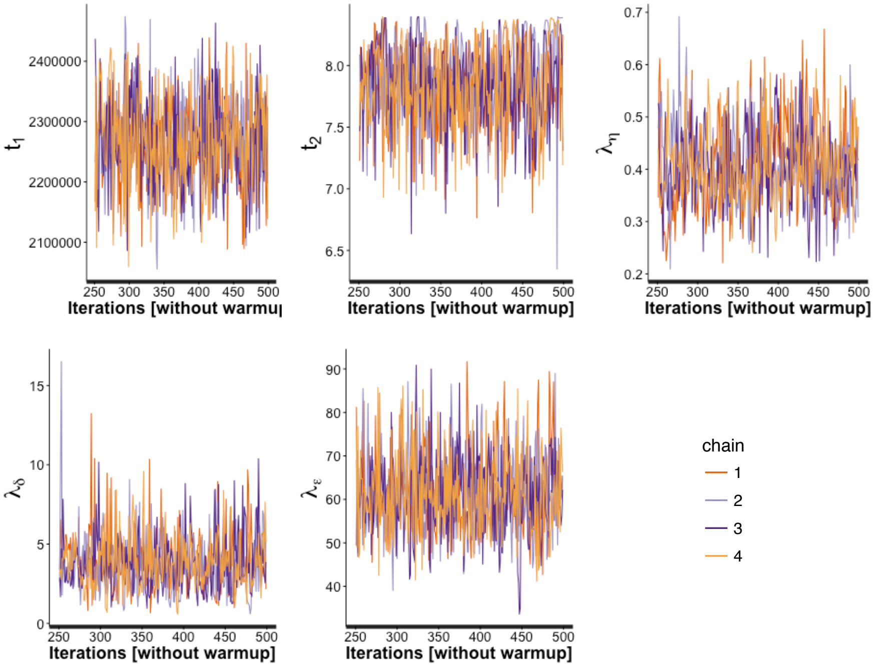

Figure A.1: Case study 1: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using No-U-Turn-Sampler

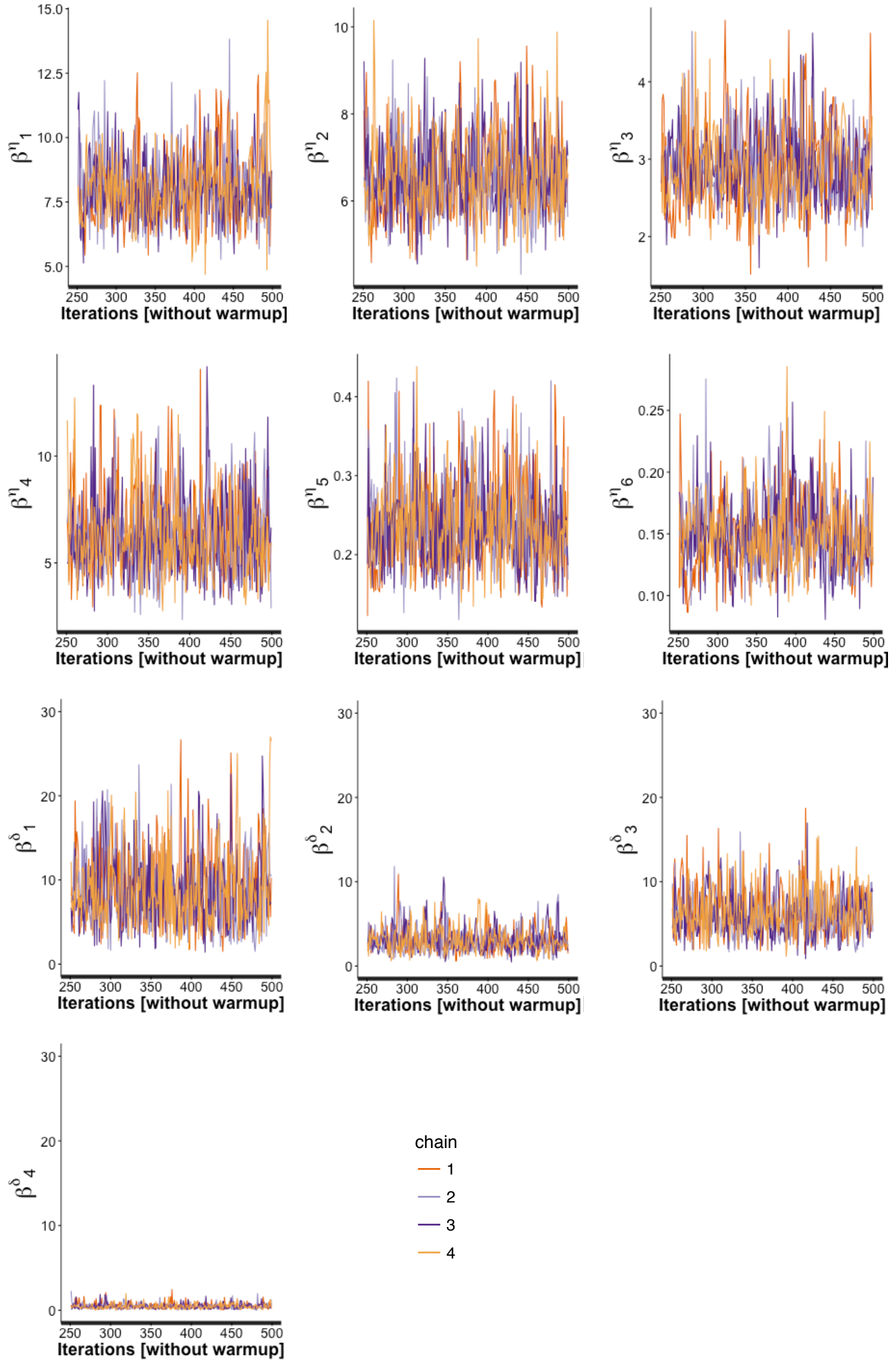

Figure A.2: Case study 1: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using No-U-Turn-Sampler

A.1.2 Case study 2 - office building in Pennsylvania U.S.A

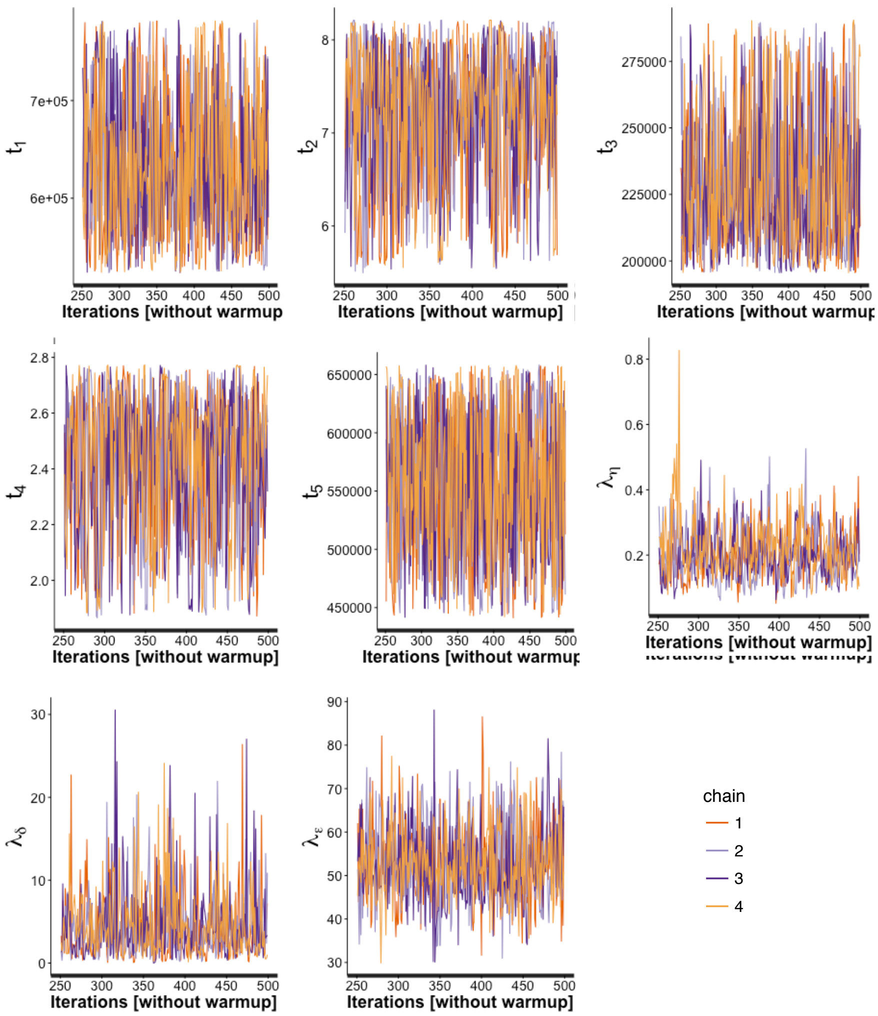

Figure A.3: Case study 2: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using No-U-Turn-Sampler

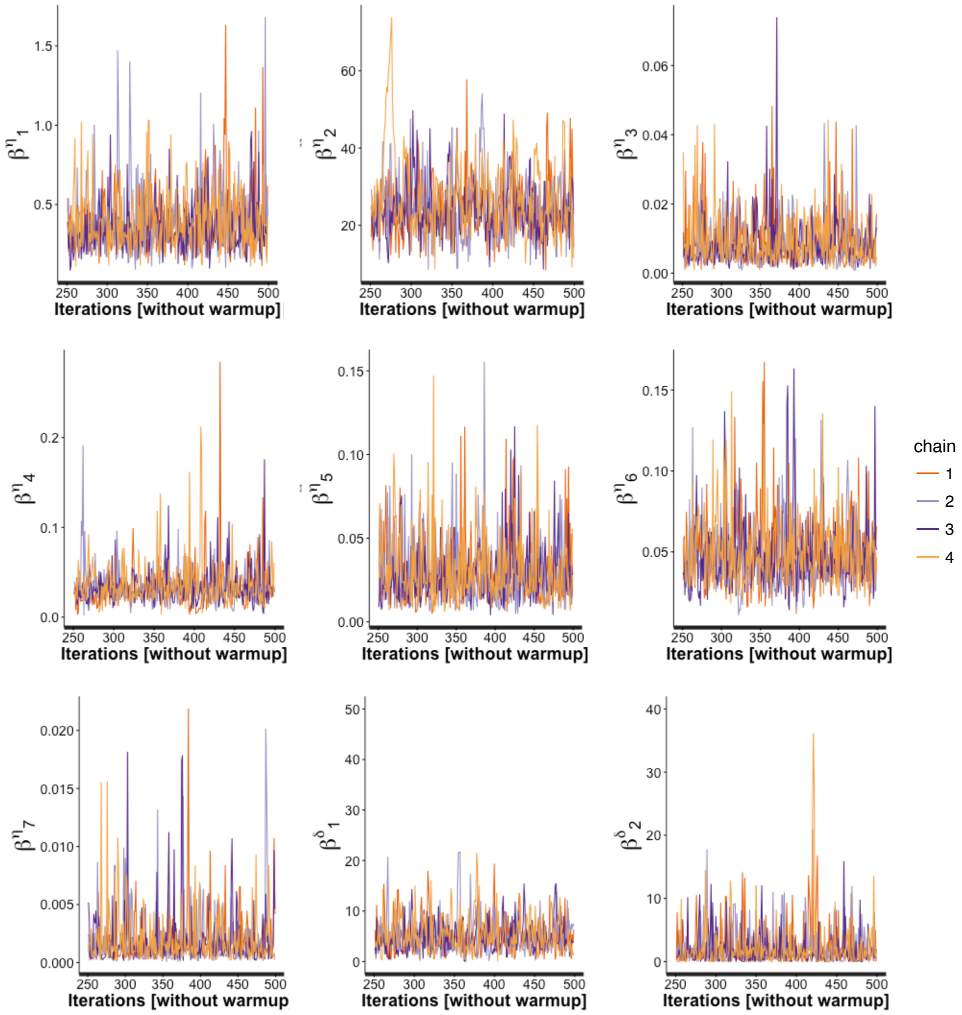

Figure A.4: Case study 2: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using No-U-Turn-Sampler

A.1.3 Case study 3 - mixed-use building on a university campus in Singapore

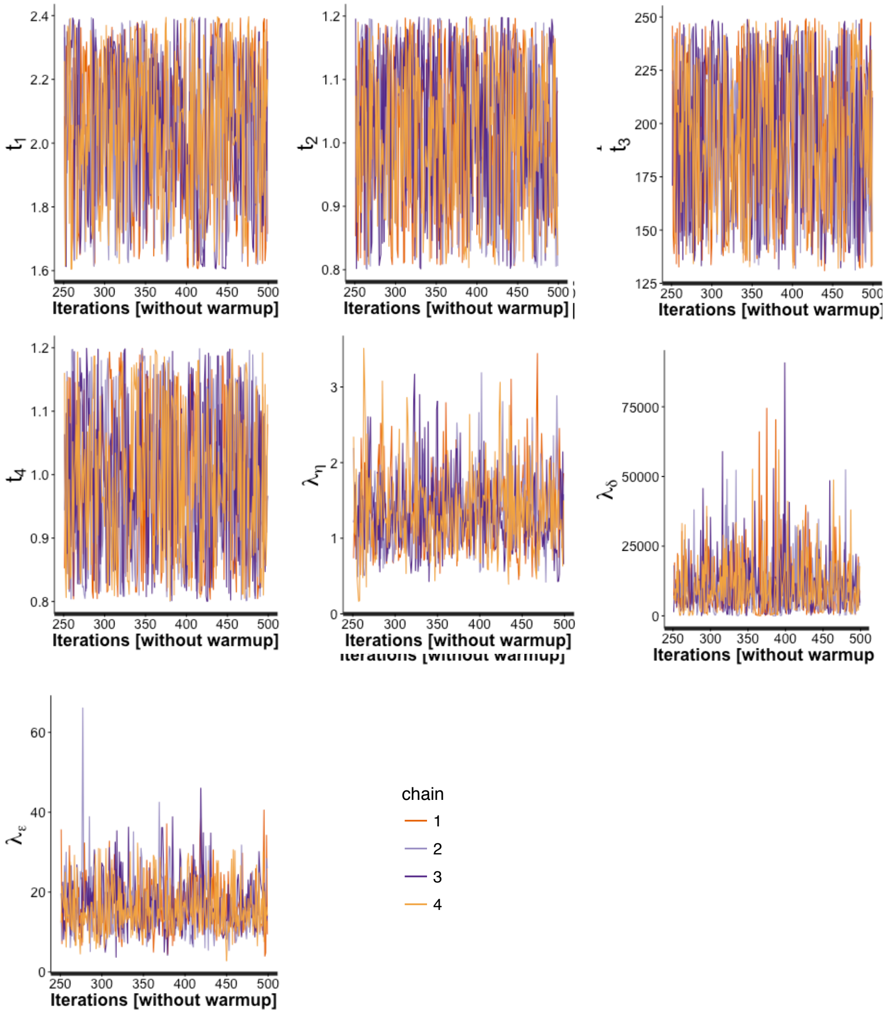

Figure A.5: Case study 3: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using No-U-Turn-Sampler

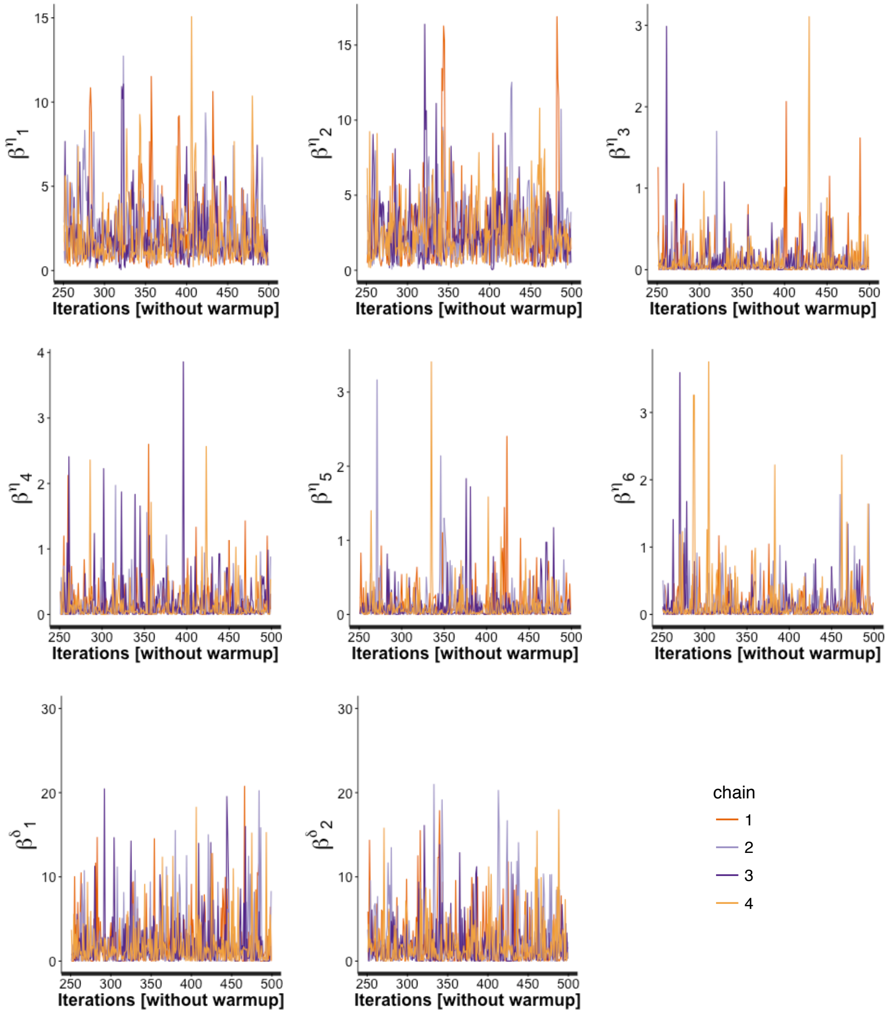

Figure A.6: Case study 3: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using No-U-Turn-Sampler

A.2 Random-Walk Metropolis

A.2.1 Case study 1 - mixed-use building in a college in Singapore

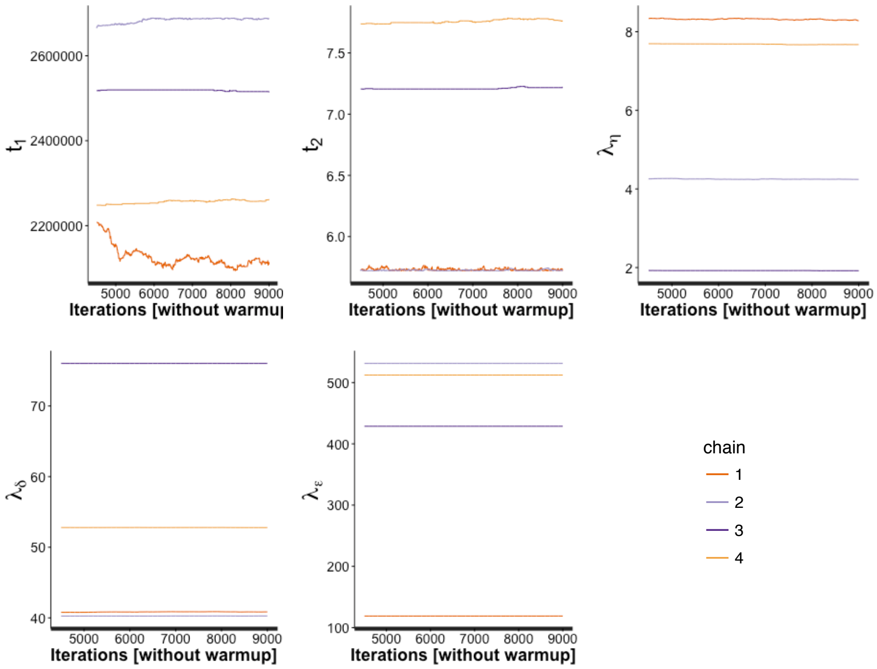

Figure A.7: Case study 1: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using random-walk Metropolis algorithm

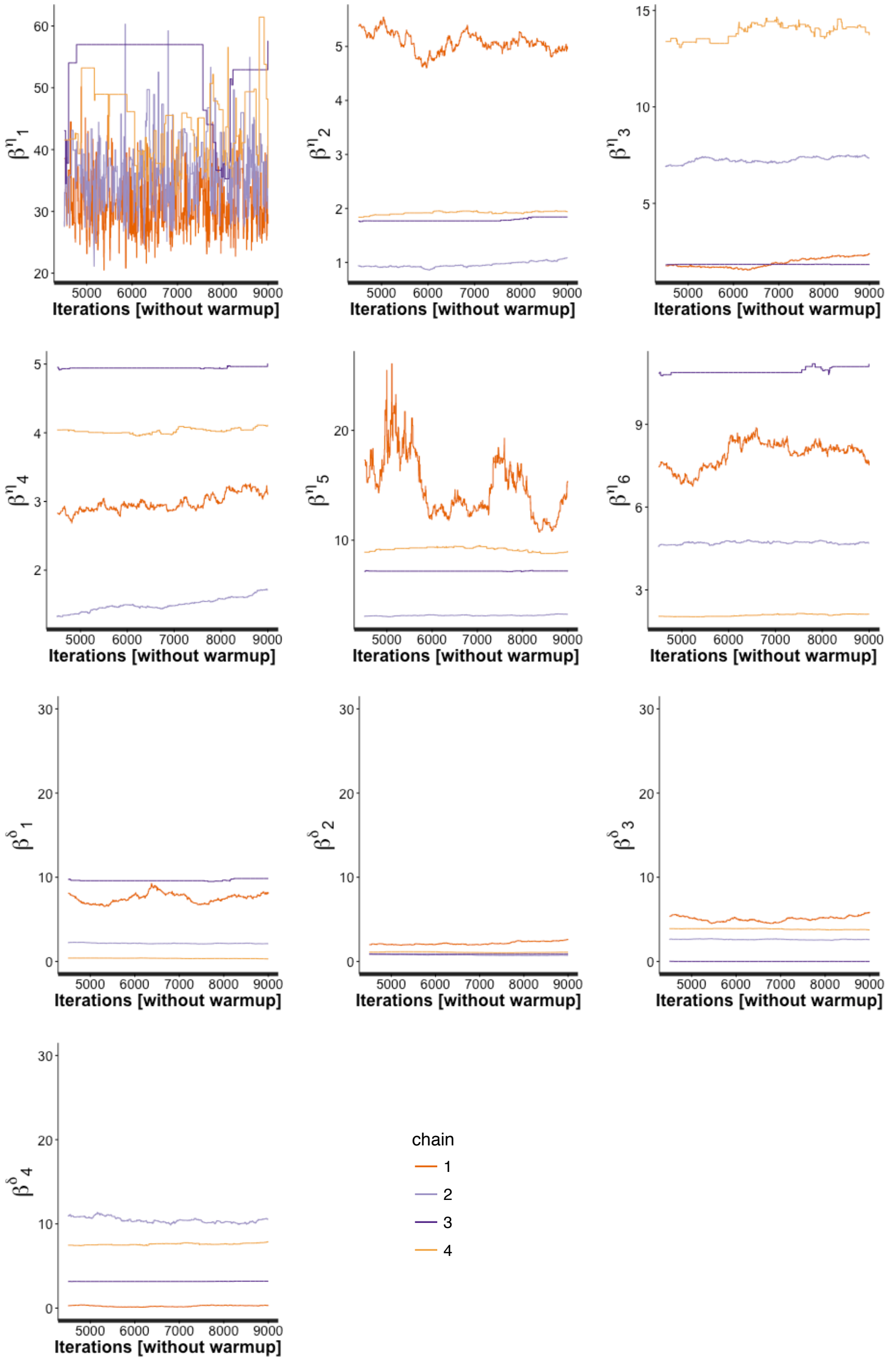

Figure A.8: Case study 1: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using random-walk Metropolis algorithm

A.2.2 Case study 2 - office building in Pennsylvania U.S.A

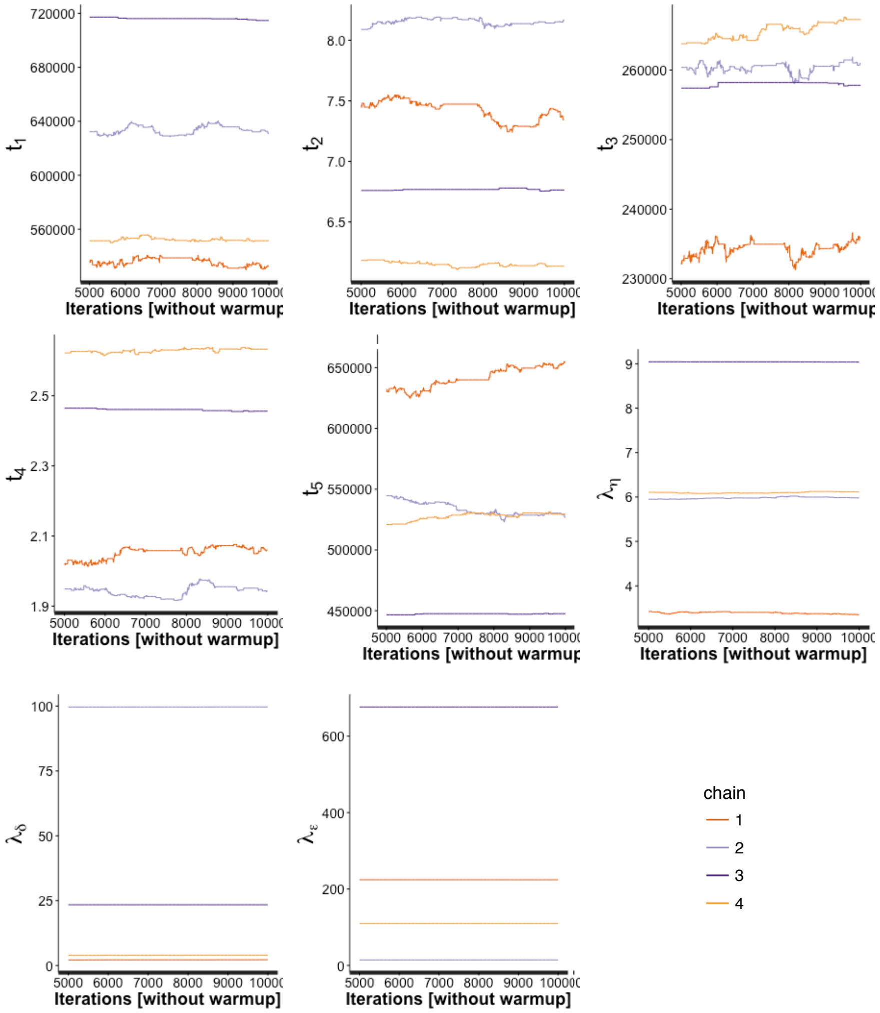

Figure A.9: Case study 2: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using random-walk Metropolis algorithm

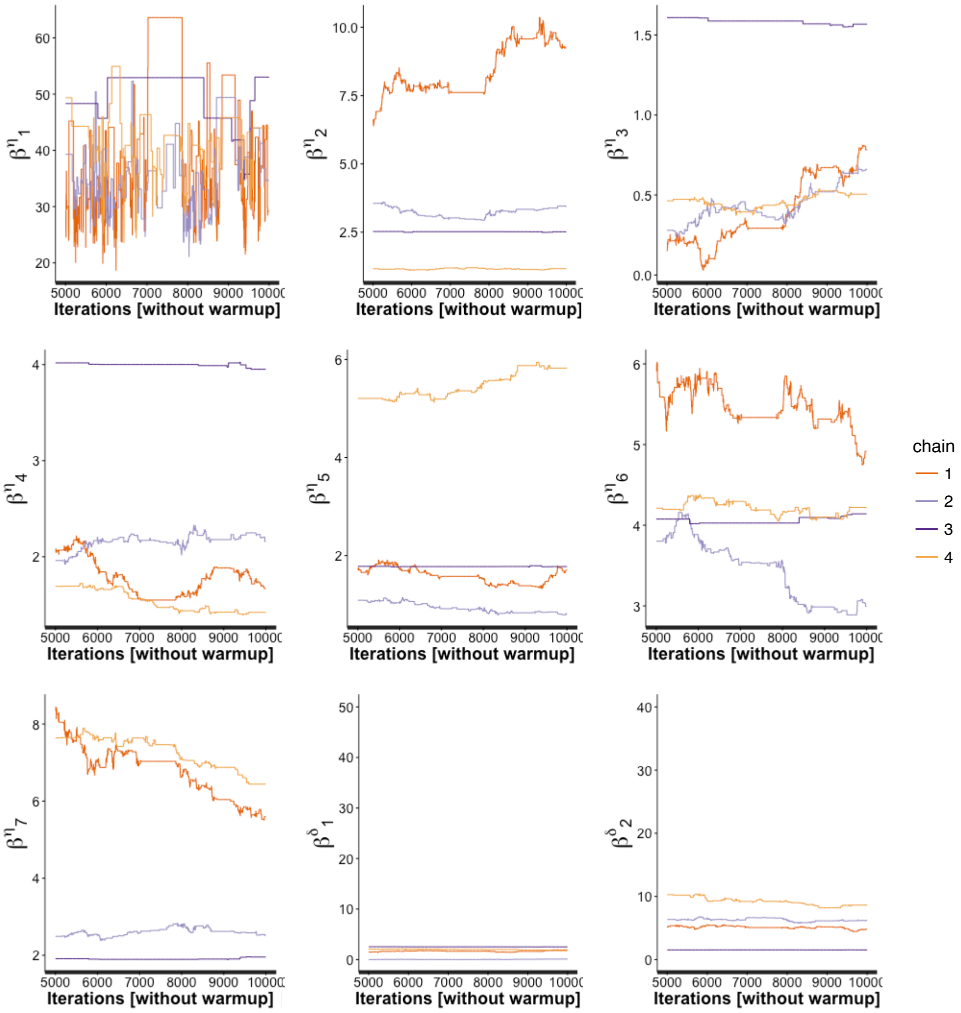

Figure A.10: Case study 2: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using random-walk Metropolis algorithm

A.2.3 Case study 3 - mixed-use building on a university campus in Singapore

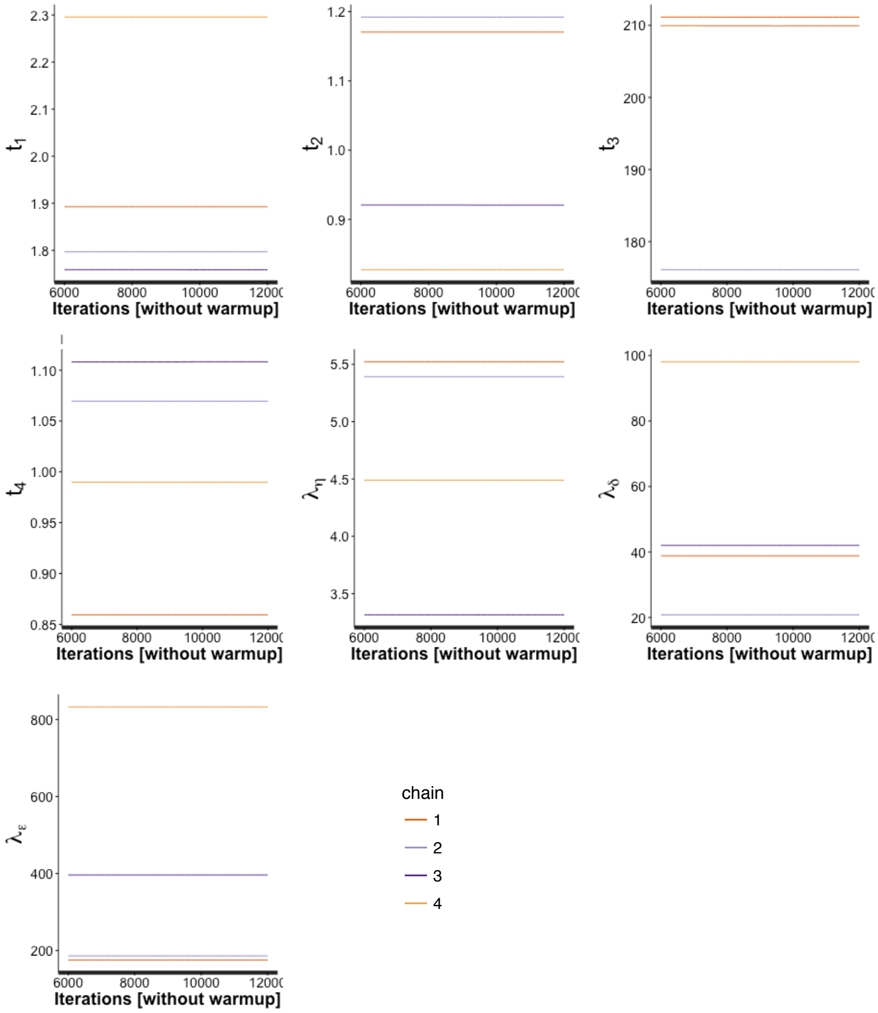

Figure A.11: Case study 3: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using random-walk Metropolis algorithm

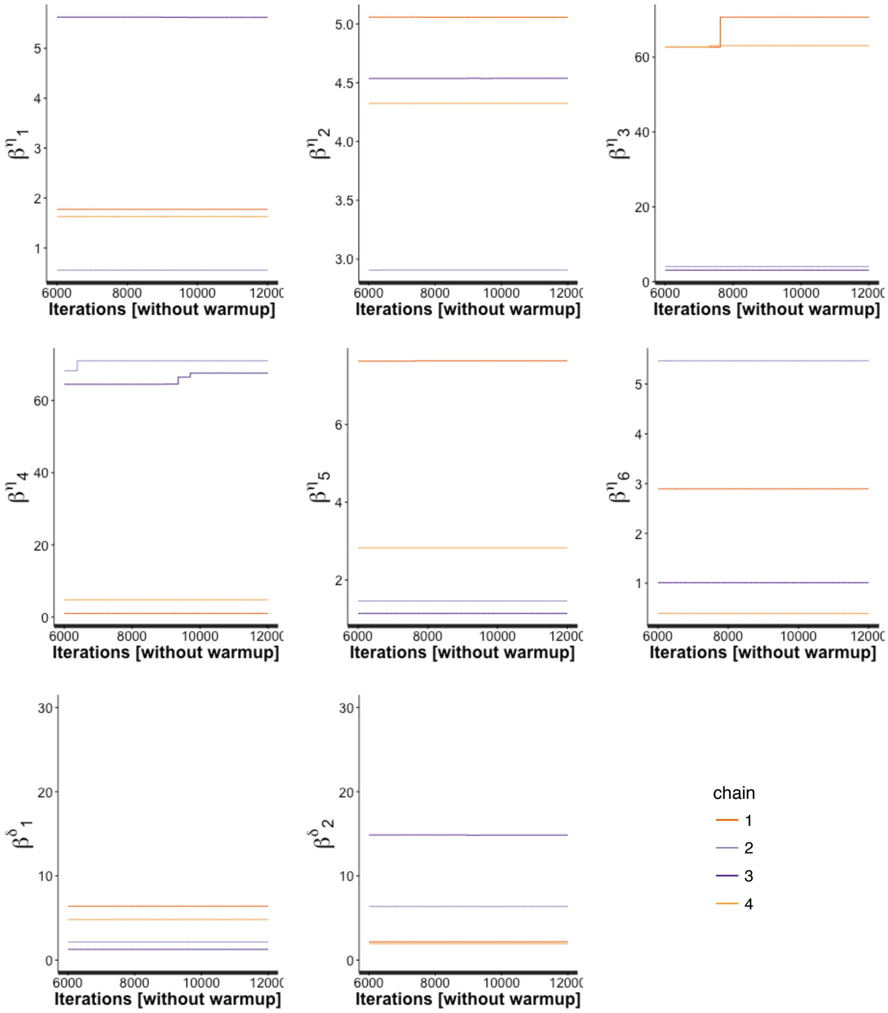

Figure A.12: Case study 3: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using random-walk Metropolis algorithm

A.3 Gibbs Sampling

A.3.1 Case study 1 - mixed-use building in a college in Singapore

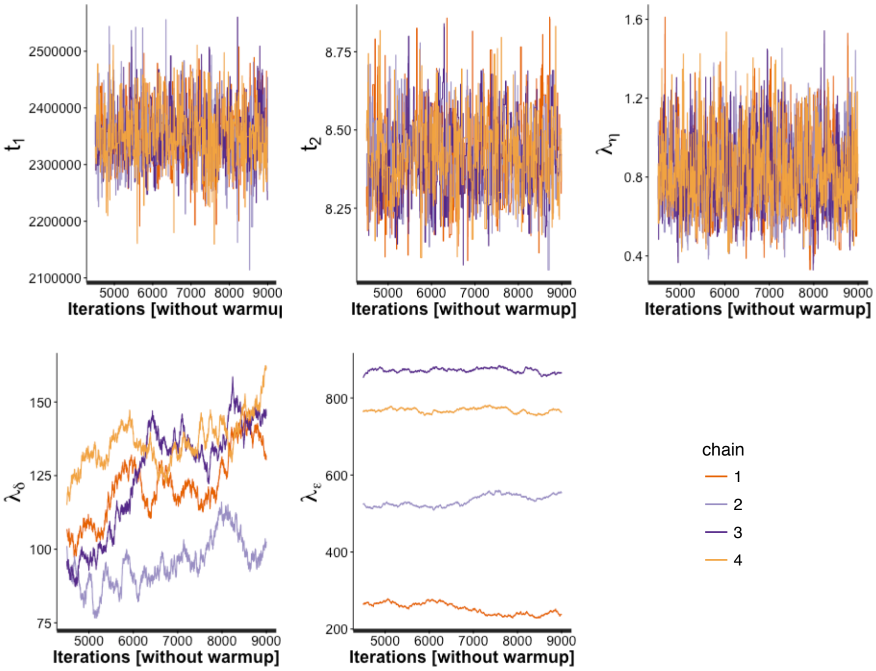

Figure A.13: Case study 1: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using Gibbs sampling algorithm

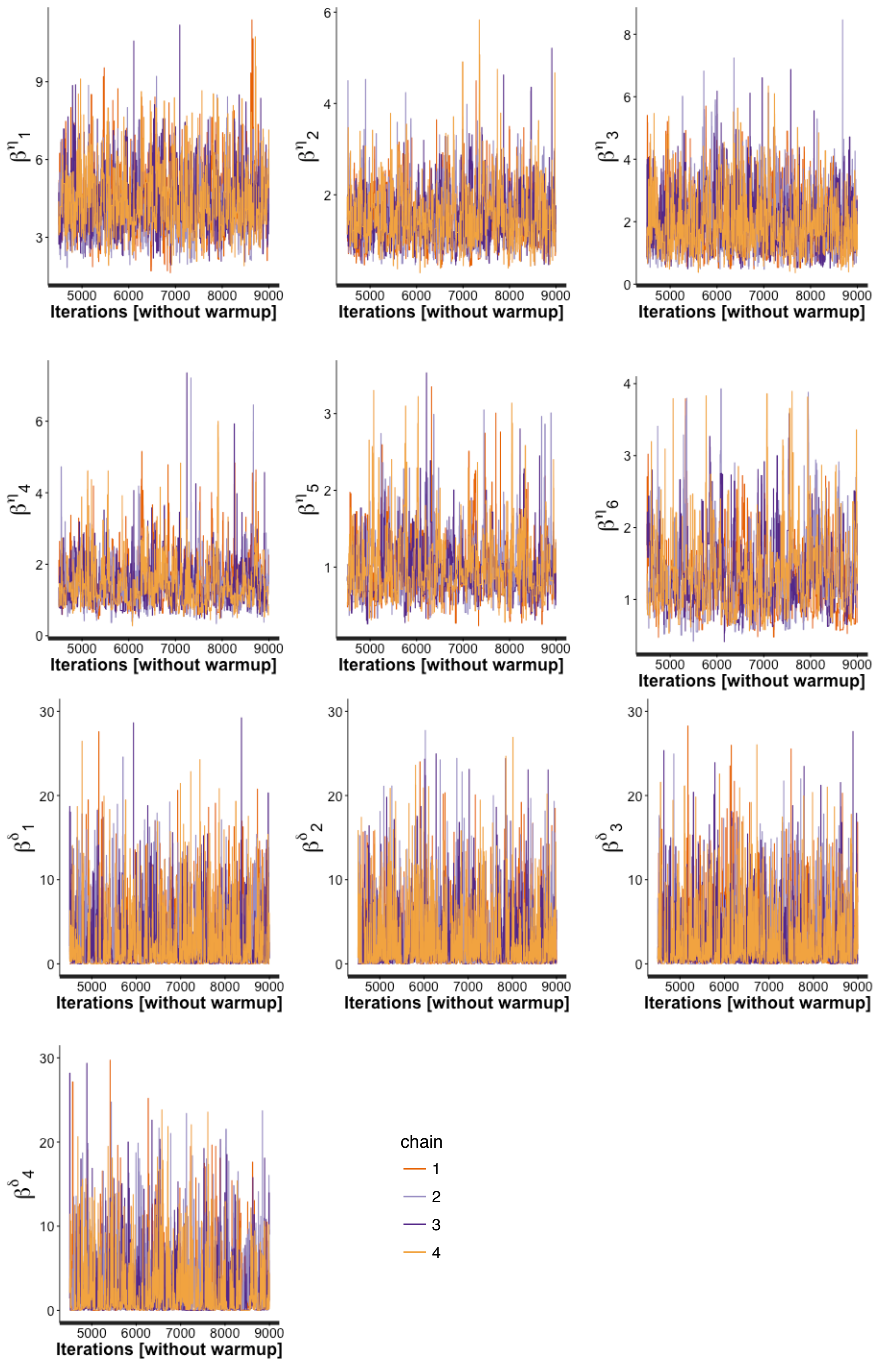

Figure A.14: Case study 1: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using Gibbs sampling algorithm

A.3.2 Case study 2 - office building in Pennsylvania U.S.A

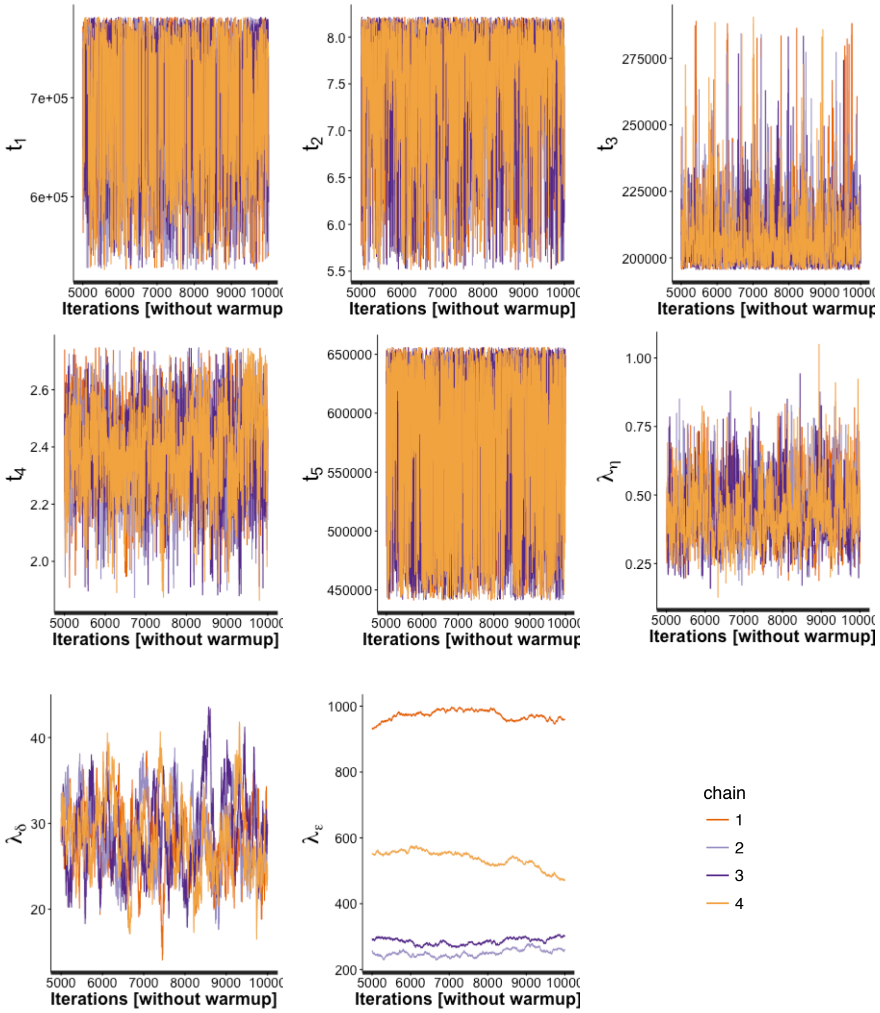

Figure A.15: Case study 2: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using Gibbs sampling algorithm

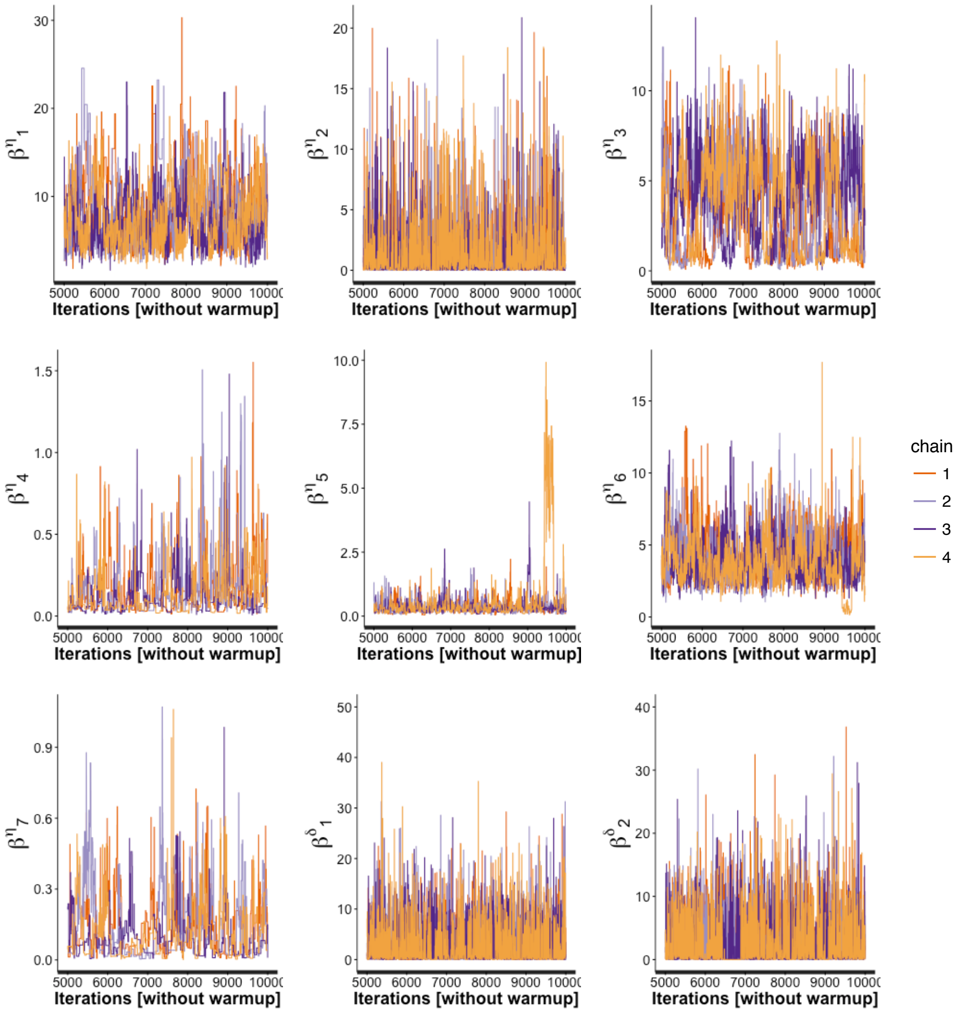

Figure A.16: Case study 2: Trace plots of calibration parameters correlation hyperparameters (β^{η} and β^{δ}) using Gibbs sampling algorithm

A.3.3 Case study 3 - mixed-use building on a university campus in Singapore

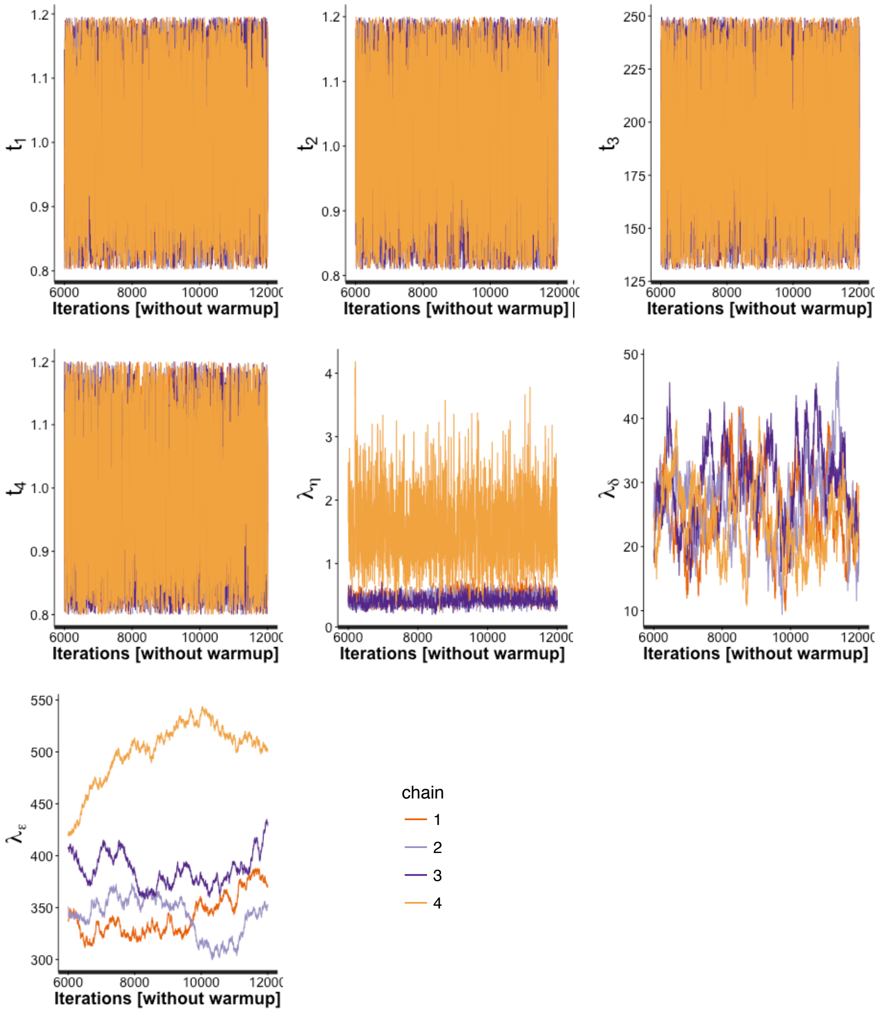

Figure A.17: Case study 3: Trace plots of calibration parameters (t) and variance hyperparameters (λ_η, λ_δ, and λ_ϵ) using Gibbs sampling algorithm

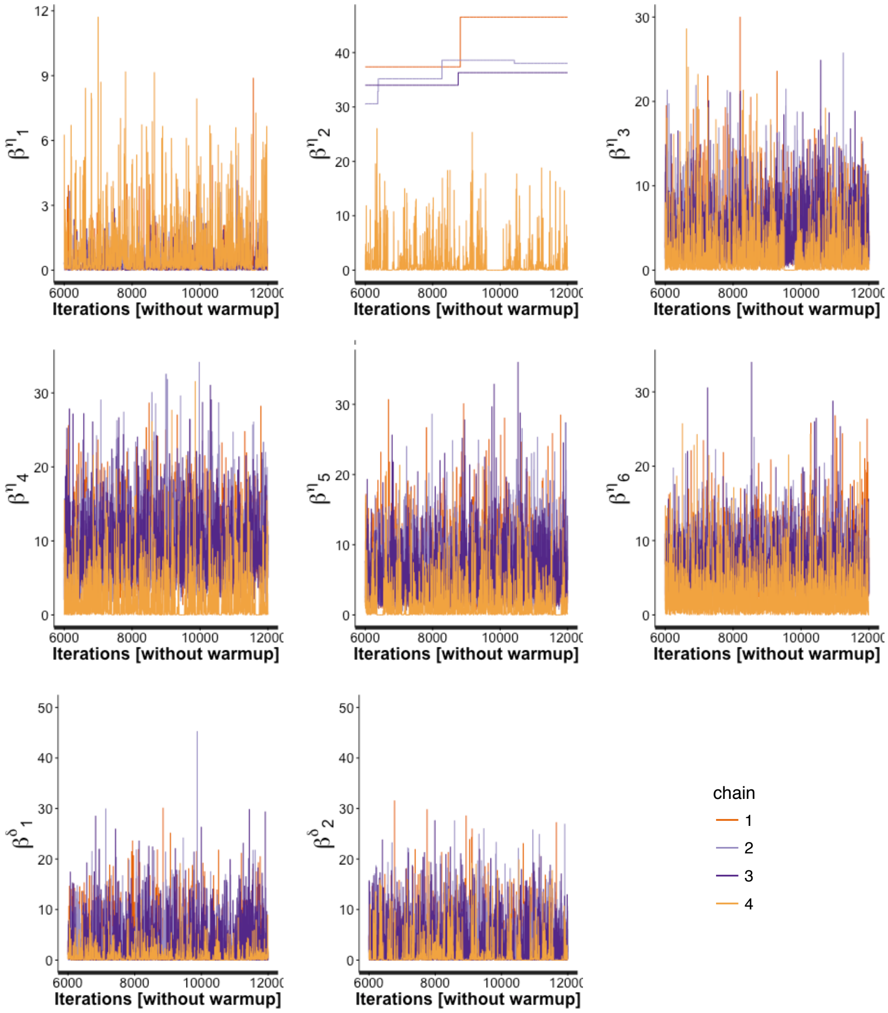

Figure A.18: Case study 3: Trace plots of calibration parameters correlation hyperparameters (β^η and β^δ) using Gibbs sampling algorithm

Appendix B

Code for Bayesian calibration using NUTS

B.1 Setting up the data

Following Chapter 2.1, suppose we have a field dataset D^F (Table B.1) and a computer simulation dataset D^S (Table B.2) stored in a data frame in R. As shown in Table B.1, D^F consists of a series of n observed outputs $y_1, y_2, ..., y_n \in \mathbb{R}$ paired with corresponding inputs $x_1, x_2, ..., x_n \in \mathbb{R}^p$. Table B.2 shows that D^F consists of a series of nm computer outputs $\eta_1, \eta_2, ..., \eta_{nm} \in \mathbb{R}$ paired with corresponding inputs $x_1, x_2, ..., x_{nm} \in \mathbb{R}^p$ and calibration parameters $t_1, t_2, ..., t_{nm} \in \mathbb{R}^q$. Note that D^S has nm samples because given n observations and m simulation runs, running each simulation at the same observed inputs $x_1, ..., x_n$ would produce a simulation dataset of size nm.

Table B.1: Field data $D^F \in \mathbb{R}^{n\times(1+p)}$ as they appear in a data frame

y	x1	x2	$\cdots$	xp
y_1	$x1_1$	$x2_1$	$\cdots$	xp_1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x1_n$	$x2_n$	$\cdots$	xp_n

Table B.2: Simulation data $D^S \in \mathbb{R}^{nm \times (1+p+q)}$ as they appear in a data frame

eta	x1	x2	$\cdots$	xp	t1	t2	$\cdots$	tq
eta_1	$x1_1$	$x2_1$	$\cdots$	xp_1	$t1_1$	$t2_1$	$\cdots$	tq_1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
eta_{nm}	$x1_{nm}$	$x2_{nm}$	$\cdots$	xp_{nm}	$t1_{nm}$	$t2_{nm}$	$\cdots$	tq_{nm}

Before fitting a Gaussian Process (GP) model, we first standardize the outputs and inputs. We first extract corresponding parts of D^F and D^S

Listing B.1: R code for setting up the data

```
# get dimension of dataset
# D.FIELD: field dataset (Table B.1)
# D.SIM: computer simulation dataset (Table B.2)
p ← ncol(D.FIELD) - 1 # number of input factors
q ← ncol(D.SIM) - p - 1 # number of calibration parameters
n ← nrow(D.FIELD) # sample size of observed field data
nm ← nrow(D.SIM) # sample size of computer simulation data

# extract data from field dataset and computer simulation dataset
# assuming D.FIELD and D.COMP is in same format as Tables B.1 and B.2
y ← D.FIELD[,1] # observed output
xf ← D.FIELD[,2:(1+p)] # observed input
eta ← D.SIM[,1] # simulation output
xc ← D.SIM[,2:(1+p)] # simulation input
t ← D.SIM[,(2+p):(1+p+q)] # calibration parameters
```

The is followed by standardization of the outputs y and η using the mean and standard deviation of η.

Listing B.2: R code for standardization of y and η

```
eta_mu ← mean(eta, na.rm = TRUE) # mean value
eta_sd ← sd(eta, na.rm = TRUE) # standard deviation
y ← (y - eta_mu) / eta_sd
eta ← (eta - eta_mu) / eta_sd
```

We then normalize the observed inputs `xf`, computer simulation inputs `xc` and calibration parameters `t` by placing them in the range of 0 to 1.

Listing B.3: R code for normalization of x and t

```
# Put design points xf and xc on [0,1]
x ← rbind(xf,xc)
for (i in (1:ncol(x))) {
  x_min ← min(x[,i], na.rm = TRUE)
  x_max ← max(x[,i], na.rm = TRUE)
  xf[,i] ← (xf[,i] - x_min) / (x_max - x_min)
  xc[,i] ← (xc[,i] - x_min) / (x_max - x_min)
}
# Put calibration parameters t on domain [0,1]
for (j in (1:ncol(t))) {
  t_min ← min(t[,j], na.rm = TRUE)
  t_max ← max(t[,j], na.rm = TRUE)
  t[,j] ← (t[,j] - t_min) / (t_max - t_min)
}
```

B.2 Simulating from a Gaussian Process using NUTS

We use Stan (Stan Development Team, 2016) for the MCMC sampling process. Stan can be set to use the No-U-Turn sampler (NUTS) (Hoffman and Gelman, 2014) for the sampling. This is carried out using RStan version 2.12, the R interface to Stan. To fit a Gaussian Process (GP) regression, we combine the observed output $y_1, ..., y_n \in \mathbb{R}$ with the simulation output $\eta_1, ..., \eta_{nm} \in \mathbb{R}$ in a single $N = n + nm$ vector $z = \big[y_1, ..., y_n, \eta_1, ..., \eta_{nm}\big]$ (Higdon et al., 2004).

$$z \sim \mathcal{N}(\mu_z, \Sigma_z) \tag{B.1}$$

where μ_z is a $N = n + nm$ vector and Σ_z is a $N \times N$ covariance matrix. The first step is to define a Gaussian Process (GP) in Stan which is parameterized by a mean function and a covariance function. As mentioned in Chapter 2.2.2, we define the mean function μ_z to always return the zero vector and covariance function Σ_z as follows (Higdon et al., 2004)

$$\Sigma_z = \Sigma_\eta + \begin{bmatrix} \Sigma_\delta + \Sigma_y & 0 \\ 0 & 0 \end{bmatrix} \tag{B.2}$$

$$\Sigma_{\eta,ij} = \frac{1}{\lambda_\eta} exp\bigg\{ - \sum_{k=1}^{p} \beta_k^\eta |x_{ik} - x_{jk}|^2 - \sum_{k'=1}^{q} \beta_{p+k'}^\eta |t_{ik'} - t_{jk'}|^2 \bigg\} \tag{B.3}$$

$$\Sigma_{\delta,ij} = \frac{1}{\lambda_\delta} exp\bigg\{ - \sum_{k=1}^{p} \beta_k^\delta |x_{ik} - x_{jk}|^2 \bigg\} \tag{B.4}$$

$$\Sigma_y = I_n/\lambda_\epsilon \tag{B.5}$$

where $\Sigma_\eta \in \mathbb{R}^{N\times N}$, $\Sigma_\delta \in \mathbb{R}^{n\times n}$ and $\Sigma_y \in \mathbb{R}^{n\times n}$. β^η and β^δ denotes the correlation hyperparameters and λ_η, λ_δ and λ_ϵ denotes the variance hyperparameters that defines the covariance function. Listing B.5 illustrates an implementation of the GP emulator in Stan. The model consists of five blocks, which includes the data block, the transformed data block, the parameters block, the transformed parameters block and the model block. The data block declares the inputs for the model and allows data to be read from a R data structure to Stan. The transformed data block allows constant to be defined and transformation of data from the data block. Here we define the output vector z and the mean function μ_z to return a vector of zeros.

Next, we define the parameters in the parameters and transformed parameters block. The parameters for Bayesian calibration include the calibration parameters $t_1, ..., t_q$, the correlation hyperparameters of the GP model $\beta_1^\eta, ..., \beta_{p+q}^\eta$ and $\beta_1^\delta, ..., \beta_p^\delta$, and the variance hyperparameters of the GP model λ_η, λ_δ and λ_ϵ. To model the correlation hyperparameters $\beta_1^\eta, ..., \beta_{p+q}^\eta$ and $\beta_1^\delta, ..., \beta_p^\delta$, we reparameterize them in the transformed parameters block with $\beta^\eta = -4.0 \log(\rho^\eta)$ and $\beta^\delta = -4.0 \log(\rho^\delta)$ respectively (Guillas et al., 2009). Since $\beta^\eta > 0$ and $\beta^\delta > 0$, $0 < \rho^\eta < 1$ and $0 < \rho^\delta < 1$.

Finally, the model block specifies the computation of the covariance function, the prior distribution and the likelihood function. To understand the Stan code in the model block, compare the Stan code in the model block with Equations B.3, B.4 and B.5. The model block begins with setting the values of the variable `inputs`, which would then be used for the computation of Σ_η (Listing B.5, lines 51-61) according to Equation B.3. This is followed by the computation of Σ_δ (Listing B.5, lines 62-72) according to Equation B.4 and Σ_y (Listing B.5, lines 73-74) according to Equation B.5. The covariance matrix Σ_z is then computed according to Equation B.2 (Listing B.5, lines 75-77). The rest of the model consists of the priors for the calibration parameters and hyperparameters defining the GP model (Listing B.5, lines 78-90) and multivariate normal likelihood (Listing B.5, lines 81-93). We use Cholesky decomposition for a more efficient implementation of the simulation model.

To fit the Stan model, we save the model (listing B.5) in a text file with a `.stan` extension. We save the Stan model in "`bc.stan`", and fit the model in R by calling the function `stan` from the `rstan` package. This is shown in Listing B.4. Data required as inputs to the Stan model is first saved as a `list` in R (Listing B.4, lines 2-3) and passed to the data block in the Stan model (Listing B.5, lines 1- 12) through the `stan` function (Listing B.4, lines 4-5). Arguments to the `stan` function indicates that we are running 4 independent chains (`chains=4`) of 500 iterations (`iter = 500`) each with the first 250 iterations discarded as warmup/burnin (`warmup=250`). The argument `cores = getOption("mc.cores", 4)` indicates that we are using 4 processors, one for each chain to run the simulations. It is recommended to use as many processors as the hardware and RAM allows (up to the number of chains).

Listing B.4: R code to run Stan model saved in “`bc.stan`”

```
library(rstan)
stanDat ← list(n=n,nm=nm,N=n+nm,p=p,q=q,
                    xf=xf,xc=xc,t=t,y=y,eta=eta)
fit ← stan(file = 'bc.stan', data = stanDat, chains = 4, iter = 500,
    warmup=250,
        cores = getOption("mc.cores", 4))
```

Listing B.5: Stan model for Bayesian calibration with Gaussian Process emulator

```
data {
  int<lower=0> n; // number of observations
  int<lower=0> nm;  // number of simulations
  int<lower=0> N; // N=n+nm
  int<lower=0> p; // number of input factors
  int<lower=0> q; // number of calibration parameters
  matrix[n,p] xf; //
  matrix[nm,p] xc; //
  matrix[nm,q] t; // calibration parameters
  vector[n] y;
  vector[nm] eta;
}
transformed data {
  vector[N] z;
  vector[N] mu_z;
  for (i in 1:N) {
    mu_z[i] = 0;
  }
  z = append_row(y,eta);
}
parameters {
  row_vector<lower=0,upper=1>[q] theta; //calibration parameters
  row_vector<lower=0,upper=1>[p+q] rho_eta;
  row_vector<lower=0,upper=1>[p] rho_delta;
  real<lower=0> lambda_eta; // precision parameter for eta
  real<lower=0> lambda_delta; // precision parameter for bias
  real<lower=0> lambda_e; // precision parameter for observation error
}
transformed parameters {
  // declare variables
  row_vector[p+q] beta_eta;
  row_vector[p] beta_delta;
  beta_eta = -4.0*(log(rho_eta));
  beta_delta = -4.0*(log(rho_delta));
}
model {
  // declare variables
  matrix[N,(p+q)] inputs;
  matrix[N,N] sigma_eta;
  matrix[n,n] sigma_delta;
  matrix[N,N] sigma_z;
  matrix[n,n] sigma_y;
  matrix[N,N] L; // cholesky decomposition of covariance matrix
  row_vector[p] temp_delta;
  row_vector[p+q] temp_eta;
  // set values of inputs which would be used to compute sigma_eta
  inputs[1:n,1:p] = xf;
  inputs[(n+1):N,1:p] = xc;
  inputs[1:n,(p+1):(p+q)] = rep_matrix(theta,n);
  inputs[(n+1):N,(p+1):(p+q)] = t;
  // diagonal elements of sigma_eta
  sigma_eta = diag_matrix(rep_vector((1/lambda_eta),N));
  // off-diagonal elements of sigma_eta
  for (i in 1:(N-1)) {
    for (j in (i+1):N) {
      temp_eta = inputs[i,1:(p+q)] - inputs[j,1:(p+q)];
      sigma_eta[i,j] = beta_eta .* temp_eta * temp_eta';
      sigma_eta[i,j] = exp(-sigma_eta[i,j]) / lambda_eta;
      sigma_eta[j,i] = sigma_eta[i,j];
    }
  }
  // diagonal elements of sigma_delta
  sigma_delta = diag_matrix(rep_vector((1/lambda_delta),n));
  // off-diagonal elements of sigma_delta
  for (i in 1:(n-1)) {
    for (j in (i+1):n) {
      temp_delta = xf[i,1:p] - xf[j,1:p];
      sigma_delta[i,j] = beta_delta .* temp_delta * temp_delta';
      sigma_delta[i,j] = exp(-sigma_delta[i,j]) / lambda_delta;
      sigma_delta[j,i] = sigma_delta[i,j];
    }
  }
  // diagonal elements of sigma_e with off-diagonal elements left as 0
  sigma_y = diag_matrix(rep_vector((1.0/lambda_e),n));
  // computation of covariance matrix sigma_z
  sigma_z = sigma_eta;
  sigma_z[1:n,1:n] = sigma_eta[1:n,1:n] + sigma_delta + sigma_y;
  // Priors
  for (i in 1:(p+q)){
  rho_eta[i] ~ beta(1,0.5);
  }
  for (j in 1:p){
  rho_delta[j] ~ beta(1,0.4);
  }
  for (k in 1:q){
  theta[k] ~ normal(0.5,0.15);
  }
  lambda_eta ~ gamma(10, 10); // gamma (shape, rate)
  lambda_delta ~ gamma(10, 0.3); // gamma (shape, rate)
  lambda_e ~ gamma(10, 0.03); // gamma (shape, rate)
  // cholesky decomposition of covariance matrix
 L = cholesky_decompose(sigma_z);
  z ~ multi_normal_cholesky(mu_z,L);
}
```

B.3 Assessing convergence

We recommend here two practical convergence diagnostics for the assessing convergence in the generated samples. One is to look at the trace plots of multiple chains. After running the `stan` function, trace plots of each component of the posterior can be extracted in R as shown in Listing B.6.

The second convergence diagnostics that we use to assess convergence is the Gelman-Rubin statistics $\hat{R}$. For adequate convergence, $\hat{R}$ should be approximately 1 ± 0.1. This information can be easily obtained with the following line in R with `print(fit)`. `print(fit)` also provides the quantiles of the posterior distribution for each parameter.

Listing B.6: R code for setting up the data

```
# trace plots for calibration parameters
stan_trace(fit, pars = c(’theta’), inc_warmup = FALSE)
# trace plots for correlation hyperparameters beta_eta
stan_trace(fit, pars = c(’beta_eta’), inc_warmup = FALSE)
# trace plots for correlation hyperparameters beta_delta
stan_trace(fit, pars = c(’beta_delta’), inc_warmup = FALSE)
# trace plots for variance hyperparameters
stan_trace(fit, pars = c(’lambda_eta’, ’lambda_delta’, ’lambda_e’),
    inc_warmup = FALSE)
```

Bibliography

ASHRAE (2002). Guideline 14-2002, measurement of energy and demand savings. *American Society of Heating, Ventilating, and Air Conditioning Engineers, Atlanta, Georgia*. (document), 1.1, 1.4.4, 1.1, 1.5, 2.4.4

Augenbroe, G. (2002). Trends in building simulation. *Building and Environment*, 37(8):891–902. 1.1

Beven, K. (2006). A manifesto for the equifinality thesis. *Journal of hydrology*, 320(1):18–36. 1.1

Campolongo, F., Cariboni, J., and Saltelli, A. (2007). An effective screening design for sensitivity analysis of large models. *Environmental modelling & software*, 22(10):1509–1518. 2.2.1, 2.2.1

Carroll, W. and Hitchcock, R. (1993). Tuning simulated building descriptions to match actual utility data: methods and implementation. *ASHRAE Transactions-American Society of Heating Refrigerating Airconditioning Engin*, 99(2):928–934. 1.4.2

Chong, A. and Lam, K. P. (2015). Uncertainty analysis and parameter estimation of hvac systems in building energy models. In *Proceedings of the 14th IBPSA Building Simulation Conference*. 1.1, 1.4.3

Chong, A. and Lam, K. P. (2017). A comparison of mcmc algorithms for the bayesian calibration of building energy models. In *Proceedings of the 15th IBPSA Building Simulation Conference*. 4.1, 5.1

Clarke, J., Strachan, P., and Pernot, C. (1993). An approach to the calibration of building energy simulation models. *TRANSACTIONS-AMERICAN SOCIETY OF HEATING REFRIGERATING AND AIR CONDITIONING ENGINEERS*, 99:917–917. 1.4.1

Clarke, J. A. (2001). *Energy simulation in building design*. Routledge. (document), 1.1

Coakley, D., Raftery, P., and Keane, M. (2014). A review of methods to match building energy simulation models to measured data. *Renewable and Sustainable Energy Reviews*, 37:123–141. 1.4, 1.4.1, 1.4.4, 1.4.4

Crawley, D. B., Hand, J. W., Kummert, M., and Griffith, B. T. (2008). Contrasting the capabilities of building energy performance simulation programs. *Building and environment*, 43(4):661–673. 1.2

Crawley, D. B., Lawrie, L. K., Winkelmann, F. C., Buhl, W. F., Huang, Y. J., Pedersen, C. O., Strand, R. K., Liesen, R. J., Fisher, D. E., Witte, M. J., et al. (2001). Energyplus: creating a new-generation building energy simulation program. *Energy and buildings*, 33(4):319–331. 1.2

De Wit, S. and Augenbroe, G. (2002). Analysis of uncertainty in building design evaluations and its implications. *Energy and Buildings*, 34(9):951–958. 1.3, 2.2.1

DOE, U. (2008). M&v guidelines: Measurement and verification for federal energy project. *Version*, 3:4–22. 1.1

Dong, B., Cao, C., and Lee, S. E. (2005). Applying support vector machines to predict building energy consumption in tropical region. *Energy and Buildings*, 37(5):545–553. 1.4.2

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid monte carlo. *Physics letters B*, 195(2):216–222. 2.4.3

Eisenhower, B., O'Neill, Z., Fonoberov, V. A., and Mezić, I. (2012a). Uncertainty and sensitivity decomposition of building energy models. *Journal of Building Performance Simulation*, 5(3):171–184. 1.3

Eisenhower, B., O'Neill, Z., Narayanan, S., Fonoberov, V. A., and Mezić, I. (2012b). A methodology for meta-model based optimization in building energy models. *Energy and Buildings*, 47:292–301. 1.4.2

ESRU (1974). Esp-r. `http://www.esru.strath.ac.uk/Programs/ESP-r.htm`. 1.2

EVO (2012). International performance measurement and verification protocol: Concepts and options for determining energy and water savings volume 1. *Efficiency Valuation Organization*. (document), 1.1, 1.4.4, 1.1

EVO (2014). Statistics and uncertainty for ipmvp. *Efficiency Valuation Organization*. 1.3

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2014). *Bayesian data analysis*, volume 2. Taylor & Francis. (document), 1.1, 1.5, 2.3, 2.3, 2, 2.4.3, 2.4.4, 2.4.4, 4.1

Gelman, A., Roberts, G. O., Gilks, W. R., et al. (1996). Efficient metropolis jumping rules. *Bayesian statistics*, 5(599-608):42. 4.1

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, 6:721–741. 2.2.3, 2.2.3, 2.4.3

Gu, B., Liu, B., Hu, F., and Liu, H. (2001). Efficiently determining the starting sample size for progressive sampling. In *European Conference on Machine Learning*, pages 192–202. Springer. 2.4.2, 2.4.2

Guillas, S., Rougier, J., Maute, A., Richmond, A., and Linkletter, C. (2009). Bayesian calibration of the thermosphere-ionosphere electrodynamics general circulation model (tie-gcm). *Geoscientific Model Development*, 2(2):137. 3, B.2

Hausser, J. and Strimmer, K. (2009). Entropy inference and the james-stein estimator, with application to nonlinear gene association networks. *Journal of Machine Learning Research*, 10(Jul):1469–1484. (document), 2.4.2, 2.1

Heo, Y., Augenbroe, G., Graziano, D., Muehleisen, R. T., and Guzowski, L. (2015a). Scalable methodology for large scale building energy improvement: Relevance of calibration in model-based retrofit analysis. *Building and Environment*, 87:342–350. 1.1, 1.4.3

Heo, Y., Choudhary, R., and Augenbroe, G. (2012). Calibration of building energy models for retrofit analysis under uncertainty. *Energy and Buildings*, 47:550–560. 1.1, 1.4.2, 1.4.3, 1.4.3, 2.1, 2.2.1

Heo, Y., Graziano, D. J., Guzowski, L., and Muehleisen, R. T. (2015b). Evaluation of calibration efficacy under different levels of uncertainty. *Journal of Building Performance Simulation*, 8(3):135–144. 1.1

Higdon, D., Kennedy, M., Cavendish, J. C., Cafeo, J. A., and Ryne, R. D. (2004). Combining field data and computer simulations for calibration and prediction. *SIAM Journal on Scientific Computing*, 26(2):448–466. 1.1, 5, 2.2.2, 2.2.2, 2.2.2, 3.1.2, 3.2.3, 3.3.3, B.2, B.2

Higdon, D., Nakhleh, C., Gattiker, J., and Williams, B. (2008). A bayesian calibration approach to the thermal problem. *Computer Methods in Applied Mechanics and Engineering*, 197(29):2431–2441. 3

Hittle, D. (1979). Building loads analysis and system thermodynamics (blast) users manual. *US Army Construction Engineering Research Laboratory (USA-CERL), Champaign, IL*. 1.2

Hoffman, M. D. and Gelman, A. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623. (document), 6, 2, 2.4.3, 2.1, 2.4.3, 4.1, 4.1, 5.1, B.2

Hydeman, M. and Gillespie Jr, K. L. (2002). Tools and techniques to calibrate electric chiller component models/discussion. *ASHRAE Transactions*, 108:733. 3.2.1

Ian Shapiro, P. (2009). Energy audits. *ASHRAE Journal*, 15:18–27. 1.4.1

Integrated Environment Solutions (2015). The ies virtual environment user guide. 1.2

Jeffreys, H. (1946). An invariant form for the prior probability in estimation problems. In *Proceedings of the Royal Society of London a: mathematical, physical and engineering sciences*, volume 186, pages 453–461. The Royal Society. 2.1

Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464. 1.1, 1.4.3, 1.4.3, 1.4.3, 2.1

Kern, J. C. (2000). *Bayesian process-convolution approaches to specifying spatial dependence structure*. PhD thesis, Duke University. 3

Klein, S., Beckman, W., Mitchell, J., Duffie, J., Duffie, N., and Freeman, T. (2012). Trnsys 17: A transient system simulation program. madison, usa: Solar energy laboratory, university of wisconsin. 1.2

Kristensen, M. H. and Petersen, S. (2016). Choosing the appropriate sensitivity analysis method for building energy model-based investigations. *Energy and Buildings*, 130:166–176. 2.2.1

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86. 1.5, 2.4.2, 2.4.2, 5.1

LBNL (2011). Genopt generic optimization program: User manual version 3.1.0. *US Department of Energy*. 1.4.2

LBNL (2016a). Energyplus engineering reference: the reference to energyplus calculations. *US Department of Energy*. 3.2.1, 3.2.1

LBNL (2016b). Energyplus getting started. *US Department of Energy*. 1.2

LBNL (2016c). Energyplus input output reference: the encyclopedic reference to energyplus input and output. *US Department of Energy*. 3.2.1, 3.2.1, 3.2.1

Li, Q., Augenbroe, G., and Brown, J. (2016). Assessment of linear emulators in lightweight bayesian calibration of dynamic building energy models for parameter estimation and performance prediction. *Energy and Buildings*, 124:194–202. 1.1, 1.4.2, 1.4.3

Liu, G. and Liu, M. (2011). A rapid calibration procedure and case study for simplified simulation models of commonly used hvac systems. *Building and Environment*, 46(2):409–420. 1.4.1

Macdonald, I. and Strachan, P. (2001). Practical application of uncertainty analysis. *Energy and Buildings*, 33(3):219–227. 1.3

Macdonald, I. A. (2002). *Quantifying the effects of uncertainty in building simulation*. University of Strathclyde. 3.3.2

Manfren, M., Aste, N., and Moshksar, R. (2013). Calibration and uncertainty analysis for computer models–a meta-model based approach for integrated building energy simulation. *Applied energy*, 103:627–641. 1.4.2, 1.4.3

Manke, J., Hittle, D., and Hancock, C. (1996). Calibrating building energy analysis models using short term test data. In *Proceedings of the 1996 ASME International Solar Engineering Conference*, pages 369–378. 1.4.1

Menberg, K., Heo, Y., and Choudhary, R. (2016). Sensitivity analysis methods for building energy models: Comparing computational costs and extractable information. *Energy and Buildings*, 133:433–445. 2.2.1

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092. 2.2.3, 2.4.3

Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174. 2.2.1, 2.2.1

Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods. 2.4.3

Neal, R. M. (2011). Mcmc using hamiltonian dynamics. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X.-L., editors, *Handbook of markov chain monte carlo*, volume 2, chapter 5, pages 113–162. Chapman and Hall/CRC. 2.4.3

Neto, A. H. and Fiorelli, F. A. S. (2008). Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption. *Energy and buildings*, 40(12):2169–2176. 1.4.2

Pedrini, A., Westphal, F. S., and Lamberts, R. (2002). A methodology for building energy modelling and calibration in warm climates. *Building and Environment*, 37(8):903–912. 1.4.1

Perks, W. (1947). Some observations on inverse probability including a new indifference rule. *Journal of Institute of Actuaries*, 73:285–334. 2.1

Provost, F., Jensen, D., and Oates, T. (1999). Efficient progressive sampling. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 23–32. ACM. 2.4.2

Pujol, G., Iooss, B., with contributions from Khalid Boumhaout, A. J., Veiga, S. D., Fruth, J., Gilquin, L., Guillaume, J., Le Gratiet, L., Lemaitre, P., Ramos, B., Touati, T., and Weber, F. (2016). *sensitivity: Global Sensitivity Analysis of Model Outputs*. R package version 1.12.2. 2.2.1, 3.2.2, 3.3.2

R Core Team (2015). R: A language and environment for statistical computing, version 3.2.3. `https://www.R-project.org/`. 2.4.3

Raftery, P., Keane, M., and Costa, A. (2011). Calibrating whole building energy models: Detailed case study using hourly measured data. *Energy and Buildings*, 43(12):3666–3679. 1.4.1, 1.4.4

Reddy, T. A. (2006). Literature review on calibration of building energy simulation programs: Uses, problems, procedures, uncertainty, and tools. *ASHRAE transactions*, 112(1). 1.1, 1.4.1, 1.4.2, 1.4.4

Riddle, M. and Muehleisen, R. T. (2014). A guide to bayesian calibration of building energy models. In *ASHRAE/IBPSA-USA Building Simulation Conference*. 1.1

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008). *Global sensitivity analysis: the primer*. John Wiley & Sons. (document), 2.2.1, 2.2, 2.2.1

Soebarto, V. I. (1997). Calibration of hourly energy simulations using hourly monitored data and monthly utility records for two case study buildings. In *Proceedings of the 4th IBPSA Building Simulation Conference*, pages 411–419. 1.4.1

Sokol, J., Davila, C. C., and Reinhart, C. F. (2017). Validation of a bayesian-based method for defining residential archetypes in urban building energy models. *Energy and Buildings*, 134:11–24. 1.4.3

Stan Development Team (2016). Rstan: the r interface to stan, version 2.9.0. `http://mc-stan.org`. 2.4.3, B.2

Stein, M. (1987). Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151. 2.2.2, 3.1.2, 3.2.3, 3.3.3

Subbarao, K. (1988). Pstar–primary and secondary terms–analysis and renormalization: a unified approach to building and energy simulations and short-term testing–a summary. september 1988. Technical report, SERI/TR-254-3347. Colorado, USA: Solar Energy Research Institute. 1.4.1

Sun, Y., Heo, Y., Tan, M., Xie, H., Jeff Wu, C., and Augenbroe, G. (2014). Uncertainty quantification of microclimate variables in building energy models. *Journal of Building Performance Simulation*, 7(1):17–32. 1.3

Taheri, M., Tahmasebi, F., and Mahdavi, A. (2012). A case study of optimization-aided thermal building performance simulation calibration. *Proceedings of the 13th International IBPSA Conference*, pages 603–607. 1.4.2

Thornton, J., Bradley, D., McDowell, T., Blair, N., Duffy, M., LaHam, N., and Naik, A. (2014). Tesslibs 17: Hvac library mathematical reference. 3.1.1

Tian, W. (2013). A review of sensitivity analysis methods in building energy analysis. *Renewable and Sustainable Energy Reviews*, 20:411–419. 2.2.1

Trimble Buildings (2017). Sefaira. 1.2

Trybula, S. (1958). Some problems of simultaneous minimax estimation. *The Annals of Mathematical Statistics*, pages 245–253. 2.1

Winkelmann, F., Birdsall, B., Buhl, W., Ellington, K., Erdem, A., Hirsch, J., and Gates, S. (1993). Doe-2 supplement: version 2.1 e. Technical report, Lawrence Berkeley Lab., CA (United States); Hirsch (James J.) and Associates, Camarillo, CA (United States). 1.2

Yoon, J., Lee, E.-J., and Claridge, D. (2003). Calibration procedure for energy performance simulation of a commercial building. *Journal of solar energy engineering*, 125(3):251–257. 1.4.1