Validation of a Computer Code for the Energy Consumption of a Building, with Application to Optimal Electric Bill Pricing

Merlin Keller, Guillaume Damblin, Alberto Pasanisi, Mathieu Schumann, Pierre Barbillon, Fabrizio Ruggeri

Article

Validation of a Computer Code for the Energy Consumption of a Building, with Application to Optimal Electric Bill Pricing

Merlin Keller [1], **Guillaume Damblin** [2], **Alberto Pasanisi** [3], **Mathieu Schumann** [4], **Pierre Barbillon** [5], **Fabrizio Ruggeri** [6,*] **and Eric Parent** [5]

1   Électricité de France, 78401 Chatou, France
2   CEA, Université Paris-Saclay, 91191 Gif-sur-Yvette, France
3   Edison, 20121 Milano, Italy
4   Électricité de France, 91120 Palaiseau, France
5   Université Paris-Saclay, AgroParisTech, INRAE, UMR MIA Paris-Saclay, 91120 Palaiseau, France
6   CNR IMATI, Via Alfonso Corti 12, 20133 Milano, Italy
*   Correspondence: fabrizio@mi.imati.cnr.it

Abstract: In this paper, we present a case study aimed at determining a billing plan that ensures customer loyalty and provides a profit for the energy company, whose point of view is taken in the paper. The energy provider promotes new contracts for residential buildings, in which customers pay a fixed rate chosen in advance, based on an overall energy consumption forecast. For such a purpose, we consider a practical Bayesian framework for the calibration and validation of a computer code used to forecast the energy consumption of a building. On the basis of power field measurements, collected from an experimental building cell in a given period of time, the code is calibrated, effectively reducing the epistemic uncertainty affecting the most relevant parameters of the code (albedo, thermal bridge factor, and convective coefficient). The validation is carried out by testing the goodness of fit of the code with respect to the field measurements, and then propagating the posterior parametric uncertainty through the code, obtaining probabilistic forecasts of the average electrical power delivered inside the cell in a given period of time. Finally, Bayesian decision-making methods are used to choose the optimal fixed rate (for the energy provider) of the contract, in order to balance short-term benefits with customer retention. We identify three significant contributions of the paper. First of all, the case study data were never analyzed from a Bayesian viewpoint, which is relevant here not only for estimating the parameters but also for properly assessing the uncertainty about the forecasts. Furthermore, the study of optimal policies for energy providers in this framework is new, to the best of our knowledge. Finally, we propose Bayesian posterior predictive *p*-value for validation.

Keywords: uncertainty quantification; Bayesian analysis; energy contracts

1. Introduction

Today, more and more green buildings are designed to keep energy consumption low with the aim of dividing CO_2 emissions by four by 2050. This challenge implies, among other things, to be able to predict the energy performance of existing or future buildings, a task which can be tackled using thermal building models, implemented in numerical simulation computer codes, which can be thought of as deterministic functions with an input, some fixed parameters that can be uncertain, and an output.

An important example is *retrofit analysis*, in which stakeholders are tasked with investigating and recommending specific restructuring measures from a wide range of possible options, as described in Pasanisi and Ojalvo (2008). The retrofit analysis is based on all available information concerning the building, contained for example in the building information model (BIM) (Eastman et al. 2011), as well as on energy consumption models implemented into computer codes.

For instance, building energy models specifically designed for the retrofit analysis of office and school buildings are described in Refs. Rysanek and Choudhary (2012) and Tian and Choudhary (2011). In both works, sensitivity analyses are provided in the form of standard regression coefficient (SRC) indices (Saltelli et al. 2000), which help detect the inputs that have the greatest impact on predicted energy consumption, and correction factors are applied to make the model predictions more similar to historical consumption data. A more principled use of historical data involves a *calibration* step, in which uncertain model parameters are tuned so that model predictions fit the historical data as well as possible. In Heo et al. (2012), such a calibration is performed for normative building energy models.

Design (or *conception*) is another important application of building energy modeling. Contrary to retrofit analysis, no historical data are available at the conception stage, and predictions must be made for a whole range of possible designs in order to choose the optimal one. Furthermore, special care should be paid to assessing the level of confidence one can have in the predictions of such models, necessarily built in the absence of historical data. In Rivalin (2016), a complete uncertainty treatment framework for thermal building modeling is presented, following the popular 'non-intrusive approach' commonly used in the industrial practice; see Pasanisi and Dutfoy (2012) and Baudin et al. (2017). This includes: probabilistic modeling of different sources of uncertainty using expert opinion, sensitivity analysis, meta-modeling to accelerate calculations (Sacks et al. 1989), and uncertainty propagation using either Monte-Carlo (or quasi Monte-Carlo) methods, or quadratic summation via Taylor's approximation. Other aspects about energy in buildings have been analyzed in, e.g., Wate et al. (2020), who considered the effect of insulation thickness and window transmittance on the annual heating and cooling demand per unit floor area.

Recently, the concerns about design and maintenance of buildings and energy systems to be as sustainable and green as possible, following a computer simulation approach, have been extended to the neighborhood and the city level. This requires a systemic methodology, which raises challenging problems for modelers and statisticians (Mirakyan et al. 2015), since it has to take into account, in particular, the interactions between different uses of energy, as well as transportation, water, and waste management. As an application example, we point out the simulation platform developed for the city of Singapore (Blin et al. 2015).

However, all the above approaches rely on the capacity of the chosen physical model, and the ensuing computer code, to mimic 'well-enough' in some sense the actual behavior of a real building, even if most models are based on simplified equations, which only partially reflect the building geometry and its physical properties. This is a common question in science and engineering, when complex physical systems are studied by means of computer codes, or because physical experiments are unfeasible or economically too expensive. Indeed, large differences are regularly observed between physical measurements and code predictions, raising the question of how well the code is capable of reproducing the physical system. See Kennedy and O'Hagan (2001) for a detailed discussion, and Damblin et al. (2018) for a statistical overlook on this topic. Therefore, it is necessary to verify the agreement between these two sets of data to conclude whether or not the code can be used as a surrogate for reality.

This task, called *validation* by Bayarri et al. (2007), has already been tackled in the field of building energy simulation in Bontemps (2015), through graphical comparisons between measurements and corresponding code predictions for quantities of interest, such as temperature or power consumption. While easy to use, these methods remain qualitative and do not provide the causes of a poor fit. Many works dedicated to validation therefore propose to assess the accuracy of the code predictions in a quantitative way, taking into account all sources of uncertainty, both *aleatory* and *epistemic*, as advocated in Roy and Oberkampf (2011), and detailed below. These two sources of uncertainty have been considered by other authors in the context of energy consumption in buildings; for example, Shamsi et al. (2020) uses copulas and a nested Fuzzy Monte Carlo approach to deal with correlations between different types of uncertainties.

When modeling the thermal behavior of a building, aleatory uncertainty is essentially due to both weather conditions and the profile of the inhabitants, which are naturally variable. For instance, Spitz (2012) treats this source of uncertainty in the conception stage by conducting a local sensitivity analysis, then a global sensitivity analysis, in order to identify the factors that have the greatest impact on the code predictions. There is also an increasing interest in the effects of climate change on the energy consumption of buildings; see Fonseca et al. (2020), who dealt with uncertainty quantification using Gaussian mixture models and neural networks.

Epistemic uncertainty is strongly different by nature from aleatory uncertainty; it derives from the lack of a precise knowledge of the physical laws governing the system or the process under study. One example is the parametric uncertainty, i.e., the uncertainty affecting some model parameters, seen as physical constants (Pasanisi et al. 2012). In a Bayesian framework, parametric uncertainty can be encoded as a probability distribution which combines a prior belief about the value of the uncertain parameters with the information provided by the data; see Bernardo and Smith (1994). The quantification of this parametric uncertainty is called *calibration*; see Campbell (2006).

Another source of epistemic uncertainty also arises from the lack of knowledge about the degree of adequacy between the code and the physical phenomenon, as discussed above. This can be quantified within the calibration step, together with the parametric uncertainty, following Kennedy and O'Hagan (2001); however, this leads to a rather complicated statistical analysis. In this work, we advocate a sequential approach where, as in Cox et al. (2001), the presence of a discrepancy between available measures and code predictions is tested using the posterior predictive *p*-value, as introduced by Gelman et al. (1996). If such a test is inconclusive, in the sense that no significant discrepancy is detected, then the code is considered as valid, meaning that it can be adequately used to predict the building behavior.

Once the code has been validated, its final usage can be considered. In this study, we address the decision-making problem of an energy supplier offering a new type of contract, in which the customer pays a fixed fee chosen in advance, based on a forecast of the overall energy consumption. According to Bayesian decision theory (Bernardo and Smith (1994); French and Insua (2000); Robert and Casella (2004)), we show how to choose this fee optimally from the supplier's point of view, in order to balance the short-term benefits with customer loyalty.

We identify three significant contributions of the paper. First of all, the Bayesian analysis of the case study is new, although we rely on previous works, such as Bontemps (2015). Those authors considered the sensitivity analysis to reduce the number of quantities (parameters) that greatly affect power consumption, but we take an important step forward, considering a Bayesian approach which allows for a sound estimation of the parameters. As usual, the Bayesian approach allows us not only to estimate the parameters but also to provide a measure of uncertainty around those values, through the posterior distribution. The Bayesian approach allows for a better understanding of the effect of the three relevant parameters on energy consumption. Furthermore, the Bayesian approach allows us to undertake the second (and very relevant) new contribution: the choice of an optimal policy, from the point of view of the supplier, about fixed price contracts, taking care of the supplier profit, based on conflicting aspects such as the selling price and customer retention: when the former is too high, then it is difficult to keep the customer. Finally the other important contribution is more "technical", due to the way we used the Bayesian posterior predictive *p*-value for validation.

In Section 2, we present the case study and introduce some notations. In Section 3, after assuming a statistical model between the code outputs and the power field measurements, the calibration of the thermal code is performed using a Bayesian approach. In Section 4, the validation of the code is conducted by calculating the uncertainty affecting the average electric power delivered inside the cell over the time period. In Section 5, several optimal consumption forecasts are calculated using Bayesian decision theory. We stress that, even

though this study is based on real experiments, the application to optimal pricing must be seen as a fictitious exercise and it is not representative of EDF (Électricité de France) billing policies. The conclusions of the work are given in Section 6.

2. Overview of the Case Study

2.1. Dymola Computer Code

The computer code we use is a dynamic building energy model, which predicts the electric power consumption needed to heat a room to a predetermined temperature. The room is an experimental cell in the BESTLab laboratory, a description of which is given below. The model was built using BuildSysPro, a library of components for modeling buildings and energy systems (Plessis et al. 2014), written in the Modelica language, and used for simulation in the Dymola software. Originally developed in Elmqvist (1978), Modelica is well adapted for modeling large-scale structures, whose behavior is defined through the interaction of its components, each described by a limited set of equations, resulting in an overall system of (usually non-linear) equations, solved by general purpose algorithms. An important aspect of this system-based approach is its modularity, enabling to create quickly and simply large-scale models from a library of elementary building blocks. The final computer code itself is an executable built using the Dymola software.

In terms of notations, let P_t be the actual electric power consumption in the cell at time step t, for $t \in \{1, \ldots, T\}$, and let y_t be the corresponding code prediction. The latter is computed given the power consumption y_{t-1} at the previous time step, and a vector $\mathbf{x}_t \in \mathbb{R}^n$ of forcing variables (physical inputs) at time step t, including cell characteristics, average temperature inside the cell, and weather conditions such as wind speed, outside temperature and solar radiation, among others. In addition, this code depends on a vector of fixed parameters $\boldsymbol{\theta} \in \mathbb{R}^p$. The dimension of $\boldsymbol{\theta}$ is $p = 193$, including both some physical parameters such as albedo, convective factor, and many design variables such as floor surface, window width, and so forth. Whereas the physical inputs change over time, in general independently of the engineers' will, the parameters can be tuned to reduce, e.g., the power consumption. The engineers identified 193 fixed parameters affecting the power consumption and the computer code depends on them. Later on, we will show why we actually considered and estimated only the three most relevant ones.

Consequently, the code can be denoted by the function $g_{\boldsymbol{\theta}}$ such that for $t = 1, \ldots, T$:

$$y_t = g_{\boldsymbol{\theta}}(y_{t-1}, \mathbf{x}_t).$$

In other words, the model-defining equations are solved iteratively at each time step, reflecting how the cell dynamically reacts to variations in its environment.

Equivalently, the resulting dynamic code can be described more simply as a function $G_{\boldsymbol{\theta}}$ depending on model parameters $\boldsymbol{\theta}$, computing a power consumption forecast $\mathbf{Y} = (y_1, \ldots, y_T) \in \mathbb{R}^T$ over the whole time period, given the initial power consumption y_0, the complete matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_T) \in \mathbb{R}^{T \times n}$ of forcing variables, so that:

$$\mathbf{Y}(\boldsymbol{\theta}) = G_{\boldsymbol{\theta}}(y_0, \mathbf{X}), \tag{1}$$

where we insist on the dependence of code predictions $\mathbf{Y}(\boldsymbol{\theta})$ on the parameter $\boldsymbol{\theta}$, rather than on the initial condition y_0 and forcing variables $\mathbf{X}$, which will be kept fixed during our study. It is worth mentioning that $\mathbf{Y} : \boldsymbol{\theta} \mapsto \mathbf{Y}(\boldsymbol{\theta})$ is a non-linear function, and this will affect inferences later on.

However, note that defining $\mathbf{X}$ as input and $\mathbf{Y}$ as output of the physical model is somewhat arbitrary, since the code essentially solves a system of equations for which some variables have been fixed. Hence, it would be equally possible to redefine the code as as a calculation of the temperature as a function of the electric power, as in Bontemps et al. (2013). Our choice will become clearer when describing the particular experimental sequence we have considered here.

2.2. Experimental Data

The experimental cell is 1 of the 12 cells making up the BESTLab laboratory (see Figure 1), in the EDF research and development site of Les Renardières, about 75 km southeast of Paris. BESTLab was built in 2010, in the context of increasing demand for low energy buildings. Its primary purpose is to test innovative building envelope components and integrated solar technologies. See Bontemps et al. (2013) and Bontemps (2015) for a detailed description of the installation.
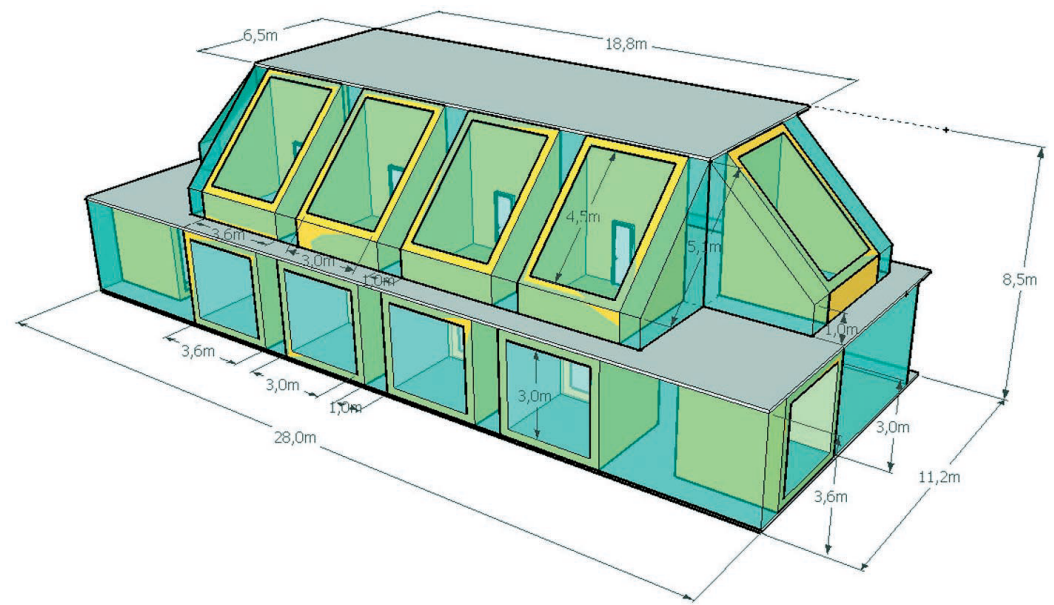

Figure 1. General layout of the EDF BESTLAB experimental platform. We will focus on the thermal behavior of one of the lower level cells.

Our study focuses on one of the ground cells (see Figure 2), comprising a single, windowed, wall in contact with the outside environment. A heating, ventilation, and air conditioning (HVAC) system allows us to heat or cool the cell at will, thus allowing us to test different types of scenarios. Figure 3 shows a typical experimental sequence, wherein the cell is alternately maintained at certain target temperatures, heated with a fixed power supply, or left unconditioned, so that the temperature evolves freely.

In our case, we will focus on the first 7 days period of this sequence, where the cell is maintained at 20 °C. The main reason for doing so is that this experimental condition mostly resembles the usual state of residential housing. Furthermore, the code can more accurately reflect certain experimental conditions then others, so it makes more sense to validate it separately for each experimental condition, before considering global validation.

Moreover, heterogeneity in experimental sequences raises the issue of defining what the inputs and outputs of the computer code are, as discussed in the previous section. Indeed, when maintaining the cell at a fixed temperature, it makes sense to consider the latter as an input, and the power consumption as the output of interest. Conversely, when the cell is heated with a fixed power supply, or when the HVAC system is turned off, it would make more sense, from a physical point of view, to consider the power consumption as an input (which can be zero), and the resulting temperature as an output.

Finally, because our final use of the code is to predict the total power consumption over a given period, the measured and computed powers are averaged, resulting in 30 time steps over 7 days, rather than the 2016 five-minute time steps initially considered. This allows us to considerably reduce the computational burden of the analysis, also reducing the variance, both in the experimental and the computer code outputs. Figure 4 shows the resulting smoother data we use for code validation.
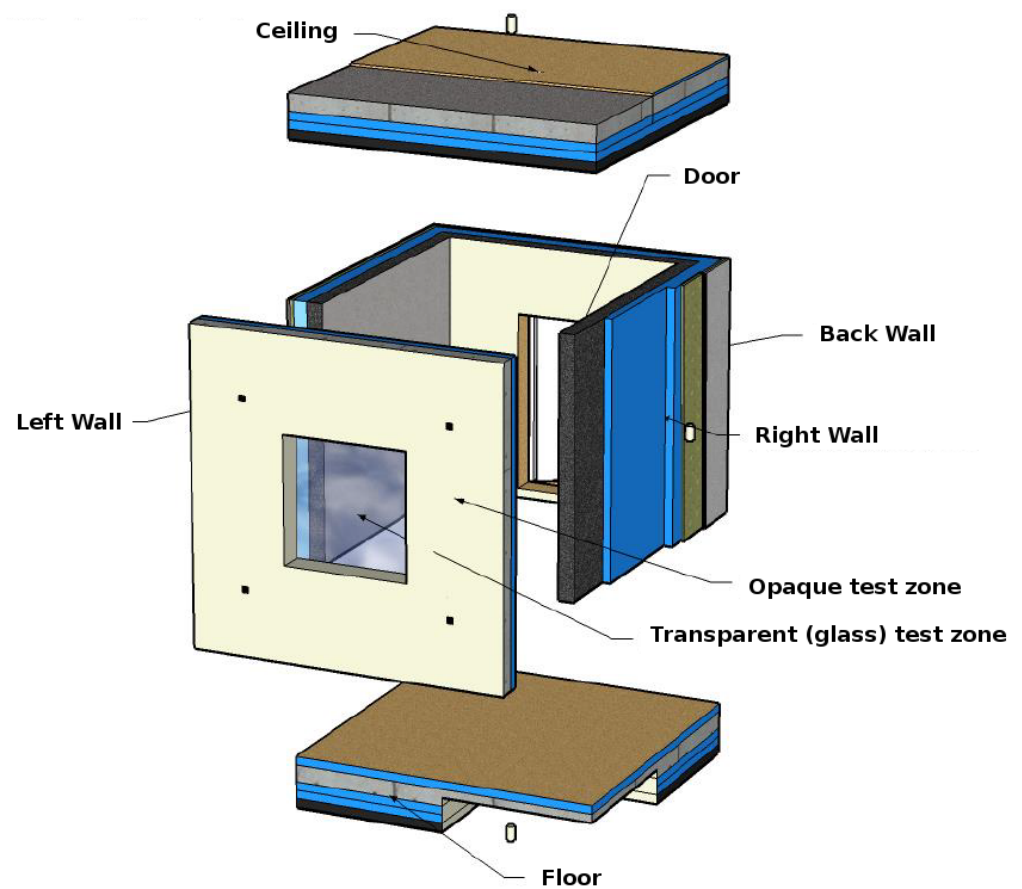
Figure 2. Experimental cell under study, equipped with captors monitoring temperature, power consumption, and other variables of interest, and whose thermal behavior is modeled by the DYMOLA code.
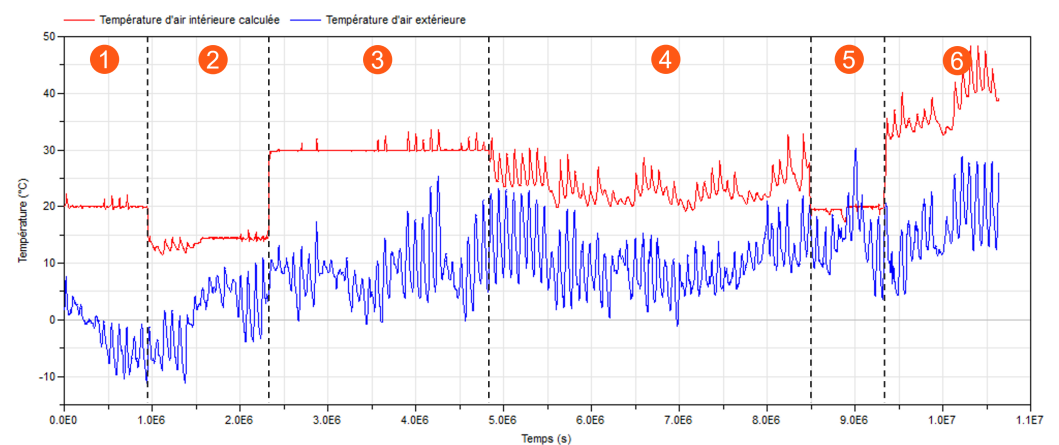
Figure 3. Four month sequence acquired in the experimental cell, comprising six periods: 1. temperature maintained at 20 °C (7 days), 2. 15 °C (15 days), 3. 30 °C (29 days), 4. no air conditioning (42 days), 5. temperature maintained at 20 °C (7 days), 6. heating with a constant power of 160 W (20 days). Red: inside temperature, blue: outside temperature.

Figure 4. 19 time-averaged power measurements from the first period of the experimental data. The time step is approximately 5 h and 30 min.

2.3. Observation Model

We assume in the following that no aleatory uncertainty affects the inputs **X** of our code, since they have been observed throughout the experiment and are thus considered as fixed quantities (covariates). On the other hand, certain model parameters are considered uncertain, including the thermal bridge factor, the albedo, and the convective factor associated with the HVAC system.

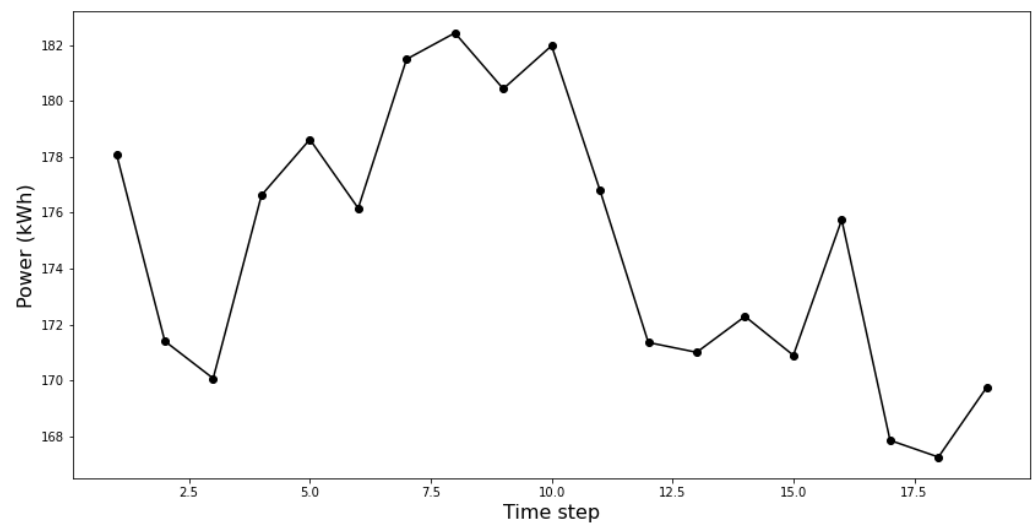The power field measurements are denoted by the vector $\mathbf{Z} = (z_1, \ldots, z_T)$. As discussed earlier, computations are performed based on the averaged power measurements rather than the complete ones, resulting in approximately four measures per day instead of one per 5 min, yielding 30 data over the 7 days. Due to measurement errors, the power measurements z_t are not exactly equal to the true power consumptions P_t. Modeling these errors as a white noise, we have:

$$
\begin{aligned}
z_t &= P_t + \epsilon_t \\
\epsilon_t &\underset{i.i.d.}{\sim} \mathcal{N}(0, \lambda^2),
\end{aligned}
\tag{2}
$$

where ϵ_t is a zero mean Gaussian random variable, with variance λ^2, which can also account for residual variability (Kennedy and O'Hagan 2001) as well as measurement errors.

2.4. Sensitivity Analysis

Since the dimension of $\boldsymbol{\theta}$ is very large ($p = 193$), direct calibration of the thermal code is not feasible. Hence, a preliminary screening step is necessary in order to identify parameters that have a negligible influence on power predictions with the goal of fixing them at the value set by the experts and excluding them from further analyses; see Saltelli et al. (2000). To this end, we have used the results from Bontemps et al. (2013), which we describe now.

In a first step, a local sensitivity analysis was conducted, using a one-at-a-time (OAT) approach, consisting in computing the deviations in the code outputs at each time step when a single parameter was changed by 5 percent around its nominal value. We thought that such a percentage was a good compromise to avoid too small, insignificant changes on the one hand or completely different, unrealistic values on the other. The local sensitivity analysis is performed through three steps, as described below.

1. First of all, the sensitivity indices were defined as the ratios of output over input deviations:

$$
S_i = \frac{\boldsymbol{Y}(\boldsymbol{\theta}^* + h_i \boldsymbol{e}_i) - \boldsymbol{Y}(\boldsymbol{\theta}^* - h_i \boldsymbol{e}_i)}{2h_i} \approx \frac{\partial \boldsymbol{Y}(\boldsymbol{\theta})}{\partial \theta_i} \Big|_{\boldsymbol{\theta} = \boldsymbol{\theta}^*},
$$

where $\boldsymbol{\theta}^*$ is the nominal value for code parameters, elicited from expert opinion, $\boldsymbol{e}_i = (0, \ldots, 1, \ldots, 0)$ is the i-th vector of the canonical orthonormal basis for $\mathbb{R}^p$ and the perturbation h_i is the 5%, in absolute value, of the i-th component of $\boldsymbol{\theta}^*$. $\boldsymbol{S}_i$ is approximately equal to the code's partial derivative with respect to the i-th parameter, computed at each time-step (keeping in mind that $\boldsymbol{Y}(\boldsymbol{\theta})$ is a vector of size T);

2. Next, the mean $S_{i,m}$ and the standard deviation $S_{i,std}$ were computed over time, resulting in a hybrid index $S_{i,d} = \sqrt{S_{i,m}^2 + S_{i,std}^2}$;

3. Finally, only the variables with $S_{i,d}$ larger than a certain threshold underwent uncertainty quantification. By thresholding this indicator, the number of parameters was eventually downsized from $p = 193$ to $p = 3$.

In a second step, a global sensitivity analysis of the remaining parameters was performed, which consisted of computing their respective time-varying Sobol indices, using a Monte-Carlo approach coupled with a polynomial chaos expansion response surface (for further details, see Bontemps et al. (2013)). This second analysis showed that the sum of first-order indices were always higher then 97%, suggesting no interaction between the parameters at all time. Furthermore, the maximum values over time of the first order Sobol indices were used to identify three parameters, with maximum values over 25%, the remaining parameters having maximum Sobol indices below or equal to 7%.

Finally, the three scalar parameters with the greatest impact on the output were found to be:

- $\theta_1 \in [0, 1]$ which is the albedo factor;
- $\theta_2 > 0$ which encodes the effect of the thermal bridges;
- $\theta_3 > 0$ which is the convective factor of the HVAC system.

We will now quantify this impact following a Bayesian approach, specifying prior distributions on these three parameters and use power field measurements and code outputs to make forecast about the electric power supplied to the cell.

3. Calibration

As discussed earlier, the thermal code depends on a vector $\boldsymbol{\theta}$ of physical parameters, typically set to an unchanged value before running the code, for instance a nominal value $\boldsymbol{\theta}^*$ set by experts. However, $\boldsymbol{\theta}$ is often uncertain, typically because it is non-measurable in the field. Code calibration consists of reducing this (epistemic) parametric uncertainty, by identifying parameter values for which the code outputs are as close as possible to physical measurements.

When calibrating, it is important to keep in mind that the code might be an imperfect representation of the physical system. Hence, it should be considered as a more or less accurate mathematical approximation of the thermal behavior inside the experimental cell. This second source of epistemic uncertainty is called *code uncertainty*, and is dealt with in Kennedy and O'Hagan (2001) by adding a discrepancy term in the statistical model used for calibration. We here advocate for another approach, based on a post-hoc statistical test to detect the presence of such a discrepancy between measures and code predictions (see Section 4).

Adopting a Bayesian perspective, calibration requires the following ingredients:

- A statistical model that links the available field measurements $\mathbf{Z}$ with the code outputs. This equation provides a likelihood function $\mathcal{L}(\mathbf{Z}|\boldsymbol{\theta}, \boldsymbol{\psi})$, where $\boldsymbol{\psi}$ is a vector of nuisance parameters attached to the model, specifying for instance the error structure between code outputs and field measurements;
- A prior density $\pi(\boldsymbol{\theta})$ encoding the uncertainty as a prior belief in favor of some values of $\boldsymbol{\theta}$, which are more probable than others, based on expert opinion. If no such prior information is available, a uniform prior distribution can be adopted. Similarly to $\boldsymbol{\theta}$, $\boldsymbol{\psi}$ is endowed with a prior density $\pi(\boldsymbol{\psi})$, which we can choose independently from $\pi(\boldsymbol{\theta})$, meaning that we form a priori independent opinions about the plausible values of both parameters.

The prior uncertainty affecting both code θ and nuisance ψ parameters is then updated according to Bayes' theorem:

$$\pi(\theta, \psi | \mathbf{Z}) = \frac{\mathcal{L}(\mathbf{Z} | \theta, \psi) \pi(\theta) \pi(\psi)}{\int_{\theta, \psi} \mathcal{L}(\mathbf{Z} | \theta, \psi) \pi(\theta) \pi(\psi) \mathrm{d}\theta \mathrm{d}\psi}$$

$$\propto \quad \mathcal{L}(\mathbf{Z} | \theta, \psi) \pi(\theta) \pi(\psi). \tag{3}$$

We now detail the choice of the statistical model and priors.

Statistical Model

Assuming that the model discrepancy is negligible, i.e., the code is a faithful representation of the cell's behavior, we have:

$$\exists \theta_0 \in \mathcal{T} \ s.t. \ \mathbf{Y}(\theta_0) = \mathbf{P}, \tag{4}$$

with $\mathbf{P} = (P_1, \ldots, P_T)$ the sequence of 'true' power consumptions inside the cell, which remain unknown. Then, (2) implies that:

$$\mathbf{Z} = \mathbf{Y}(\theta_0) + \boldsymbol{\epsilon}, \tag{5}$$

where $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_T)$ is defined by (2). We are interested in estimating θ_0 so that, based on (5), we consider the model $\mathbf{Z} = \mathbf{Y}(\theta_0) + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon}$ as in (2). Under this model, Z now depends on θ via $\mathbf{Y}$. Since ϵ_t is a Gaussian white noise, the likelihood is given by:

$$\mathcal{L}(\mathbf{Z} | \theta, \lambda^2) = \frac{1}{(\sqrt{2\pi}\lambda)^T} \exp\left[-\frac{1}{2\lambda^2} SS(\theta) \right],$$

where

$$SS(\theta) = ||\mathbf{Z} - \mathbf{Y}(\theta)||^2$$

is the sum of squares of the residuals between the power field measurements and the code outputs.

3.1. Prior Densities

The only nuisance parameter appearing in our case is the variance λ^2 of observation errors, as defined by (2), hence $\psi := \lambda^2$.

Prior distributions are, in principle, elicited from the experts' opinions about the parameters, but no expert was available at the time of this study. Therefore, we chose the least informative approach to Bayesian analysis, via uniform and Jeffreys priors, which still allows us to exploit the benefits of the framework.

Based on the available information, which was minimal, we defined $\pi(\theta)$ as a product of independent uniform distributions, for which bounds were chosen by thermal modeling experts in order for the parameter values to remain physically plausible. For instance, since the albedo θ_1 is a reflection coefficient, it is necessarily between 0 and 1. Furthermore, a non-informative Jeffreys prior is specified on the variance λ^2, yielding:

$$\pi(\theta) = \pi(\theta_1) \pi(\theta_2) \pi(\theta_3)$$

where

$$\pi(\theta_1) = \frac{\mathbf{1}_{[0,1]}(\theta_1)}{1} \quad \pi(\theta_2) = \frac{\mathbf{1}_{[0,100]}(\theta_2)}{100} \quad \pi(\theta_3) = \frac{\mathbf{1}_{[0,100]}(\theta_3)}{100},$$

and

$$\pi(\lambda^2) \propto \frac{1}{\lambda^2}.$$

3.2. Posterior Distribution

From Bayes' formula (3),

$$\pi(\boldsymbol{\theta}, \lambda^2 | \mathbf{Z}) \propto \mathcal{L}(\mathbf{Z} | \boldsymbol{\theta}, \lambda^2) \pi(\boldsymbol{\theta}, \lambda^2). \tag{6}$$

As $\boldsymbol{Y} : \boldsymbol{\theta} \mapsto \boldsymbol{Y}(\boldsymbol{\theta})$ is non-linear, the posterior distribution (6) has no closed form. It is sampled using a Metropolis–Hastings algorithm; see Robert and Casella (2004). The full conditional distributions can be obtained in a very straightforward way by multiplying the sampling distribution (or likelihood, as a function of the parameters) for the prior of the parameter of interest. As an example, the full conditional of θ_1 is just proportional to the likelihood function restricted to $0 \leq \theta_1 \leq 1$. In practice, calculations were carried out using the algorithm implemented in the Open-TURNS software platform for uncertainty treatment in numerical simulation (Baudin et al. 2017), consisting of component-wise random-walk steps with adaptive lengths, tuned to maintain the acceptance rate within reasonable bounds.

However, the posterior chain generated by this first approach turned out to be highly autocorrelated, especially concerning the observation noise parameter λ^2. This in turn meant that the convergence of the empirical estimates for posterior expectations was extremely slow. Remember that we additionally had to deal with an expensive computer code. This code had to be called four times (one per sampled parameter) at each iteration of the algorithm, for a total running time of 10 s per iteration. In practice, this meant that we were limited to a few thousands iterations, making it crucial to have good mixing properties.

A more refined approach is to integrate out λ^2 and sample only from the joint posterior distribution of the calibration parameters, formally defined as

$$\pi(\theta | \mathbf{Z}) = \int_{\lambda^2} \pi(\lambda^2, \theta | \mathbf{Z}) d\lambda^2.$$

Indeed, in the present case, the above integral can be computed analytically. To see this, note that the Jeffreys prior for the nuisance parameter λ^2, given by $\pi(\lambda) \propto \lambda^{-2}$, can be seen as a limiting case of the inverse-Gamma distribution, defined by the following density function:

$$f(x | \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-(\alpha+1)} e^{-\beta/x}.$$

Namely, it corresponds to $\alpha, \beta \to 0^+$.

As is well known, this prior is at least partially conjugate, given that θ[1] Indeed, it is easily shown that the conditional posterior distribution of λ^2 is the inverse-Gamma distribution:

$$\lambda^2 | \theta, \mathbf{Z} \quad \sim \quad \mathcal{IG}\left(\alpha + \frac{T}{2}, \beta + \frac{1}{2} ||\mathbf{Z} - \boldsymbol{Y}(\boldsymbol{\theta})||^2\right). \tag{7}$$

By integrating out this analytical posterior, we can then recover the partially-integrated joint posterior for the calibration parameters, exploiting Bayes' formula:

$$\begin{aligned}
\pi(\theta | \mathbf{Z}) &= \frac{\pi(\lambda^2, \theta | \mathbf{Z})}{\pi(\lambda^2 | \theta, \mathbf{Z})} \propto \frac{\mathcal{L}(\mathbf{Z} | \lambda^2, \theta) \pi(\lambda^2, \theta | \mathbf{Z})}{\pi(\lambda^2 | \theta, \mathbf{Z})} \\
&\propto \left(\beta + \frac{1}{2} ||\mathbf{Z} - \boldsymbol{Y}(\boldsymbol{\theta})||^2\right)^{-\left(\alpha + \frac{T}{2}\right)}.
\end{aligned}$$

The above expression can then be used as the target density in a MCMC approach, which directly samples θ's posterior density. The full joint posterior distribution of (λ^2, θ) can then be easily recovered by simulating, for each generated posterior value θ_i of the calibration parameters, the corresponding value λ_i^2 from the conditional distribution defined in (7), substituting θ_i for θ.

3.3. Results

We have implemented the MCMC algorithm targeting the marginal posterior density of the calibration parameters, after having integrated out the nuisance parameter. More precisely, the parameter components were updated sequentially, using the random-walk Metropolis–Hastings algorithm with uniform proposals, so that the step sizes correspond to the radii of support intervals. To test the convergence of our algorithm, we ran three chains, initialized using independent draws from the uniform prior distribution. We tuned the step sizes of our component-wise random-walk Metropolis–Hastings algorithm, to obtain acceptance rates between 0.2 and 0.8 for all parameters; the actual values are given in Table 1. Step size adaptation is done automatically within the algorithm each 30 iterations, by computing for each component the current acceptation rate, then increasing the corresponding step size by 10% if the rate is larger than 0.8, and reducing it by 10% if the rate is below 20%.

Each run consisted of 2100 iterations of the algorithm, taking about 11 h on a laptop equipped with an Intel 2.20 GHz processor. Most of the calculation time was due to the computer code, which took about seven seconds per run.

Table 1. Acceptance rates for the component-wise random-walk Metropolis–Hastings algorithm. Each line corresponds to a different MCMC run.

Chain	Parameter	θ_1	θ_2	θ_3
1	Acceptance rate	21%	20%	52%
2	Acceptance rate	26%	71%	58%
3	Acceptance rate	23%	70%	55%

Figure 5 shows the raw Markov chain generated by our algorithm, illustrating the good mixing properties of the chain. Based on these three chains, we computed the Gelman–Rubin convergence diagnostic Gelman and Rubin (1992), removing the 250 first iterations from each *chain* to account for the burn-in period, clearly visible for one of the chains. Table 2 shows the values of this test for each parameter component. These are all very close to 1, meaning that no convergence problem was detected.

Figure 6 confirms the good mixing properties of our algorithm by showing the auto-correlation function associated with each component of the chain: this shows that posterior draws more than ∼10 iterations apart are not significantly correlated. Additionally, Figure 7 shows that we can be reasonably confident that the empirical mean of each Markov chain is close to the actual posterior expectation.

Table 2. Results of the Gelman–Rubin convergence test for each component of the Markov chains.

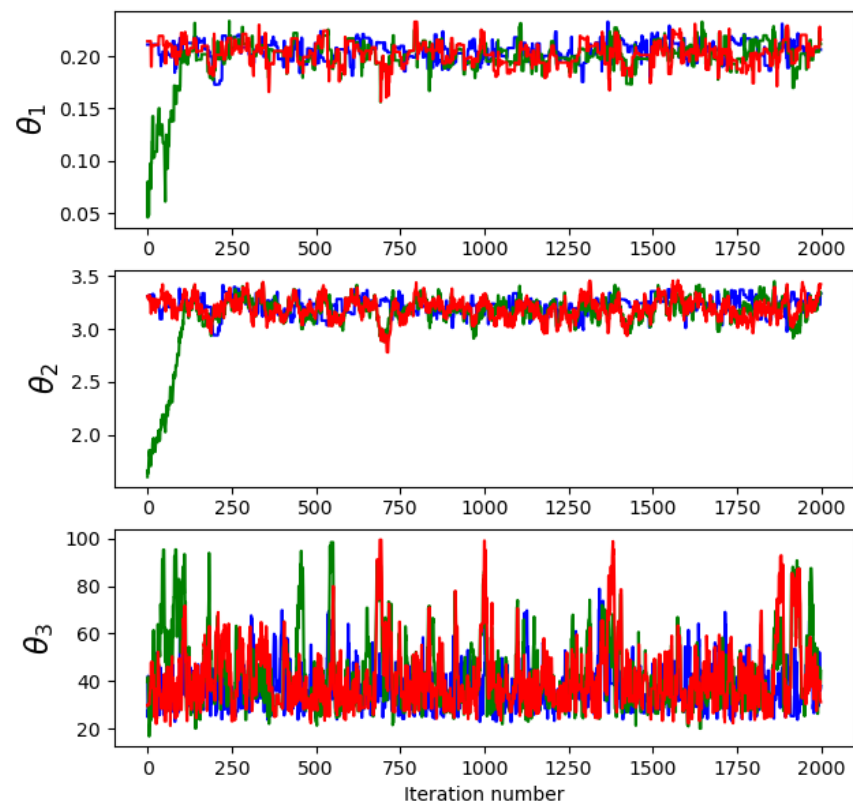| Parameter	θ_1	θ_2	θ_3
Gelman-Rubin test	1.014	1.017	1.018

Figure 5. Raw outputs of the MCMC algorithm: a Markov chain approximating the joint posterior distribution. Each color corresponds to a different MCMC run.
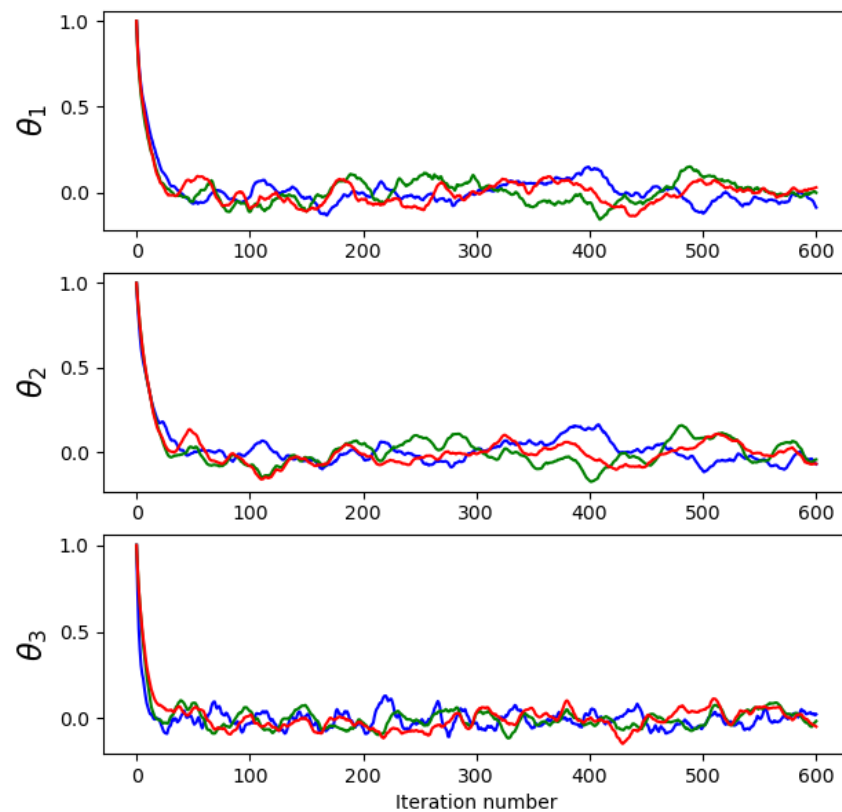

Figure 6. Componentwise autocorrelation of MCMC algorithm output. Each color corresponds to a different MCMC run.
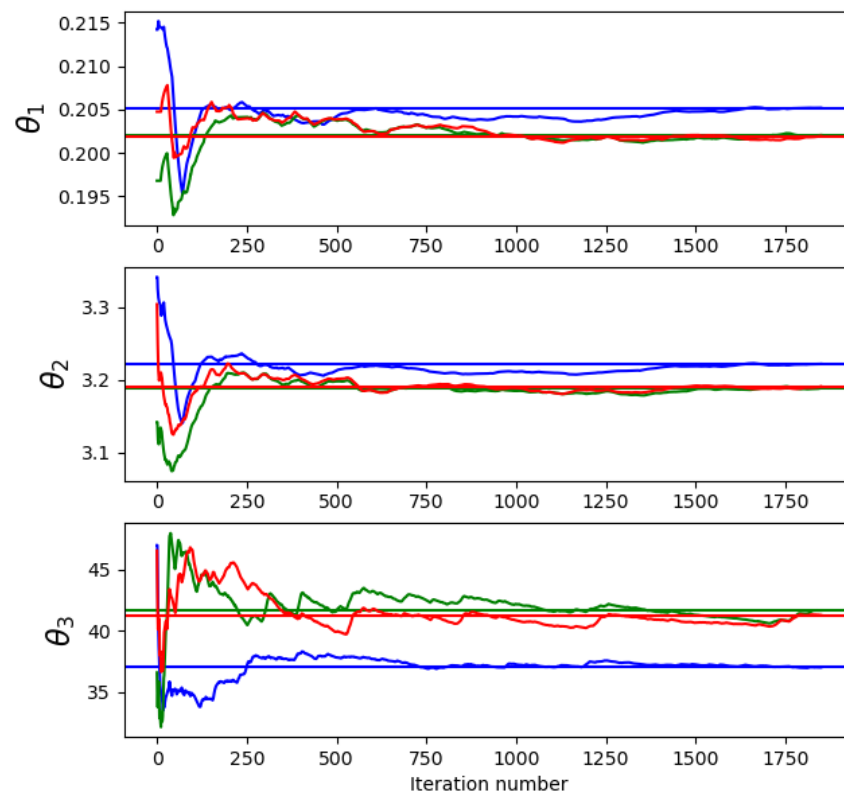
Figure 7. Cumulative ergodic means computed from the MCMC output. Each color corresponds to a different MCMC run.

Taking advantage of all the available computations, we pooled the values from our three Markov chains, having removed the 250 first iterations to account for the burn-in period, resulting in a final posterior sample of $1750 \times 3 = 5250$ draws. Figure 8 shows the densities of the marginal posterior distributions of $\pi(\boldsymbol{\theta}_i|\mathbf{Z})$, while Table 3 summarizes them with the so-called 'central' values (posterior mean and median), a dispersion measure (standard deviation) and 95% posterior credible intervals. These show that the 30 available data allowed to significantly reduce the parametric uncertainty, yielding posterior densities with masses concentrated in realistic regions of the parameter space. For instance, the albedo is seen to be with high probability between 0.18 and 0.22, values usually associated with bare soil and green grass, which happens to be precisely the environment of the BESTLAB platform.

Table 3. Marginal posterior laws characteristics: central values, dispersion metric, and 95% bilateral credible interval (lower/upper credible levels).

Parameter	Mean	Median	LCL	UCL
θ_1	0.203	0.204	0.181	0.222
θ_2	3.200	3.205	2.997	3.377
θ_3	39.960	36.446	24.884	75.378
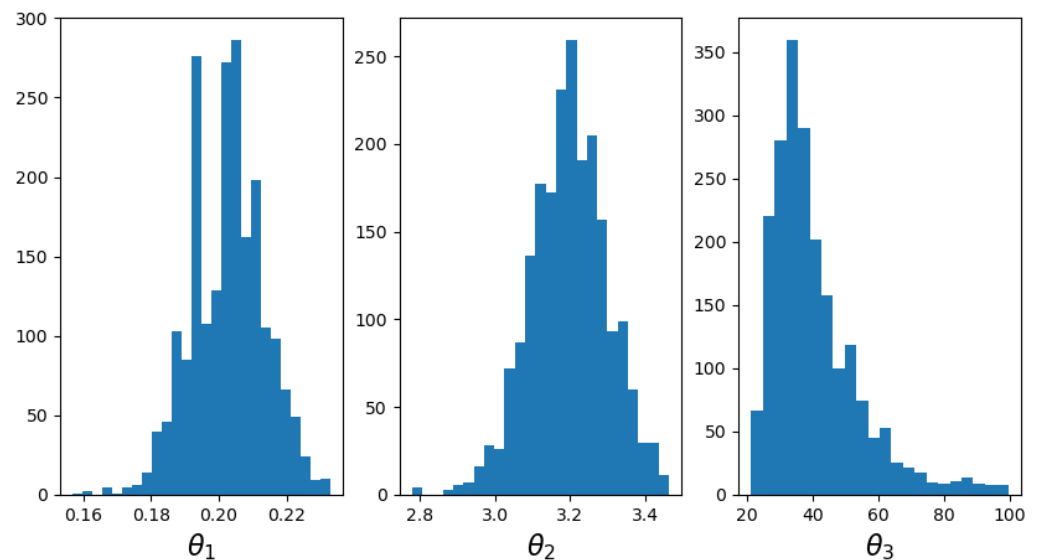
Figure 8. From left to right: marginal posterior densities of θ_1, θ_2 and θ_3.

4. Validation

Validation is the second task of the V&V (Verification and Validation) framework aimed at quantifying the accuracy of code predictions; see AIAA (1998). Verification consists in checking that all bugs inside the source code have been removed and, when the code is constructed as a finite element solution, that the discretization errors are small enough; see Roache (1998). In this paper, we are not concerned with verification although this task should be addressed before any validation study.

The goal of validation is to ensure that the mathematical representation that underlies the code is an acceptable representation of the physical system.

A naive approach, yet still very popular in everyday industrial practice, consists in visually comparing code outputs and field measurements, and then deciding if the difference is small enough. Much effort has been made over the past decade to more rigorously quantify this difference and the uncertainty affecting it. For instance, Roy and Oberkampf (2011) introduced several validation metrics, based on statistical tests taking into account that the computer code can be subject to different types of error, such as intrinsic input variability (aleatory uncertainty), and parametric (epistemic) uncertainty.

Bayarri et al. (2007) proposed a Bayesian validation framework in which the discrepancy between the code and the field measurements is modeled by a random Gaussian process, following the seminal idea of Kennedy and O'Hagan (2001). In our study, the aleatory uncertainty is negligible because $\mathbf{x}_t$ is precisely measured at each time step t in the time period and the associated measurement error is included in the noise term ϵ_t (2). On the other hand, the power predictions are affected both by the value of the code parameters and the adequacy of the code itself for the thermal system. The validation stage is thus strictly dependent on the results of the calibration stage. In Bayarri et al. (2007), the epistemic uncertainty is quantified during the calibration stage, and then propagated to the code output. We will follow the same idea, except for the fact that we do not use a Gaussian process modeling a possible code discrepancy. Indeed, introducing such a term comes at an added computational cost, and may cause identifiability issues, so that in practice the posterior distribution of calibration parameters strongly depends on the choice of a prior for the code discrepancy. This is discussed for instance in Damblin et al. (2016), wherein Bayes factors are computed between calibration models with and without a discrepancy term. We rely instead on a statistical test of the presence of such a discrepancy.

Last but not least, an important point is that a validation study should be carried out by keeping in mind the intended use of the code; see AIAA (1998). Consequently, code predictions should be judged to be sufficiently accurate according to the estimation of a quantity of interest ϕ reflecting its intended use.

In this study, we merely aim to assess the uncertainty affecting the average power delivered inside the cell over the chosen time period. Hence,

$$\phi(\boldsymbol{P}) \quad := \quad \bar{\boldsymbol{P}} = \frac{1}{T}\sum_{t=1}^{T} P_t. \tag{8}$$

It follows that the uncertainty affecting $\bar{\boldsymbol{P}}$ derives from the uncertainty affecting P_t at each time step over the period.

As detailed in Section 2, the validation of the thermal code consists in assessing the uncertainty affecting the average power $\bar{\boldsymbol{P}}$ which is delivered inside the cell over the time period. This process requires the followings steps:

1. Generate a sample $(\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_M)$ from the posterior distribution $\pi(\boldsymbol{\theta}|\mathbf{Z})$, as described in Section 3;
2. Run the code over the M samples $(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M)^2$. This leads to a sample $(\boldsymbol{Y}(\boldsymbol{\theta}_1), \ldots, \boldsymbol{Y}(\boldsymbol{\theta}_M))$ from the posterior distribution $\pi(\boldsymbol{Y}(\boldsymbol{\theta})|\mathbf{Z})$ of the electric power over the time-period. A point estimate can then be derived, such as the posterior mean:

$$\mathbb{E}[\boldsymbol{Y}(\boldsymbol{\theta})|\mathbf{Z}] = \int \boldsymbol{Y}(\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\mathbf{Z})\mathrm{d}\boldsymbol{\theta},$$

which we approximate here by the empirical mean $\frac{1}{M}\sum_{m=1}^{M}\boldsymbol{Y}(\boldsymbol{\theta}_m)$, as illustrated by the blue line in Figure 9;

3. Estimate the posterior distribution of $\pi(\bar{\boldsymbol{P}}|\mathbf{Z})$ by $\pi(\bar{\boldsymbol{Y}}(\boldsymbol{\theta})|\mathbf{Z})$, following (4), a sample of which is given by $(\bar{\boldsymbol{Y}}(\boldsymbol{\theta}_1), \ldots, \bar{\boldsymbol{Y}}(\boldsymbol{\theta}_M))$, where the upper bar denotes the mean over the time period, as defined by (8).

Then, practitioners should decide whether the uncertainty on $\bar{\boldsymbol{P}}$ is not too large in view of its intended use, for instance by calculating the quantiles at 95% of $\pi(\bar{\boldsymbol{P}})$ (see Figure 10, where we see that the average power consumption is most likely between 170 W and 178 W).
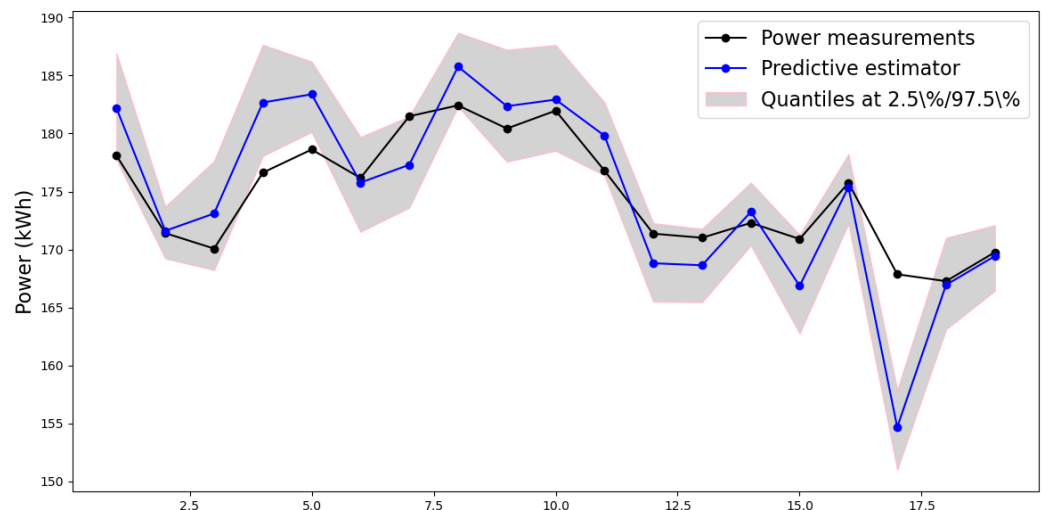

Figure 9. Measurements versus calibrated code predictions of the electric power delivered inside the cell at each time step of the period.
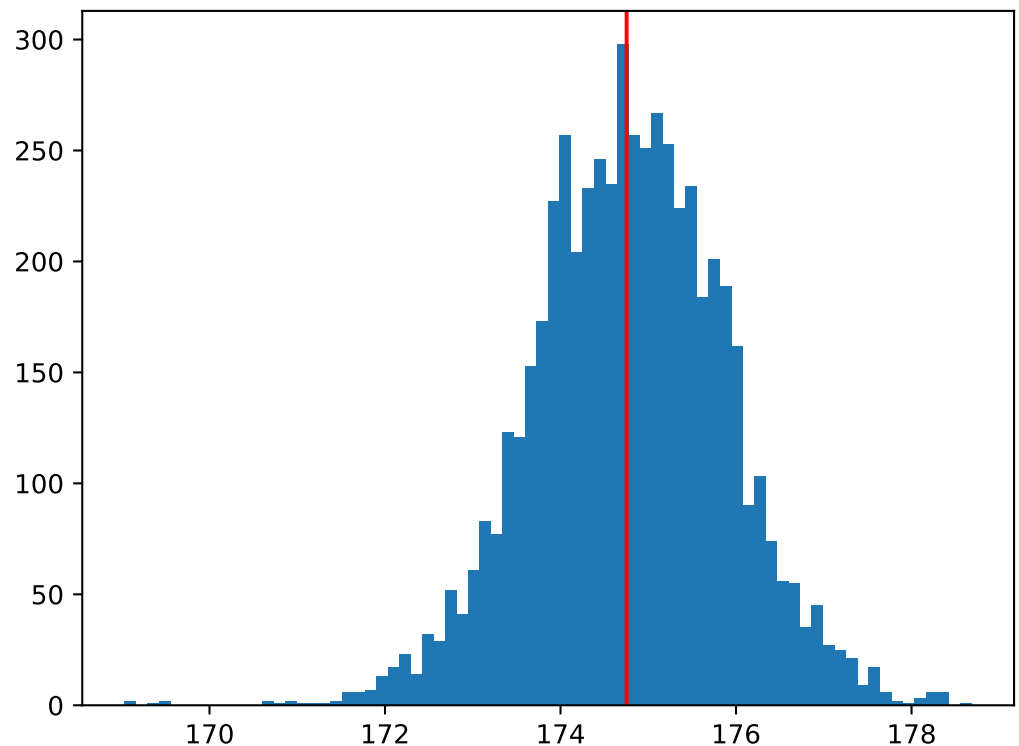
Figure 10. Probability distribution of the averaged electric power (kWh) delivered inside the cell over the time period.

Statistical Testing

To summarize, the above validation procedure consists of a Bayesian calibration step, and two consecutive uncertainty propagation steps, which crucially depend on the calibration result, that is, the probability distribution $\pi(\boldsymbol{\theta}|\mathbf{Z})$, which in turn depends on the statistical model chosen for calibration. In the literature dedicated to code calibration, as we have previously discussed, an alternative to the assumption of unbiasedness of the code with respect to reality (4) is given by:

$$Y(\boldsymbol{\theta}_0) \quad = \quad \boldsymbol{P} + \boldsymbol{b}, \tag{9}$$

where $\boldsymbol{\theta}_0$ is often defined as a 'best-fitting' value (Kennedy and O'Hagan 2001) with respect to the available data. This amounts to saying that there is no value of $\boldsymbol{\theta}$ making a perfect agreement between $\boldsymbol{P}$ and $Y(\boldsymbol{\theta})$. The residual gap $\boldsymbol{b}$ is usually called *code discrepancy*. Joint estimation of $\boldsymbol{\theta}$ and $\boldsymbol{b}$ is performed in Bayarri et al. (2007), despite several conceptual as well as practical difficulties:

- Assumption (9) leads to a statistical model in which only $Y(\boldsymbol{\theta}) + \boldsymbol{b}$ can be estimated; this results in a confounding between $\boldsymbol{\theta}$ and $\boldsymbol{b}$ which is solved in the literature by adopting a Gaussian process (GP) prior distribution on $\boldsymbol{b}$. This means that the estimated value of $\boldsymbol{\theta}$ depends entirely on the prior chosen for $\boldsymbol{b}$, irrespective of the number of available data;
- The definition of the 'real value' $\boldsymbol{\theta}_0$ for the parameter becomes problematic as soon as the code itself is no longer considered an exact depiction of reality;
- Adding the GP term makes model estimation as well as prediction more complex from a purely technical point of view.

All these difficulties may explain why this approach has not yet been widely adopted by the engineering community. As a consequence, we choose here to follow another path, relying on a statistical test of the null hypothesis $\mathcal{H}_0 : \boldsymbol{b} = \boldsymbol{0}$, that is, that our unbiased

statistical model (5) is an accurate enough description of reality, vs. $\mathcal{H}_1 : \boldsymbol{b} \neq \boldsymbol{0}$, meaning that there is a systematic bias in the model predictions, according to (9).

Let us consider, as a goodness-of-fit test statistic, the $\mathcal{X}^2$ discrepancy (Gelman et al. 1996):

$$\mathcal{X}^2(\mathbf{Z}, \boldsymbol{\theta}, \lambda^2) = \frac{||\mathbf{Z} - \boldsymbol{Y}(\boldsymbol{\theta})||^2}{\lambda^2}. \tag{10}$$

The classical testing procedure then consists in computing the following p-value:

$$p_{val}(\boldsymbol{\theta}, \lambda^2, \mathbf{Z}) = \mathbb{P}[\mathcal{X}^2(\mathbf{Z}_{rep}, \boldsymbol{\theta}, \lambda^2) > \mathcal{X}^2(\mathbf{Z}, \boldsymbol{\theta}, \lambda^2) | \boldsymbol{\theta}, \lambda^2, \mathbf{Z}, \mathcal{H}_0],$$

where $\mathbf{Z}_{rep}$ is a vector of replicated data, simulated under the null hypothesis, given here by our statistical model (5). Given a maximum acceptable type I (false positive) error rate α (typically 5%), $\mathcal{H}_0$ is then rejected for values of the p-value smaller then α. If model parameters $\boldsymbol{\theta}$ and λ^2 were perfectly known, and the observation errors normally distributed, then the null distribution would be chi-square, with T degrees of freedom (df). This means that the goodness-of-fit test is also a normality test, in the sense that both the presence of a model bias, and departures from the normality assumptions, can lead to reject $\mathcal{H}_0$.

The usual frequentist way of dealing with the uncertainty tainting $\boldsymbol{\theta}$ and λ^2 is to estimate them from the data $\mathbf{Z}$ through, say, a maximum likelihood procedure. The null distribution is then approached, either by bootstrap techniques (Efron 1981) or asymptotically, leading here to the chi-square distribution with $T - (p + 1)$ degrees of freedom; see van der Vaart (2000).

Since we have adopted a Bayesian perspective, our way to deal with parametric uncertainty is to compute the posterior predictive p-value, as introduced by Gelman et al. (1996), which takes into account the posterior distribution of $\boldsymbol{\theta}$ and λ^2, following:

$$p_B(\mathbf{Z}) = \mathbb{P}[\mathcal{X}^2(\mathbf{Z}_{rep}, \boldsymbol{\theta}, \lambda^2) > \mathcal{X}^2(\mathbf{Z}, \boldsymbol{\theta}, \lambda^2) | \mathbf{Z}, \mathcal{H}_0] \tag{11}$$

where $\mathbf{Z}_{rep}$ is a vector of replicated data, which is now simulated from the predictive density derived from (5). The probability in (11) is therefore computed with respect to the joint posterior density of $(\mathbf{Z}_{rep}, \boldsymbol{\theta}, \lambda^2)$:

$$\pi(\mathbf{Z}_{rep}, \boldsymbol{\theta}, \lambda^2 | \mathbf{Z}) = \mathcal{L}(\mathbf{Z}_{rep} | \boldsymbol{\theta}, \lambda^2) \pi(\boldsymbol{\theta}, \lambda^2 | \mathbf{Z}).$$

This means that $p_B(\mathbf{Z})$ is simply the posterior expectation of the initial p-value:

$$p_B(\mathbf{Z}) = \mathbb{E}[p_{val}(\boldsymbol{\theta}, \lambda^2, \mathbf{Z}) | \mathbf{Z}] = \int_{\boldsymbol{\theta}, \lambda^2} p_{val}(\boldsymbol{\theta}, \lambda^2, \mathbf{Z}) \pi(\boldsymbol{\theta}, \lambda^2 | \mathbf{Z}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\lambda^2,$$

which we approximate here by the empirical mean

$$\widehat{p_B}(\mathbf{Z}) = \frac{1}{M} \sum_m^M p_{val}(\boldsymbol{\theta}_m, \lambda_m^2, \mathbf{Z}),$$

as illustrated in Figure 11 (left). Recall that in our case $p_{val}(\boldsymbol{\theta}_m, \lambda_m^2, \mathbf{Z})$ is given by the probability that a chi-square variate with T df exceeds the observed discrepancy $\mathcal{X}^2(\mathbf{Z}, \boldsymbol{\theta}_m, \lambda_m^2)$. As usual in a Bayesian approach, we deal here with uncertainty on parameters by integrating over them, as opposed to setting them to estimated values, as it would be done in a frequentist setting. Finally, the resulting Bayesian test mimics the original test by rejecting $\mathcal{H}_0$ at a user-chosen level α whenever $p_B(\mathbf{Z}) \leq \alpha$.

Alternatively, $p_B(\mathbf{Z})$ can be estimated using realisations $(\mathbf{Z}_{rep,m}, \boldsymbol{\theta}_m, \lambda_m^2)_{1 \leq m \leq M}$, where the $(\boldsymbol{\theta}_m, \lambda_m^2)$ are given by the calibration procedure, and:

$$\mathbf{Z}_{rep,m} \underset{i.i.d.}{\sim} \mathcal{N}(y_{\boldsymbol{\theta}_m}(.), \lambda_m^2).$$

From (11), a second Monte-Carlo estimate of $p_B(\mathbf{Z})$ is then given by:

$$\widehat{p_B}(\mathbf{Z}) \;=\; \frac{1}{M}\sum_m^M \mathbf{1}\Big\{ \mathcal{X}^2(\mathbf{Z}_{rep,m}, \boldsymbol{\theta}_m, \lambda_m^2) > \mathcal{X}^2(\mathbf{Z}, \boldsymbol{\theta}_m, \lambda_m^2) \Big\}.$$

This estimator is more general than the previous one, since it requires no evaluation of the conditional p-value $p_{val}(\boldsymbol{\theta}, \lambda^2, \mathbf{Z})$, and thus can be used if the latter has no closed form[3].

Figure 11 (right) shows the scatterplot of $\mathcal{X}^2(\mathbf{Z}_{rep}, \boldsymbol{\theta}, \lambda^2)$ against $\mathcal{X}^2(\mathbf{Z}, \boldsymbol{\theta}, \lambda^2)$. By either method, p_B is found equal to 0.45, meaning that there is no evidence to reject model (5) nor the underlying normality assumption.
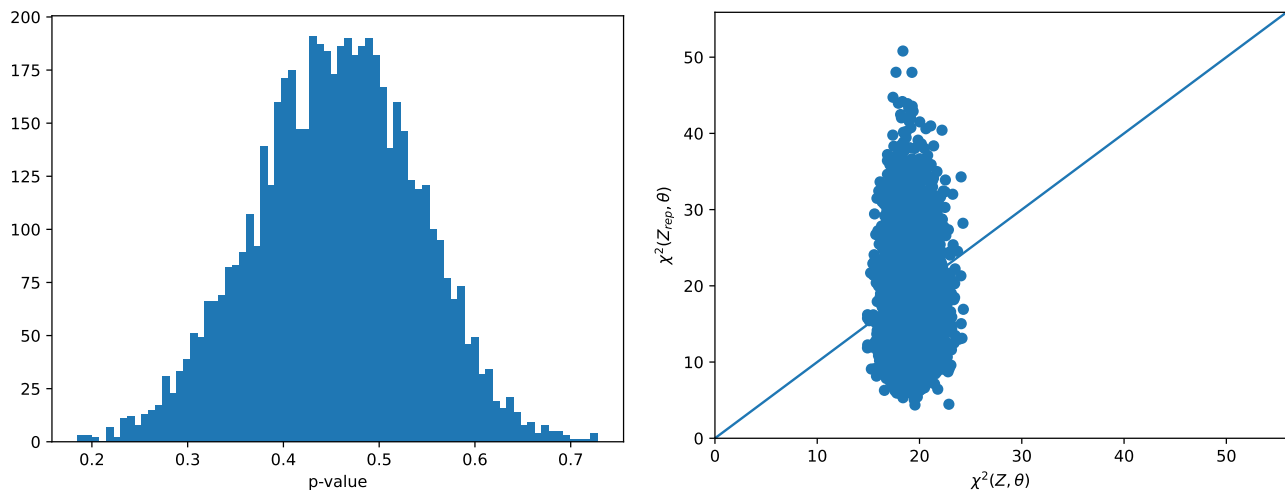

Figure 11. Left: Histogram of the posterior sample of $p_{val}(\boldsymbol{\theta}, \lambda^2, \mathbf{Z})$: the posterior p-value is estimated by the average over this sample. Right: Scatterplot of predictive vs. realized $\mathcal{X}^2$ discrepancies: the posterior p-value is estimated by the proportion of points above the $y = x$ line.

Complimentary to the above test, we have also considered some additional graphics to check the assumption of normality of the observation errors, which are estimated here by the residuals. As shown in Figure 12 (left), we considered the popular QQ-plot, which compares the order statistics of the standardized residuals, to the corresponding standard normal quantiles. However, the residuals depend on model parameters and are hence uncertain. We have summarized this uncertainty by computing the median QQ-plot, as well as its 95% credible bounds. As can be seen, these credible bounds contain the $y = x$ line, meaning that no departure from normality has been detected. In Figure 12 (right), we have also represented the posterior predictive density of the code outputs vs the corresponding observed value. Again, the predicted and recorded values seem reasonably close, except for a single data point, which is clearly underestimated by the code.
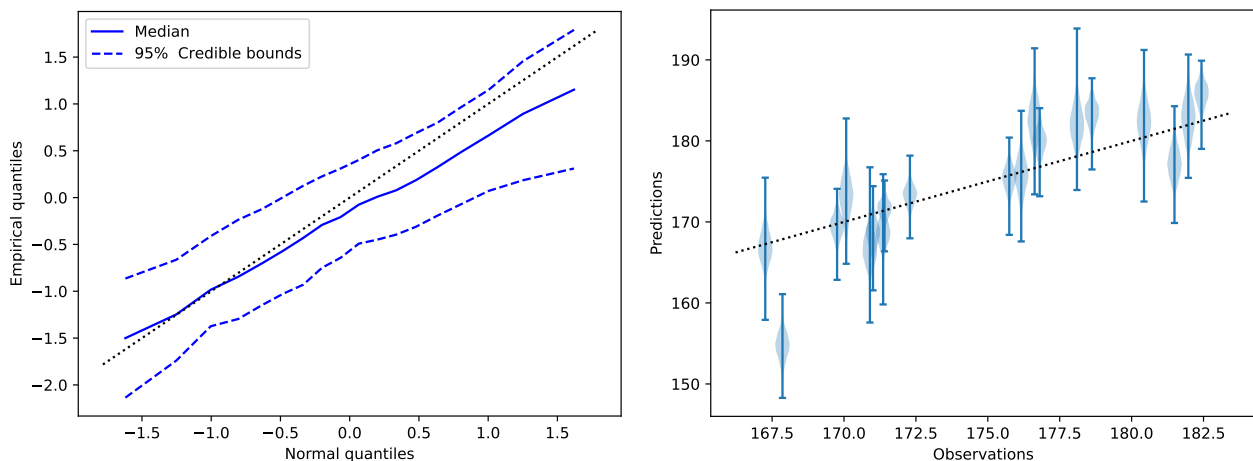
Figure 12. (**Left**): posterior median (solid line) and 95% credible intervals of the normal QQ-plot. (**Right**): Posterior distribution of code outputs versus observed values.

5. Optimal Power Consumption Forecasts

Most energy contracts are based on a description of the building occupied by customers, accounting for all electrical devices inside the building. This description is used to predict the nominal level of power included in the contract, which can be modified later if the effective consumption is far from the initial forecast. In all cases, the monthly bill paid by the customers depends both on the type of contract they have and their effective consumption.

In contrast, it is now common in many other lines of business, such as web access/data services, to propose so-called 'unlimited access' contracts, for which the customers pay a fixed fee, based on an initial estimation of their needs. In this section, we investigate how such energy contracts could be designed, based on probabilistic forecasts, as illustrated in Figure 10.

In short, the problem can be summarized as that of computing a point estimate of the average electric consumption $\bar{Y}(\theta)$ delivered inside the experimental cell over a certain period of time. The standard approach would suggest computing a central value, such as the mean, median, or maximum of the a posteriori predictive distribution; see Berger (1985).

However, such choices do not take into account the underlying stakes, which is hardly possible in an industrial context.

The link between statistical estimation and decision under uncertainty in industrial studies has been carefully studied by Pasanisi et al. (2012), grounded in Bayesian decision theory (described, e.g., in Berger (1985); Bernardo and Smith (1994); French and Insua (2000); Robert and Casella (2004)). We now recall the main ingredients of this approach applied to the field of energetic building modeling.

Bayesian inference provides a measure of the uncertainty about quantities of interest under the form of a probability distribution. If a point estimate is required, a Bayesian estimator should be calculated based both on the probability distribution of the quantity and a cost function, which assesses the economical consequences of all possible estimation errors, that is, the differences between all candidate point estimates and the unknown true value of the quantity. Eliciting the cost function is a complex task, closely related to that of eliciting prior distributions of unknown quantities, as explained in the above references. In practice, obtaining a realistic cost function is often very difficult, so we aim instead for a simplified expression which summarizes the analyst's understanding of the problem. A good example is the LINEX asymmetric loss function Chang and Hung (2007).

More formally, let d be the energy forecast used to define an energy contract for a new customer. The cost function, denoted by $C(d, \bar{P})$, measures the economical consequences induced by the choice of d instead of the true average consumption $\bar{P}$ (of course unknown in advance).

Equivalently, the cost function can be replaced by a utility function as an assessment of the profit instead of the loss. In this case, the associated Bayes estimate $\hat{d}$ maximizes the expected utility:

$$\hat{d} = \underset{d}{\text{argmax}} \int_{\bar{P}} U(d, \bar{P}) \pi(\bar{P}|\mathbf{Z}) d\bar{P}.$$

For illustration purposes, we now describe a possible utility function that can be used to optimize an 'unlimited access' energy contract, assuming that the customer pays a fixed fee d. In the case where the effective consumption $\bar{P}$ exceeds d, the energy supplier commits to pay the surplus amount, thus ensuring customer satisfaction (and hence loyalty). This first goal alone suggests drastically underestimating $\bar{P}$, with the risk of inciting customers to waste the energy they have not paid for. Hence, a company that proposes such a contract should carefully assess its benefits.

To this end, we suggest using the following utility function, based on a linear energy price:

$$U(d, \bar{P}) = (m \times d)\mathbf{1}_{\{\bar{P}>d\}} + \frac{m \times d}{c(d - \bar{P}) + 1}\mathbf{1}_{\{\bar{P}\leq d\}}, \tag{12}$$

where

- m is the marginal electricity price,
- c characterizes the probability $1 - (c(d - \bar{P}) + 1)^{-1}$ that the customer breaks the contract, given that he/she pays more than he/she has consumed. For small values of c, this probability is approximately equal to $c(d - \bar{P})$, hence c can be interpreted as a *customer defection rate*, assuming that the number of defecting customers is proportional to the amount of unused energy they have paid for.

Figure 13 shows the optimal energy forecast $\hat{d}$ corresponding to several values of the defection rate c, given a certain marginal energy price m. Note that, due to our linear price assumption, m simply acts as a scaling factor and does not in fact influence the choice of $\hat{d}$, which is entirely driven by c. Consequently, a small defection rate $c = 1\%$ (solid line) leads to $\hat{d} = 175$, close to the MAP (Maximum A Posteriori) estimate (as seen in Figure 10), while the higher rate of $c = 20\%$ (dot-dashes) leads to the much lower optimal value $\hat{d} = 172.8$, necessarily yielding lower benefits (the dot-dashed curve has a lower maximum than the solid line curve).
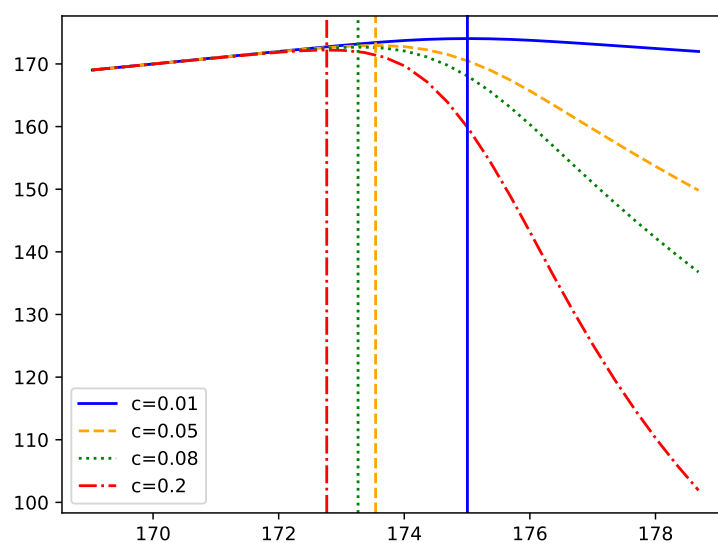

Figure 13. Expected utilities as functions of the fixed contract fee d, for different defection rates c.

These results show how, with the help of the proposed utility function, we can quantify precisely how, in a highly competitive market, companies must reduce their benefits in order to preserve market share. Obviously, such results are highly sensitive to the fine-

tuning of *m* and *c*, which should be carefully assessed with the help of the company's commercial sector. These in turn could be estimated on the basis of existing data, in which case they would be added to the list of uncertain parameters on which the expectation of the utility is computed.

6. Discussion

In this paper, we have proposed a practical Bayesian methodology for the calibration and validation of a computer code used to forecast a quantity of interest in an industrial study, used to guide some strategic decisions for the company, and taking into account the stakes behind these decisions, based on Bayesian decision theory. We illustrated this methodology in a case study concerned with forecasting the energy consumed to heat a building, in view of optimizing an electrical contract including an energy supply guarantee.

We have used a dynamic thermal code implemented with the *Dymola* software, but other computer codes are available, such as the stochastic building performance simulator (S-BPS) considered in Wate et al. (2020), who used a Gaussian process emulator. We have considered a Bayesian approach but others have been proposed in literature; for example, Shamsi et al. (2020) took a fuzzy approach.

The method presented here is quite generic, and can in principle be applied to any application domain, when decisions need to be taken, based on the behavior of a physical system with uncertain outputs.

It is important to note that the three stages of the validation process we propose, i.e.,

- Statistical testing of the code's goodness of fit;
- Parametric uncertainty propagation through the code;
- And posterior utility maximization;

are entirely based on the output of the Bayesian calibration step, and require no additional code runs. This makes our approach very attractive from a computational perspective, since it requires only minimal efforts once calibration has been done.

Nevertheless, the practical implementation of our approach still depends on the different elements at hand. In the simplified setting presented here, the calculations were fairly standard, but many exciting challenges remain to be met in order to apply the proposed framework to more realistic settings, as discussed below.

To begin with, we are currently working on extending the time period over which the code is calibrated, then validated. This raises several issues, even when considering a single cell (rather than a complete building). Indeed, running the code over an extended period increases the computational cost, requiring the use of a metamodel to speed up calculations, such as the dynamic emulator proposed in Liu and West (2009). However, one should be warned about the difficulties of using an emulator in a calibration context. Indeed, this can result in significant distortions of the posterior distribution, even when the pointwise emulation error is small, as shown in Damblin et al. (2018). Dealing with this issue is not straightforward, which is why in the paper we have chosen to work directly with the computer code in spite of the added computation time, as our aim is to demonstrate a simple, robust, and widely applicable methodology.

Since data from different periods correspond to different experimental settings, considering different versions of the code (depending on whether the temperature or the power, or both, are considered outputs of interest), subject to dynamic constraints, is a promising perspective. Another challenge is posed by the albedo. In the paper, this uncertain parameter is assumed constant but it might be more realistic to model it using a time series. This would require more sophisticated calibration procedures.

In the paper we have considered the three parameters identified by Bontemps et al. (2013) and we presented our analysis, given them. As pointed out by one referee, it could have been possible to perform variable selection implementing Bayesian penalization methods such as shrinkage priors. This is an important suggestion that is worthy of future studies.

Furthermore, we have specified some utility functions to commit for an overall consumption forecast to customers according to a new type of energy contract. How to build

them remains a considerable challenge, which should only be addressed with the help of the energy supplier's commercial sector. The simplified functions proposed here should be viewed as guidelines for building more realistic ones.

Finally, the Bayesian test proposed here, which mimics the behavior of a classical test, only allows to control the type I (false positive) error rate, but not the type II (false negative) error rate, meaning that we cannot ensure that the code is valid with a given confidence level. In addition, we have not dealt with the case where the test rejects the null hypothesis, meaning that a significant discrepancy between model and reality has been detected.

Both issues can be overcome in the more general context of Bayesian model selection and averaging, as suggested in Damblin et al. (2016). This would imply comparing the predictions from the statistical model used here, which assumes that the code is a perfect representation of reality, with the statistical model in Kennedy and O'Hagan (2001), which adds a discrepancy term to account for the model errors. This would allow to either select the best model for prediction, or combine predictions from both models, weighted by their respective posterior probabilities given the available data.

In any case, there is a rising trend for the use of numerical simulations to guide industrial choices, in an increasingly competitive market, under higher and higher safety and regulatory constraints. Hence, the issue of assessing model uncertainty and its impact on decision making is becoming a central question. The methodology we have introduced in this paper is a contribution to addressing this challenge, but represents in no way a definitive answer.

Author Contributions: Conceptualization, G.D., P.B., M.K., A.P., F.R. and M.S.; methodology, G.D., P.B., M.K., A.P., F.R. and M.S.; software, G.D. and M.K.; validation, G.D. and M.K.; formal analysis, G.D., P.B., M.K., A.P., F.R. and M.S.; investigation, G.D., P.B., M.K., A.P., F.R., M.S. and E.P.; resources, P.B., M.K., A.P. and M.S.; data curation, G.D. and M.K.; writing—original draft preparation, G.D., M.K. and F.R.; writing—review and editing, M.K. and F.R.; visualization, M.K.; supervision, M.K., F.R. and E.P.; project administration, P.B., M.K., A.P. and M.S.; funding acquisition, P.B., M.K., A.P. and M.S. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The data are not publicly available due to their confidentiality.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Notes

[1] Complete conjugacy is only attained for linear codes, for which the full calibration posterior distribution is explicit.

[2] In fact, these runs are necessarily performed during the calibration step. A good practice is therefore to store all the computer code evaluations done during calibration, to avoid having to do them all over again for the validation.

[3] This generality comes at a cost, since the Monte-Carlo variance associated with this second estimator is systematically higher than that of the first one.

References

AIAA. 1998. *Guide for the Verification and Validation of Computational Fluid Dynamics Simulations.* Reston: American Institute of Aeronautics and Astronautics.

Baudin, Michaël, Anne Dutfoy, Bertrand Iooss, and Anne-Laure Popelin. 2017. Open TURNS: An industrial software for uncertainty quantification in simulation. In *Springer Handbook on Uncertainty Quantification.* Edited by Roger Ghanem, David Higdon and Houman Owhadi. Cham: Springer, pp. 2001–38.

Bayarri, Maria Jesus, James O. Berger, Rui Paulo, Jerry Sacks, John A. Cafeo, James Cavendish, Chin-Hsu Lin, and Jian Tu. 2007. A framework for validation of computer models. *Technometrics* 49: 138–54. [CrossRef]

Berger, James O. 1985. *Statistical Decision Theory and Bayesian Analysis*, 2nd ed. New York: Springer.

Bernardo, Jose Maria, and Adrian Frederick Melhuish Smith. 1994. *Bayesian Theory*. London: Wiley.

Blin, David, Fabrice Casciani, Pierre Imbert, Benjamin Mousseau, Alberto Pasanisi, Pascal Terrien, and Pablo Viejo. 2015. A software platform to help Singapore to build a more smart and sustainable city. Paper presented at Energy Science Technology Conference, Karlsruhe, Germany, May 20–22.

Bontemps, Stéphanie. 2015. Empirical Validation of Models: Application to Low-Energy Buildings. Ph.D. thesis, HESAM University, Paris, France. (In French)

Bontemps, Stéphanie, Aurélie Kaemmerlen, Rémi Le Berre, and Laurent Mora. 2013. La fiabilité d'outils de simulation thermique dynamique dans le contexte des bâtiments basse consommation. Paper presented at Congrés Français de Thermique 2013, Gerardmer, France, May 28–31 .

Campbell, Katherine. 2006. Statistical calibrations of computer simulations. *Reliability Engineering & System Safety* 91: 1358–63.

Chang, Yen-Chang, and Wen-Liang Hung. 2007. LINEX Loss Functions with Applications to Determining the Optimum Process Parameters. *Quality & Quantity* 41: 291–301.

Cox, Dennis D., Jeong Soo Park, and Clifford E. Singer. 2001. A statistical method for tuning a computer code to a data base. *Computational Statistics and Data Analysis* 37: 77–92. [CrossRef]

Damblin, Guillaume, Merlin Keller, Pierre Barbillon, Alberto Pasanisi, and Eric Parent. 2016. Bayesian Model Selection for the Validation of Computer Codes. *Quality and Reliability Engineering International* 32: 2043–54. [CrossRef]

Damblin, Guillaume, Pierre Barbillon, Merlin Keller, Alberto Pasanisi, and Eric Parent. 2018. Adaptive Numerical Designs for the Calibration of Computer Codes. *SIAM/ASA Journal on Uncertainty Quantification* 6: 151–79. [CrossRef]

Eastman, Chuck, Paul Tiecholz, Rafael Sacks, and Kathleen Liston. 2011. *CBIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*, 2nd ed. Hoboken: Wiley.

Efron, Bradley. 1981. Nonparametric Estimates of Standard Error: The Jackknife, the Bootstrap and Other Methods. *Biometrika* 68: 589–99. [CrossRef]

Elmqvist, Hilding. 1978. A Structured Model Language for Large Continuous Systems. Ph.D. thesis, Lund University, Lund, Sweden.

Fonseca, Jimeno A., Ido Nevat, and Gareth W. Peters. 2020. Quantifying the uncertain effects of climate change on building energy consumption across the United States. *Applied Energy* 277: 115556. [CrossRef]

French, Simon, and David Rios Insua. 2000. *Statistical Decision Theory*. London: Wiley.

Gelman, Andrew, and Donald B. Rubin. 1992. Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science* 7: 457–72. [CrossRef]

Gelman, Andrew, Xiao-Li Meng, and Hal Stern. 1996. Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica* 6: 733–807.

Heo, Yeonsook, Ruchi Choudhary, and Godfried A. Augenbroe. 2012. Calibration of Building Energy Models for Retrofit Analysis under Uncertainty. *Energy and Buildings* 47: 550–60. [CrossRef]

Kennedy, Marc C., and Anthony O'Hagan. 2001. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63: 425–64. [CrossRef]

Liu, Fei, and Mike West. 2009. A dynamic modelling strategy for Bayesian computer model emulation. *Bayesian Analysis* 4: 393–412. [CrossRef]

Mirakyan, Atom, Alexandru Nichersu, Alberto Pasanisi, Muhammad Saed, Nico Schweiger, Maria Sipowicz, and Jochen Wendel. 2015. Applied Statistics in Support of Cities Simulation: Some Examples and Perspectives. Paper presented at ENBIS-2015 Conference, Prague, Czech Republic, September 6–10.

Pasanisi, Alberto, and Anne Dutfoy. 2012. An Industrial Viewpoint on Uncertainty Quantification in Simulation: Stakes, Methods, Tools, Examples. In *Uncertainty Quantification in Scientific Computing*. Edited by Andrew M. Dienstfrey and Ronald F. Boisvert. Berlin: Springer, pp. 27–45.

Pasanisi, Alberto, and Joseph Ojalvo. 2008. A multi-criteria decision tool to improve the energy efficiency of residential buildings. *Foundations of Computing and Decision Sciences* 33: 71–82.

Pasanisi, Alberto, Merlin Keller, and Eric Parent. 2012. Estimation of a quantity of interest in uncertainty analysis: Some help from Bayesian Decision Theory. *Reliability Engineering & System Safety* 100: 93–101.

Plessis, Gilles, Aurélie Kaemmerlen, and Amy Lindsay. 2014. BuildSysPro: A Modelica library for modelling buildings and energy systems. Paper presented at 10th International Modelica Conference, Lund, Sweden, March 10–12.

Rivalin, Lisa. 2016. Vers une démarche de garantie des consommations énergétiques dans les bâtiments neufs: Méthodes d'évaluation des incertitudes associées à la simulation thermique dynamique dans le processus de conception et de réalisation. Ph.D. thesis, HESAM University, Paris, France.

Roache, Patrick J. 1998. Verification of codes and calculations. *AIAA Journal* 36: 696–702. [CrossRef]

Robert, Christian P., and George Casella. 2004. *Monte Carlo Statistical Methods*, 2nd ed. Berlin: Springer.

Roy, Christofer J., and William L. Oberkampf. 2011. A comprehensive framework for verification, validation and uncertainty quantification in scientific computing. *Computer Methods in Applied Mechanics and Engineering* 20: 2131–44. [CrossRef]

Rysanek, Adam Martin, and Ruchi Choudhary. 2012. A decoupled whole-building simulation engine for rapid exhaustive search of low-carbon and low-energy building refurbishment options. *Building and Environment* 50: 21–33. [CrossRef]

Sacks, Jerome, William J. Welch, Toby J. Mitchell, and Henry P. Wynn. 1989. Design and analysis of computer experiments. *Technometrics* 31: 41–47. [CrossRef]

Saltelli, Andrea, Karen Chan, and Evelyn Marian Scott. 2000. *Sensitivity Analysis*. New York: Wiley.

Shamsi, Mohammad Haris, Usman Ali, Eleni Mangina, and James O'Donnell. 2020. A framework for uncertainty quantification in building heat demand simulations using reduced-order grey-box energy model. *Applied Energy* 275: 115141. [CrossRef]

Spitz, Clara. 2012. Analyse de la fiabilité des outils de simulation et des incertitudes de métrologie appliquée à l'efficacité énergétique des bâtiments. Ph.D. thesis, Université de Grenoble, Grenoble, France.

Tian, Wei, and Ruchi Choudhary. 2011. Energy use of buildings at urban scale: A case study of London school buildings. Paper presented at Building Simulation 2011: 12th Conference of International Building Performance Simulation Association, Sydney, Australia, November 14–16. pp. 1702–9.

van der Vaart, Aad. 2000. *Asymptotic Statistics*. Cambridge: Cambridge University Press.

Wate, Parag, Marco Iglesias, Volker Coors, and Darren Robinson. 2020. Framework for emulation and uncertainty quantification of a stochastic building performance simulator. *Applied Energy* 258: 11375. [CrossRef]