Meta-Learned Bayesian Optimization for Calibrating Building Simulation Models with Multi-Source Data

Sicheng Zhan
Department of Built Environment
National University of Singapore
szhan@u.nus.edu

Gordon Wichern
Mitsubishi Electric Research Labs,
Cambridge, MA, USA.
wichern@merl.com

Christopher R. Laughman
Mitsubishi Electric Research Labs,
Cambridge, MA, USA.
laughman@merl.com

Ankush Chakrabarty*
Mitsubishi Electric Research Labs,
Cambridge, MA, USA.
chakrabarty@merl.com

Abstract

Well-calibrated building simulation models are key to reducing greenhouse gas emissions and optimizing building performance. Current calibration algorithms do not leverage data collected during previous calibration tasks. In this paper, we employ attentive neural processes (ANP) to meta-learn a distribution using multi-source data acquired during previous calibration. The ANP informs a meta-learned Bayesian optimizer to accelerate calibration of new, unseen tasks. The few-shot nature of our proposed algorithm is demonstrated on a library of residential buildings validated by the United States Department of Energy (USDoE).

1 Introduction

Buildings account for almost 40% of global greenhouse gas emissions (UN Environment, 2020) and model-based control can reduce energy use up to 28% (Nguyen et al., 2014; Drgoňa et al., 2020), showcasing its critical role in the campaign of tackling climate change. Indeed, proper calibration of building simulation models is critical for downstream analysis, control, and performance optimization (Zhan and Chong, 2021). To avoid repeated manual calibration of building simulation models, Bayesian algorithms such as Markov chain Monte Carlo (MCMC) are commonly used (Chong and Menberg, 2018). While providing a promising approach for estimating model parameters and quantifying uncertainty, these methods require a large number of model simulations. This is often impractical, because each simulation is time-consuming. Bayesian Optimization (BO) has recently been proposed as an efficient method for calibration (Chakrabarty et al., 2021a). Classical BO employs Gaussian processes (GP) to learn the parameter $\mapsto$ objective function mapping. GPs are well-known to suffer from poor scalability due to escalation in training complexity with the number of optimization iterations and the dimensionality of the parameter space.

Each optimization-based or sampling-based building calibration task produces a dataset of parameter-objective pairs. These multi-source (different buildings, architecturally, geographically, etc.) datasets are often archived, but seldom used during calibration of a new target building model, since the general assumption is that only data obtained from the target building itself is useful for calibration. That is, current calibration methodologies ignore this highly-relevant, often abundant, archived dataset and perform building calibration 'from scratch' for each new calibration task. This is a missed opportunity in the extreme: *we demonstrate, for the first time, that data obtained during calibration*

*Corresponding author.

Tackling Climate Change with Machine Learning workshop at NeurIPS 2021.

of related, albeit non-identical, buildings often contain useful information about general building dynamics that can significantly accelerate the calibration of new building models.

Meta-learning attempts to mimic human's "learning to learn" process by training a meta (high-level) model that learns distributions of optimization-relevant quantities from previously seen tasks to improve inference quality (Hospedales et al., 2020). It has been applied in many scenarios where it is often impractical to learn everything from scratch, such as hyper-parameter optimization of deep networks (Finn et al., 2017) and few-shot image classification (Ren et al., 2018). This paper proposes the use of meta-learning to learn from multi-source building calibration data to enable few-shot BO-based calibration of unseen building simulation models.

2 Methods

An overview of the proposed methodology is provided in Figure 1. This section describes how we generate source and target tasks to obtain data for meta-learning (in practice, data from source tasks would already be available from archival data), along with how to perform meta learning with attentive neural processes (ANPs) for few-shot calibration via BO.

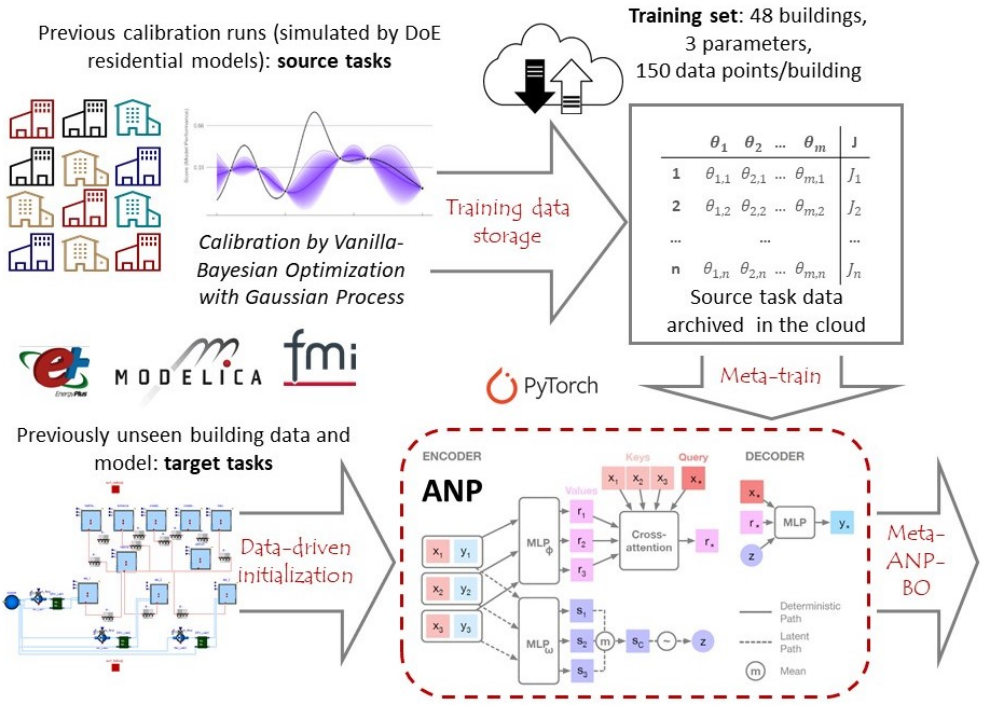


Figure 1: Pipeline for building calibration via ANP-based meta-learned BO.

Dataset Generation: We begin by constructing a set of building simulation models from which optimization-relevant data can be generated for meta-training. To this end, we convert the USDOE validated Energy Residential Prototype Buildings (ERPB) Library (DOE, 2021) into Modelica. The ERPB library comprises a group of EnergyPlus building simulation models for similarly, but not identically, constructed houses located across different climate zones. We convert these EnergyPlus models into Modelica because (i) they can be easily connected to high-fidelity existing models of HVAC and other space heating/cooling components; (ii) Modelica allows seamless integration with Python machine learning toolchains via the Functional Mockup Interface (FMI); and, (iii) the selected models represent the typical geometry of a single-family house in the US while covering a sufficient amount of variability in the materials and constructions required for a rich meta-learning training set.

The resulting data-generating simulation models are constructed using Modelica's Buildings library (Wetter et al., 2014, 2016). We design $4 \times 15 = 60$ simulation models, with 4 unique foundation architectures and 15 unique climate zones. Variations in the foundation construction result

in varying effects from exogenous disturbances and boundary conditions, while altering material properties of the floor of the building. A wide range of climate zones generate buildings that exhibit myriad thermal behavior. In consideration of these variations, our goal is to calibrate 3 model parameters for both source (previously seen) and target (new, unseen) calibration tasks. These include the *external roof solar emissivity*, the *window thermal conductivity*, and the *effective infiltration leakage area of the room*; hereafter, we denote a vector of these parameters by θ. For all calibration tasks, the models were simulated for 72 hours, and room temperatures and relative humidities were measured at 60-min intervals, which is practically feasible. Further details about model construction and calibration is in Appendix A.

We randomly choose 48 of the 60 building simulation models for generating source task data. For each of these 48 tasks, data is collected for 150 GP-BO iterations; see Appendix B for details on GP-BO-based calibration. Similarly, the test dataset is constructed with the 12 remaining building models, although no BO was performed on those. Instead 50 parameter sets are randomly sampled within the admissible range for each parameter, and only random subsets of the data are used for validating our proposed method. Formally, the datasets are described by

$$\mathcal{D}^{\text{train}} := \cup_{k=1}^{48}\{(\theta_t^k, J_t^k)\}_{t=1}^{150}, \quad \text{and} \quad \mathcal{D}^{\text{test}} := \cup_{k=49}^{60}\{(\theta_t^k, J_t^k)\}_{t=1}^{50},$$

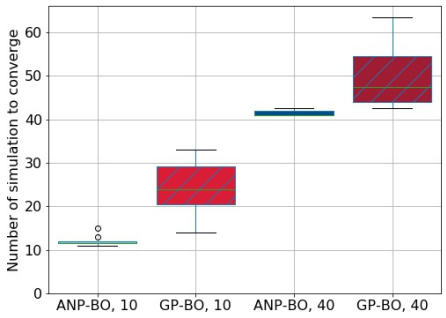
where θ_t^k denotes building parameters and J_t^k the corresponding calibration objective function value. Given our chosen objective function (see (3) in Appendix B), $J_t^k \leq 1$ for any t and k.

ANP-based Meta-Learned BO: Learning from the generated training dataset across multiple tasks calls for two critical properties of the meta-learner: i) being able to model the distribution of objective functions from all tasks while accounting for the uncertainty arising from inter-task variations; and, ii) ensuring scalability to high-dimensional parameter spaces; which could manifest in large number of BO iterations and/or massive quantities of data obtained from the simulation models. Neural Processes (NP) satisfy these requirements through a deterministic encoder that infers the target function given some context points and a latent encoder that models a global latent variable to represent the inter-task uncertainty (Garnelo et al., 2018). However, the original NP applies mean-aggregation to the context points and therefore tends to underfit the target function. Attentive neural processes (Kim et al., 2019) solved this problem by integrating the attention mechanism into the encoders so that the context points can be properly weighted. Furthermore, recent work has demonstrated ANP's utility in building calibration (Chakrabarty et al., 2021b). Consequently, we employ ANP for meta-learning in this work.

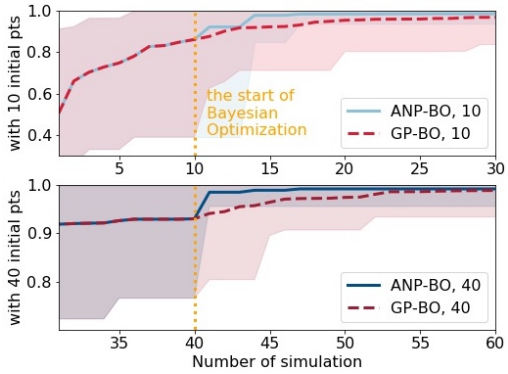
Briefly, for a given task k, ANPs exploit deep neural architectures to estimate a conditional distribution $p(J_\mathcal{T}|\theta_\mathcal{T}, J_\mathcal{C}, \theta_\mathcal{C}, z)$, where $J_\mathcal{C}$ and $J_\mathcal{T}$ are subsets of cost function values partitioned into context and target sets, respectively, located at parameter points $\theta_\mathcal{C}, \theta_\mathcal{T} \subset \Theta$, and z is a latent variable that can be sampled to obtain different realizations of the learned stochastic process. By standard assumptions of Gaussianity, along with statistical arguments commonly seen in variational methods, the training loss function for the ANP can be cast as maximization of an evidence lower bound (ELBO). More details regarding the ANP are provided in Appendix C. A benefit of the ANP method is that the inference of the ANP depends strongly on the context points provided to it: these context points, along with the attention mechanism, ensures tight uncertainty bounds around previously observed data pairs, and high uncertainty in unknown regions of the parameter space, lending itself organically to BO-like frameworks. Consequently, after training on random context/target set partitions for all the tasks in $\mathcal{D}^{\text{train}}$, we employ ANP as a surrogate model for BO on the target calibration tasks. For the k-th target task, in the spirit of few-shot BO (Wistuba and Grabocka, 2021), we select a small subset of (θ, J) from $\mathcal{D}^{\text{test},k}$ and use them as context points for the ANP. Even though this subset is very small (usually 10–40 points for the 3-parameter calibration task), the ANP quickly reconfigures its predictions based on these context points, enabling much more accurate inference of the BO surrogate cost functions compared to vanilla BO. We choose expected improvement with exploration factor $\varepsilon = 0.1$ as an acquisition function. We also use batch-BO as described in Chakrabarty et al. (2021b) to avoid getting trapped in a local minimum. This is to enforce exploration across the admissible parameter space, since we observed that latent sampling alone with few context points did not result in significantly different realizations of the calibration objective function; so non-batch BO always selects candidate parameters in small-radius clusters near the peaks of the acquisition function, leading to poor performance.

3 Results and Discussion

We begin by comparing the performance of meta-learned ANP-BO with the baseline GP-based BO (GP-BO). From Figure 2(a), we observe that ANP-BO achieves successful calibration with many fewer model simulations compared to GP-BO, for both cases when the initial number of context points (equivalently, randomly sampled for GP-BO) were 10 or 40; this was repeated 50 times. In both cases, the GP-BO requires 10–20 more simulations and exhibits higher variance across runs. The improved convergence of meta-learned ANP-BO is further corroborated in subplot (b), where we present the plots of the best incumbent reward value versus BO iterations (note that the first 10, or 40, simulations are the same for both for fairness); statistics are calculated over 10 runs. The convergence rate of ANP-BO increases once random exploration stops and optimization iterations start, especially with 40 initial points. In contrast, GP-BO progressed on a similar pace from the random initialization stage (slightly faster with 40 initial points).



(a) The number of simulation used to converge for all testcases under different BO setups.



(b) The best score so far over iterations of different BO setups (median, continuous line, and interquartile range, shading) across all testcases.

Figure 2: Comparison between the ANP-based and GP-based BO experiment results.

To explain these results, we compare the inference quality of the ANP and GP with varying numbers of context points. As we can see from Figure 6 in Appendix D, having 10 context points is insufficient for GP to approximate the objective function to high accuracy, whereas ANP can leverage its meta-training to infer a correct rough trend of the overall function, even though the accuracy is not much better. Increasing to 40 context points results in the GP approximation being more accurate. However, this comes at the cost of sharp and spiky interpolation at unseen points, owing to small length scales of the kernel. Consequently, the predicted error at around the 65th point leads to an inaccurate understanding of the global maximum. This spiky inference is avoided by the ANP because the inference is somewhat 'regularized' (in some sense) by source task objective function data. This enables ANP to predict a surrogate objective function that is a better reflection of the real objective function, resulting in faster convergence of the calibration mechanism.

4 Conclusions

This paper proposes meta-learned BO for building model calibration based on ANPs. The concept is empirically proven using an open-source USDOE-validated library of residential building models across different climate zones and with different construction types. We show that ANP-BO successfully learned the trends of calibration objective functions for a wide range of building types, resulting in few-shot calibration compared to a baseline GP-BO. We posit that this methodology can tackle climate change using state-of-the-art AI, by promoting model-based optimal building operation and reduction of greenhouse emissions.

References

Chakrabarty, A., Maddalena, E., Qiao, H., and Laughman, C. (2021a). Scalable Bayesian optimization for model calibration: Case study on coupled building and HVAC dynamics. *Energy and Buildings*, page 111460.

Chakrabarty, A., Wichern, G., and Laughman, C. (2021b). Attentive neural processes and batch Bayesian optimization for scalable calibration of physics-informed digital twins. *arXiv preprint arXiv:2106.15502*.

Chong, A. and Menberg, K. (2018). Guidelines for the bayesian calibration of building energy models. *Energy and Buildings*, 174:527–547.

DOE, B. E. C. P. (2021). Residential prototype building models. `https://www.energycodes.gov/prototype-building-models#Residential`.

Drgoňa, J., Arroyo, J., Figueroa, I. C., Blum, D., Arendt, K., Kim, D., Ollé, E. P., Oravec, J., Wetter, M., Vrabie, D. L., et al. (2020). All you need to know about model predictive control for buildings. *Annual Reviews in Control*.

Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., and Teh, Y. W. (2018). Neural processes. *arXiv preprint arXiv:1807.01622*.

Hospedales, T. M., Antoniou, A., Micaelli, P., and Storkey, A. J. (2020). Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*.

Kim, H., Mnih, A., Schwarz, J., Garnelo, M., Eslami, A., Rosenbaum, D., Vinyals, O., and Teh, Y. W. (2019). Attentive neural processes. In *International Conference on Learning Representations*.

Nguyen, A.-T., Reiter, S., and Rigo, P. (2014). A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy*, 113:1043–1058.

Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J. B., Larochelle, H., and Zemel, R. S. (2018). Meta-learning for semi-supervised few-shot classification. In *International Conference on Learning Representations*.

UN Environment, I. E. A. (2020). 2020 global status report for buildings and construction. *Available at https://globalabc.org*.

Wetter, M., Bonvini, M., and Nouidui, T. S. (2016). Equation-based languages–a new paradigm for building energy modeling, simulation and optimization. *Energy and Buildings*, 117:290–300.

Wetter, M., Zuo, W., Nouidui, T. S., and Peng, X. (2014). Modelica Buildings library. *Journal of Building Performance Simulation*, 7:253–270.

Wistuba, M. and Grabocka, J. (2021). Few-shot Bayesian optimization with deep kernel surrogates. *arXiv preprint arXiv:2101.07667*.

Zhan, S. and Chong, A. (2021). Data requirements and performance evaluation of model predictive control in buildings: A modeling perspective. *Renewable and Sustainable Energy Reviews*, page 110835.

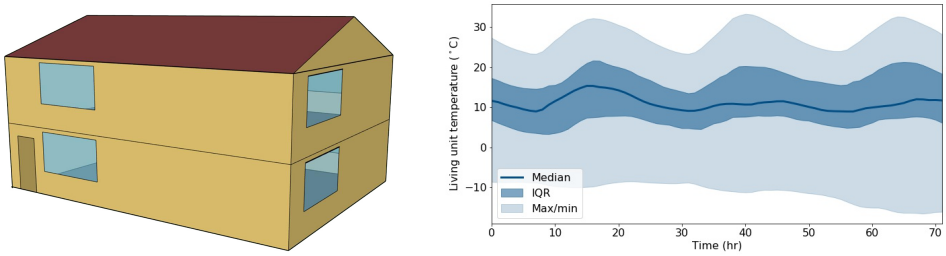
Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code and data are not provided, but implementation details are described in detail.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] e.g. in Fig. 2(a).
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A] All experiments were performed on CPU.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendix

A Building model construction and calibration experiment setup

The original Residential Prototype Buildings Library created by the US Department of Energy (DOE, 2021) is based on the latest International Energy Conservation Code (2018 IECC) and in the form of EnergyPlus models[2]. The models have similar dimensions, consisting of a conditioned two-story-height living unit and an unconditioned attic with inclined roofs (see Figure 3(a)). The floor of the living unit is exposed to one of the four foundation types: slab, crawl space, heated basement, and unheated basement. Each type of building geometry is located in the 15 typical climate zones in the US with varied envelope characteristics, such as the thickness of insulation layers. Figure 3(b) shows the variation across models reflected in 3-day simulation results when there are no internal heat gains or active space conditioning systems.



(a) Geometry of the building model rendered by SketchUp ®.

(b) Median, interquartile range, and min/max of living unit temperatures of the 60 EnergyPlus models over 72 hours.

Figure 3: Comparison between the ANP-based and GP-based BO experiment results.

We replicated the prototype models using the Modelica[3] Buildings library (Wetter et al., 2014). In addition to configuring the model structures and parameters to ensure a correct correspondence, several adjustments were applied to ensure the consistency between the Modelica and EnergyPlus models. First, the default Surface Convection Algorithm `DOE-2` in Energyplus was replaced with the more precise `TARP` (Thermal Analysis Research Program) that calculates the convective heat transfer coefficient with temperature difference and wind speed. Next, year-long simulations were conducted for the EnergyPlus models to generate signals of internal heat gains, which were fed into the Modelica models as boundary conditions. Lastly, the thermostats were disabled to enable the prediction of the free-floating temperatures and thereby validate the building-side model dynamics. Figure 4 illustrates the model outputs for five consecutive days for the purposes of comparison. The remaining little discrepancy between the two models are caused by different use of solvers and underlying calculations, such as the radiative heat transfer coefficient.

The Modelica models were compiled into Functional Mock-up Units[4] (FMUs) to facilitate the calibration experiments. The calibration algorithms or other machine learning frameworks can directly interact with FMUs through `FMPy` on any Windows or Linux machine, requiring no prior knowledge about building physics and the Modelica models. The models were calibrated against the room temperature and relative humidity for the same three consecutive days. Three parameters that are sensitive to these outputs were calibrated. To examine the robustness of the Meta-learned Bayesian Optimization, these parameters were selected as they have an increasing level of variability across buildings: external roof solar emissivity (constant), room effective infiltration leakage area (slightly varied), and window thermal conductivity (varied).

[2]EnergyPlus is a console-based whole building energy simulation program that engineers, architects, and researchers use to model both energy consumption and indoor environment in buildings.

[3]Modelica Language is a non-proprietary, object-oriented, equation based language to conveniently model complex physical systems across many domains, containing, e.g., mechanical, electrical, electronic, hydraulic, thermal, control, electric power or process-oriented subcomponents.

[4]The Functional Mock-up Interface (FMI) is a free standard that defines a container and an interface to exchange dynamic models using a combination of XML files, binaries and C code zipped into a single file.

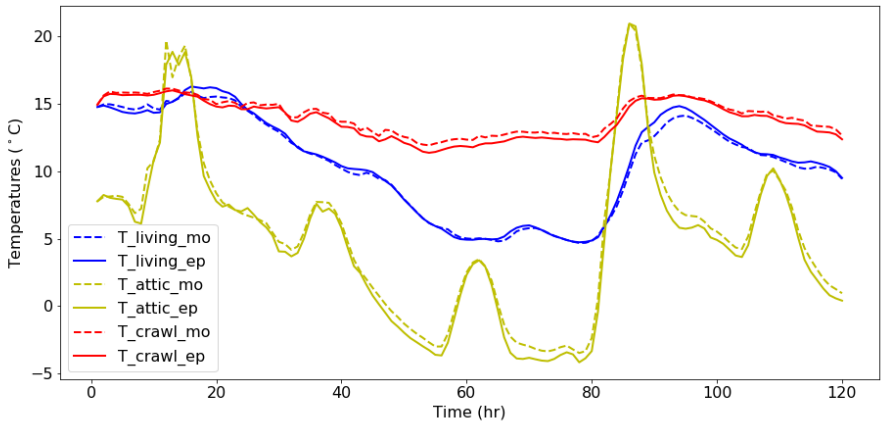


Figure 4: Free-floating temperature predictions of a pair of Modelica and Energyplus models.

B Building calibration with Gaussian Process-based Bayesian Optimization (GP-BO)

This calibration problem can be abstracted by considering a predictive simulation model

$$y_{0:T} = \mathcal{M}_T(\theta), \tag{1}$$

where $\theta \in \Theta \subset \mathbb{R}^{n_\theta}$ denotes the constant parameters used to parameterize the building dynamics. A search domain of parameters Θ is assumed to be available, and we assume Θ is a box in n_θ-space defined by bounded intervals. The output vector $y_{0:T} \in \mathbb{R}^{n_y \times T}$ denotes the outputs that have been measured using the real building sensors over a time-span $[0, T]$. We do not make any assumptions on the underlying mathematical structure of the model $\mathcal{M}_T(\theta)$, except that it has been designed based on building physics, implying that the parameters and outputs are interpretable physical quantities. Simulating $\mathcal{M}_T(\theta)$ forward with a set of parameters $\theta \in \Theta$ yields a vector of outputs $y_{0:T} := [y_0 \quad y_1 \quad \cdots \quad y_t \quad \cdots \quad y_T]$, with $y_t \in \mathbb{R}^{n_y}$.

The generic calibration task is to estimate a parameter set $\theta^\star \in \Theta$ that minimizes (in some goodness-of-fit sense) the modeling error $y^\star_{0:T} - \mathcal{M}_T(\theta^\star)$, where $y^\star_{0:T}$ denotes the measured outputs collected from a real system, and $\mathcal{M}_T(\theta^\star)$ denotes the estimated outputs from the model $\mathcal{M}_T(\theta)$ using the estimated parameters $\theta^\star$. To this end, we propose optimizing a calibration cost function $J(y^\star_{0:T}, \mathcal{M}_T(\theta))$ to obtain the optimal parameters

$$\theta^\star = \arg\min_{\theta \in \Theta} J(y^\star_{0:T}, \mathcal{M}_T(\theta)). \tag{2}$$

Let $\mathcal{D}^{\text{train}}_{k,t} := \{(\theta^k_i, J^k_i)\}_{i=0}^{N_0+t}$ denote the data (parameter/cost pairs) collected up to the t-th iteration of BO for the k-th source task, where N_0 is the size of an initial set of parameter-objective pairs. Calibration with GP-BO involves:

(i) training the GP with $\mathcal{D}^{\text{train}}_{k,t}$ at every t,

(ii) sampling parameters points $\theta_{\mathcal{T}} \subset \Theta$ and evaluating an acquisition function at those points,

(iii) selecting the point that maximizes the acquisition function as the best candidate $\theta^{k,\star}_t$,

(iv) evaluating the cost for $\theta^{k,\star}_t$ and append this pair to $\mathcal{D}^{\text{train}}_{k,t}$, and,

(v) retraining the GP, and repeating from (i) for iteration $t+1$.

In this work, the exponential of the negative mean squared error (MSE) is taken as the objective function (reward), which has a theoretical optimal value of 1 when when $y^\star_{0:T} = \mathcal{M}_T(\theta^\star)$; assuming that the outputs $y^\star_{0:T}$ are realizable with the model $\mathcal{M}_T(\theta)$. Formally, the calibration objective function (reward) used in this paper has the maximizer:

$$\theta^\star = \arg\max_{\theta \in \Theta} \quad \exp(-\text{MSE}(y^\star_{0:T}, \mathcal{M}_T(\theta))). \tag{3}$$

When the MSE approaches 0, the objective value (3) exceeds 0.98; we take this threshold to infer that the optimization has converged and the building model is successfully calibrated.

C Architecture and description of Attentive Neural Processes (ANP)

We borrow the following brief description of ANPs from Chakrabarty et al. (2021b). In the context of Bayesian optimization for digital twin calibration, the ANP (Kim et al., 2019) is a regressor that defines stochastic processes with digital twin parameters serving as inputs $\theta_i \in \mathbb{R}^{n_\theta}$, and function evaluations serving as outputs $J_i \in \mathbb{R}$. Given a dataset $\mathcal{D} = \{(\theta_i, J_i)\}$, we learn an ANP for a set of $n_\mathcal{T}$ target points $\mathcal{D}_\mathcal{T} \subset \mathcal{D}$ conditioned on a set of $n_\mathcal{C}$ observed context points $\mathcal{D}_\mathcal{C} \subset \mathcal{D}$. The ANP is invariant to the ordering of points in $\mathcal{D}_\mathcal{T}$ and $\mathcal{D}_\mathcal{C}$; furthermore, the context and target sets are not necessarily disjoint. The ANP additionally contains a global latent variable z with prior $q(z|\mathcal{D}_\mathcal{C})$ that generates different stochastic process realizations , thereby incorporating uncertainty into the predictions of target function values $J_\mathcal{T}$ despite being provided a fixed context set.

Concretely, given a context set $\mathcal{D}_\mathcal{C}$ and target query points $\theta_\mathcal{T}$, the ANP estimates the conditional distribution of the target values $J_\mathcal{T}$ given by $p(J_\mathcal{T}|\theta_\mathcal{T}, \mathcal{D}_\mathcal{C}) := \int p(J_\mathcal{T}|\theta_\mathcal{T}, r_\mathcal{C}, z)\, q(z|s_\mathcal{C})\, \mathrm{d}z$, where $r_\mathcal{C} := r(\mathcal{D}_\mathcal{C})$ is the output of the transformation induced by the *deterministic* path of the ANP, obtained by aggregating the context set into a finite-dimensional representation that is invariant to the ordering of context set points (*e.g.*, passing through a neural network and taking the mean). The function $s_\mathcal{C} := s(\mathcal{D}_\mathcal{C})$ is a similar permutation-invariant transformation made via a *latent* path of the ANP. The aggregation operator in the latent path is typically the mean, whereas for the deterministic path, the ANP aggregates using a cross-attention mechanism, where each target query attends to the context points $\theta_\mathcal{C}$ to generate $r_{\mathcal{C}\times\mathcal{T}}(J_\mathcal{T}|\theta_\mathcal{T}, r_\mathcal{C}, z)$. Note that the ANP builds on the variational autoencoder (VAE) architecture, wherein $q(z|s)$, $r_\mathcal{C}$, and $s_\mathcal{C}$ form the encoder arm, and $p(J|\theta, r_{\mathcal{C}\times\mathcal{T}}, z)$ forms the decoder arm. The architecture of ANP with both paths is provided in figure 5.

For implementation, we make simplifying assumptions: (1) that each point in the target set is derived from conditionally independent Gaussian distributions, and (2) that the latent distribution is a multivariate Gaussian with a diagonal covariance matrix. This enables the use of the reparametrization trick and we train the ANP to maximize the evidence-lower bound loss $\mathsf{E}\big[\log p(J_\mathcal{T}|\theta_\mathcal{T}, r_{\mathcal{C}\times\mathcal{T}}, z)\big] - \mathsf{KL}\left[q(z|s_\mathcal{T})||q(z|s_\mathcal{C})\right]$ for randomly selected $\mathcal{D}_\mathcal{C}$ and $\mathcal{D}_\mathcal{T}$ within $\mathcal{D}$. Maximizing the expectation term $\mathsf{E}(\cdot)$ ensured good fitting properties of the ANP to the given data, while minimizing (maximizing the negative of) the KL divergence embeds the intuition that the targets and contexts arise from the same family of stochastic processes. The complexity of ANP with both self-attention and cross-attention is $\mathbf{O}\left(n_\mathcal{C}(n_\mathcal{C} + n_\mathcal{T})\right)$. Empirically, we observed that only using cross-attention does not deteriorate performance while resulting in a reduced complexity of approximately $\mathbf{O}\left(n_\mathcal{C} n_\mathcal{T}\right)$, which is beneficial because $n_\mathcal{T}$ is fixed, but $n_\mathcal{C}$ grows with BO iterations.

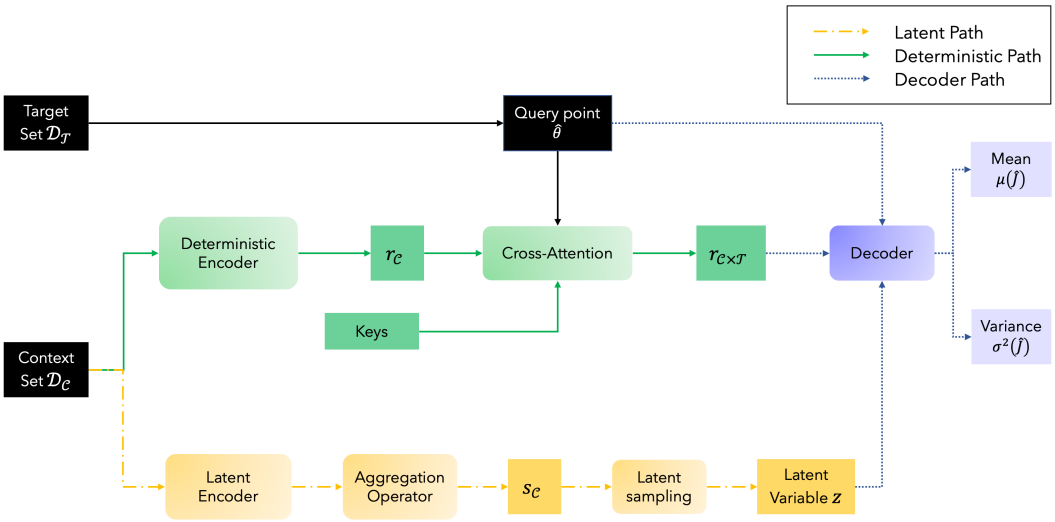


Figure 5: ANP Architecture. Figure source: Chakrabarty et al. (2021b).

D Effect of varying the number of context points

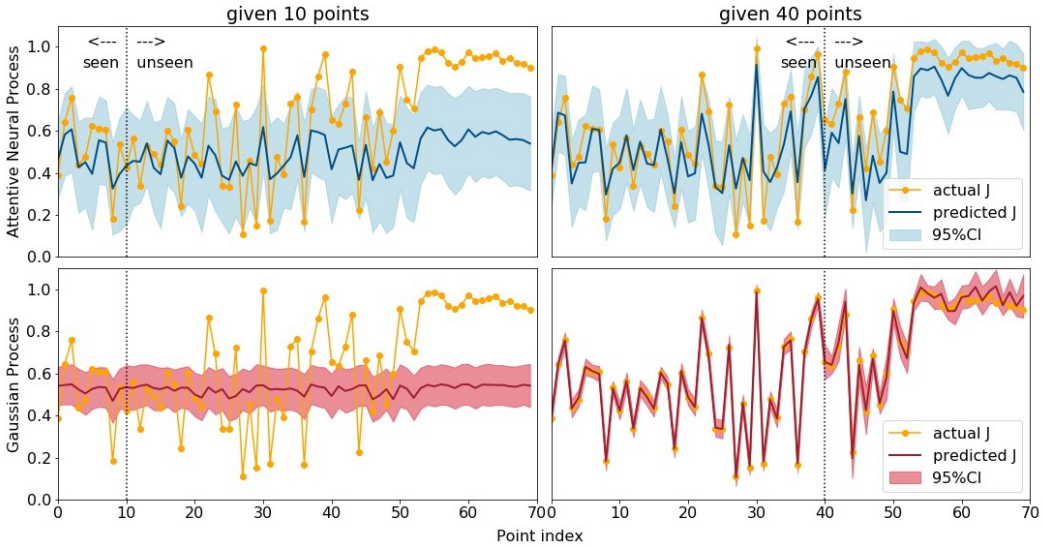


Figure 6: Inference of ANP vs. GP with varying number of context points.