TBMEnv - An Environment to Assess the Accuracy of Thermal Building Models under Realistic Conditions

David Wölfle
FZI Research Center for Information Technology
Karlsruhe, Germany
woelfle@fzi.de

Samed Voßberg
FZI Research Center for Information Technology
Karlsruhe, Germany
vossberg@fzi.de

Hartmut Schmeck
Karlsruhe Institute of Technology
FZI Research Center for Information Technology
Karlsruhe, Germany
hartmut.schmeck@kit.edu

ABSTRACT

Several environments have been published to support the objective evaluation of building energy optimization algorithms. Regarding the evaluation of model-based optimization algorithms, utilizing these environments will likely yield information about the general performance of the planning algorithm or policy but not necessarily regarding the model component of the evaluated approach. The latter follows from inevitable usage of building simulation code within the environment. Observing good performance, it appears reasonable to conclude that the model component of a model-based candidate algorithm is well suited to reflect the simplified physics of the simulation. However, this is at most an indication about the performance of the utilized model in the more complex setting of a real building, as it may be that the model is only capable of representing the simplified physics. To tackle this issue we introduce TBMEnv, an environment to assess the accuracy of thermal building models. TBMEnv implements the task of predicting the development of zone temperatures in two scenarios, each representing an existing office building. The environment is based on measurements recorded in the represented buildings in order to provide a realistic setting for the thermal building models to be evaluated. Besides the environment we present a state of the art thermal model as strong baseline for each scenario. Finally, all code and data are released open source on https://github.com/fzi-forschungszentrum-informatik/TBMEnv

CCS CONCEPTS

• **Software and its engineering → Development frameworks and environments**.

KEYWORDS

building energy management, building control, environment, benchmark, evaluation, reinforcement learning, smart building

ACM Reference Format:
David Wölfle, Samed Voßberg, and Hartmut Schmeck. 2023. TBMEnv - An Environment to Assess the Accuracy of Thermal Building Models under Realistic Conditions. In *The 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '23), November 15–16, 2023, Istanbul, Turkey*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3600100.3625683

BuildSys '23, November 15–16, 2023, Istanbul, Turkey

ACM ISBN 979-8-4007-0230-3/23/11…$15.00
https://doi.org/10.1145/3600100.3625683

1 INTRODUCTION

A large amount of scientific work has been published within the last decades proposing Building Energy Optimization Algorithms (BEOAs), see e.g. [12] for a review. However, objectively comparing the results of such publications is usually not possible with reasonable effort, as the building simulation codes used for evaluation are not identical. To overcome this issue, it has been suggested to establish a shared set of environments for the evaluation of BEOAs [16], which led to a proposal of a development guide for these [17] and finally the publication of several ready-to-use environments for the evaluation of BEOAs, like BOPTEST [1] or Sinergym [6].

In this work the term environment relates to a, usually open source, software that can be utilized to evaluate the performance of algorithms for solving sequential decision problems in a repeatable fashion. The concept originates from research on reinforcement learning and it is thus common that the evaluated algorithm (often called agent) interacts directly with the environment, i.e. the candidate receives an observation (data reflecting the current state of the environment) as well as a reward signal (information how well the optimization target is met) and is expected to compute an action for influencing the environment. Considering the thermal optimization of a building, the environment could provide temperature measurements as observation, a combination of energy costs and thermal comfort as reward while the evaluated BEOA in turn must compute setpoint temperatures as action. Furthermore, the interaction between environment and agent is cyclic, i.e. the environment will compute observation and reward based on the action previously conveyed by the agent. This cyclic interaction pattern between agent and environment implies the necessity that the environment must execute a simulation code in order to quantify the influence of the actions on the state of the represented system (which is required to compute the resulting observation and reward).

Regarding the usage of environments for the evaluation of BEOAs addressed in this work, the utilization of simulation code introduces potential issues on the interpretability of evaluation scores. Consider for example the utilization of a typical environment for the evaluation of a model-based BEOA displayed in Figure 1. The potential issue arises from the fact that the evaluated BEOA will only interact with the environment and hence that the building model

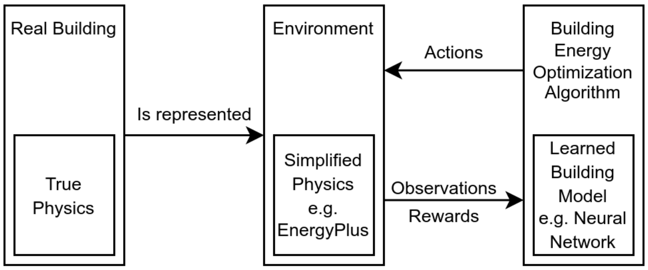

Figure 1: Relationship between real building and a model-based BEOA while using a typical environment for evaluation.

of the BEOA will be trained on data that is generated with simulation code of the environment. However, as the simulation code is inevitably a simplification of the true building physics, using the environment does not provide sufficient information whether the building model of the BEOA will perform well in a real building, too.

In order to (partly) overcome this issue we introduce TBMEnv, an environment that is designed to assess the accuracy of thermal building models under realistic conditions. The environment consists of two scenarios, representing two existing office buildings. In contrast to existing environments, TBMEnv utilizes no simulation code but measured data of the operation of the actual buildings. Consequently, the task of this environment is not to compute actions, but to provide predictions of the measured zone temperatures as accurately as possible. Furthermore, this work presents a strong baseline for each of the two scenarios, i.e. well calibrated state of the art thermal building models that serve as a reference for scores achieved by candidate models that utilize TBMEnv. Finally, instructions describing the intended usage of the environment are provided.

2 SCENARIO 1

The first scenario of the environment utilizes data of a three story office building (with an additional basement) on the campus of a larger German university. The ground, first and second floor of the building are equipped with Concrete Core Activated Ceilings (CCACs) for cooling during summer. More details about the building are provided in [18].

The utilized data have been recorded in 2021, between July 17 and September 30, and contain 15 minute averages of the following quantities:

- Six temperature measurements, two each on ground, first, and second floor.
- The thermal power transferred by the cooling liquid flowing towards the CCAC system.
- Ambient air temperature as well as solar irradiance recorded in close proximity.

It should be noted that the temperature measurements are averaged on floor level. This is done to generate zone temperatures, one zone per floor, which are required by the baseline approach introduced below.

The task of the scenario is to predict the three zone temperatures for the upcoming 96 time steps (24 hours in 15 minute resolution) every 15 minutes (which corresponds to a BEOA algorithm operating with a 24 hour planning horizon every 15 minutes) starting on August 1, 2021. The ground-truth values of cooling power, ambient air temperature, and solar irradiance (for the time range to be predicted) are provided to that end. The corresponding values between July 17 and August 1 are provided as training data.

The main relevance of the scenario is to allow an evaluation whether models are capable of reflecting the latent variables typically encountered (but not observed) in thermal building dynamics, e.g. the temperatures in the walls or the CCAC system. Hand engineered thermal building dynamics models, like the ones promoted in [9] or [18], usually consider latent variables to account for thermal mass in the represented building. Hence it can be expected that models that are not capable to reflect the latent state of the building will not perform well in this scenario.

Finally, it is worth noting that the scenario can be perceived as a test case for thermal building models under ideal assumptions, in particular as perfect forecasts for solar irradiance and ambient temperature values are provided, while in reality these would be subject to a prediction error. Furthermore, the value of the thermal power transferred by the cooling liquid into the cooling system is available, which is often not measured but needs to be inferred from e.g. temperature setpoints and/or energy consumption.

3 BASELINE FOR SCENARIO 1

In this section we introduce the baseline for the evaluation of candidate models on Scenario 1. In particular, we leverage the linear approach utilized in [18] as the model has been developed specifically for the building represented in the scenario. The building is modelled in a state-space representation as:

$$\dot{s_t} = \mathbf{A}s_t + \mathbf{B_1}a_t + \mathbf{B}_2d_t \tag{1}$$

Here[1], $s \in \mathbb{R}^{16}$ is the state vector of the represented building, consisting of three observed (representing the zones on ground, first and second floor) and 13 latent temperature variables while the corresponding state matrix is denoted by $\mathbf{A} \in \mathbb{R}^{16\times16}$. $a \in \mathbb{R}^6$ is the action vector, consisting of six values of which three represent the cooling power flow towards the CCAC systems of the three equipped floors. The remaining three values are the respective quantity for the heating radiators, but always set to zero for the considered scenario. The influence of a on s is represented by the input matrix $B_1 \in \mathbb{R}^{16\times6}$. Similarly, $B_2 \in \mathbb{R}^{16\times8}$ is the disturbance matrix describing the influence of the disturbance vector $d \in \mathbb{R}^8$ on s. The disturbance vector contains heat gains and losses caused by the ambient outside temperature, as well as solar and internal gains. Additional details about the state-space model are omitted here for brevity as the reasoning of the model design can be found in [18] and implementation details are given in the documentation of the source code provided.

In order to utilize the model introduced above in the proposed environment two aspects need to be addressed. First it is worth noting that (1) describes $\dot{s}$, i.e. the first derivative of the state, while the task of the scenario is to predict the zone temperatures at discrete time steps of 15 minutes. While it is generally possible to

[1]Note that, different from here, it is common in literature rooted in control theory to denote the state as x and the action (often also referred to as control vector) as u.

numerically integrate over $\dot{s}$, e.g. by using the well-known *ode45* schema based on the work of [5], we refrain from this approach for the sake of computational performance. Considering performance is especially relevant as the integration schema is executed very often during the tuning step introduced below. Hence, we resort to a simple finite differencing schema. That is, we apply:

$$s_{t+1} \approx s_t + (\mathbf{A}s_t + \mathbf{B}_1 a_t + \mathbf{B}_2 d_t)\Delta t \tag{2}$$

The second aspect to be addressed arises from the fact that the state vector contains latent variables, i.e. only three of the 16 state variables are observed. Thus, it is necessary to estimate the initial state s_0 from the available observations while applying (2) in order to solve the task of the scenario, i.e. to produce the 24 hour forecast of zone temperatures. To this end we apply a *Kalman filter*, which is a common approach in linear control theory [3], that has been applied successfully to building models, e.g. in [9], and which we implement following [15].

Utilizing (2) in combination with the values for A, B_1 and B_2 matrices that have been computed according to [18], the baseline achieves a Mean Absolute Error (MAE) of 3.11 $°C$, respectively a Mean Absolute Percentage Error (MAPE) of 13.6%. However, this is not considered a strong baseline yet as values of the matrices have not been optimized for the measurements utilized in the present scenario. To overcome this issue, we tune the variables of the baseline model using the HEBO [4] black box optimization algorithm, which has been chosen as it has performed well in a recent benchmark [7]. We initially tune the matrices using the training data and repeat the procedure after every 672 forecasts (using all the training and evaluation data observed until that point), i.e. corresponding to a week of building operation in a real world setting. Finally, using the optimized parameters, the baseline performance improves to a MAE of 0.44 $°C$ and respectively a MAPE of 1.9%.

4 SCENARIO 2

The second scenario of the environment utilizes data of a single floor, equipped with a conventional Air Conditioning (AC) system, of an office building in tropical climate. More details about the building are provided in [14].

The available data have been recorded in 2014 and contain 5 minute averages of the following quantities:

- The zone temperature of the air inside the floor.
- The CO_2 of the air inside the floor.
- The supply air temperature of the AC system.
- Ambient outside air temperature recorded by the AC system of the building.

It is worth noting that the represented building has been utilized in an environment before, which is the Tropical Precooling Environment (TPE) introduced in [17]. The task considered by the TPE is to find an optimized operation strategy for the AC system in order to minimize energy costs by leveraging a reduced night time energy tariff. The scenario introduced in the present work takes over this use case, in particular as the building model implemented in the TPE has been optimized for this particular mode of operation and as the building model of the TPE is used as baseline (see below). Thus, the task of the scenario is to predict the zone temperature from 5 am to 5 pm in a 5 minute resolution once per day. To that end, the zone temperatures between 4 am and 5 am as well as the setpoint schedule are provided besides a perfect forecast[2] of the ambient air temperature. Furthermore, probabilistic[3] forecasts for supply air temperature as well as the CO_2 levels are provided. In order to make the scenario as realistic as possible, these are based on well established methods. In particular, we use Prophet [13] to compute predictions for CO_2 and a Gaussian Process (GP) [11] to generate the forecast of the supply air temperature. Further implementation details are provided in the supplementary material section at the end of this work. Using this data, the prediction needs to be carried out for 57 days in spring and early summer while 62 days in late summer and autumn are available as training data. As in the TPE, only workdays are part of the task and the training data to match the occupancy of the represented building.

This scenario is considered to be more challenging compared to Scenario 1 introduced above as data that is usually not available in buildings, like the thermal power towards the cooling system, is not provided. Furthermore, parts of the input data are provided as non-perfect forecasts, which is more realistic but an additional challenge for the building model. That is, compared to the true values, the forecasts have a MAPE of approx. 8.2% for the CO_2 values and 6.0% for the supply air temperature respectively.

5 BASELINE FOR SCENARIO 2

As already discussed in the previous section, the baseline for Scenario 2 reuses the thermal building model utilized in the TPE, which is a gray box model first introduced in [14]. For the sake of brevity, the mathematical formulation of the model is not repeated here. However, a summary of the latter can be found in [17]. The baseline model exposes eight variables that need to be tuned for optimal performance. To this end we utilize the procedure that has been proposed by [14] to fit the model to data. That is, after the true zone temperature values for any day are available, these are used in combination with the predictions for ambient temperature, supply air temperature, and $CO2$ levels to compute the best fitting variables using a least squares optimization algorithm. Once a forecast needs to computed with this model, the variables will be set to the average of the N_a last days, where [14] has used $N_a = 5$ but we we use $N_a = 1$ as this significantly improves predictive accuracy. Utilizing this approach the baseline achieves a MAE of 0.37 $°C$, respectively a MAPE of 1.5%.

6 PERFORMANCE MEASURE

Following the suggestion in [17], i.e. in order to improve the comparability between evaluated candidates, the environment computes a performance measure after the tasks of the scenarios have been completed. This performance measure is the MAE, i.e. the absolute prediction error averaged over all forecasts provided by the candidate model. We have chosen a linear error weighting, in contrast to the frequently used quadratic error, as thermal comfort as well as energy costs are expected to have an approximately linear relation to temperature.

[2]The ambient temperature is provided as perfect forecast as it was not possible to retrieve a realistic, i.e. historic, forecast for the represented location with reasonable effort.

[3]The forecasts contain the mean value as well as the upper and lower boundaries of the 95% confidence interval.

7 INTENDED USAGE OF THE ENVIRONMENT

In order to make the environment easily usable for the community the interface follows the pattern from OpenAI's Gym [2, 10]. However, some minor adaptions were required as TBMEnv expects the evaluated algorithm to provide predictions of state variables instead of actions. The intended usage of the environment is as follows:

```
from tbmenv import Scenario1
env = Scenario1()
model = Candidate()
model.observe(*env.get_training_data())
model.observe(*env.reset())
done = False
while not done:
    predicted_states = model.predict()
    o, a, d, done = env.step(predicted_states)
    model.observe(o, a, d)
score = env.compute_performance_measure()
```

Here o, a and d are the observations, actions and disturbances emitted by the environment while score holds the result of the evaluation, i.e. the MAE of predictions. The code above should work for any Candidate that follows the API pattern implemented by the provided baselines. Hence, executing the baseline is as simple as replacing Candidate with Baseline1 (which can be imported from tbmenv too). Further details can be found in the source code provided.

SUPPLEMENTARY MATERIAL

Source Code

The source code of TBMEnv including further documentation can be found at: https://github.com/fzi-forschungszentrum-informatik/TBMEnv

Implementation Details

We implemented a time linear Kalman Filter to estimate the initial state of Baseline 1. Due to the absence of measured data for all state values, the zonal readings for the respective floor are used as first guess. We assume the measurement noise R to be uncorrelated with a standard deviation of $0.05°C$. Finally, the process noise Q is modeled as white noise with zero mean and standard deviation of $0.0023°C$ following the suggestions in [8].

We have used the prophet code provided by facebook in version 1.1.4 to generate the CO_2 predictions. We have used the default settings apart from flat growth, zero change points, and disabled yearly seasonality in order to prevent the model from overfitting a non existing year scale pattern.

We have used the GPy package with version 1.10.0 to compute the predictions of the supply air temperature. We implemented a GP with a constant mean module and a squared exponential kernel. Besides the perfect forecast of ambient air temperature the minutes since midnight were provided as features.

ACKNOWLEDGMENTS

The authors would like to thank the Institute for Automation and Applied Informatics of the Karlsruhe Institute of Technology for providing the data and information about the building that has been used as blueprint for Scenario 1.

This research has partly been funded by the German Federal Ministry for Economic Affairs and Climate Action within the projekt "Flexkälte – Flexibilisierung vorhandener Kälteanlagen und deren optimierter Einsatz in einer Realweltanwendung".

REFERENCES

[1] David Blum, Javier Arroyo, Sen Huang, Ján Drgoňa, Filip Jorissen, Harald Taxt Walnum, Yan Chen, Kyle Benne, Draguna Vrabie, Michael Wetter, and Lieve Helsen. 2021. Building Optimization Testing Framework (BOPTEST) for Simulation-Based Benchmarking of Control Strategies in Buildings. *Journal of Building Performance Simulation* 14, 5 (Sept. 2021), 586–610. https://doi.org/10.1080/19401493.2021.1986574

[2] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI Gym. arXiv:1606.01540 [cs]

[3] Steven L. Brunton and J. Nathan Kutz. 2019. *Data Driven Science & Engineering: Machine Learning, Dynamical Systems, and Control.* Cambridge University Press, Cambridge New York Port Melbourne New Delhi Singapore. https://doi.org/10.1017/9781108380690

[4] Alexander I. Cowen-Rivers, Wenlong Lyu, Rasul Tutunov, Zhi Wang, Antoine Grosnit, Ryan Rhys Griffiths, Alexandre Max Maraval, Hao Jianye, Jun Wang, Jan Peters, and Haitham Bou-Ammar. 2022. HEBO: Pushing The Limits of Sample-Efficient Hyper-parameter Optimisation. *Journal of Artificial Intelligence Research* 74 (July 2022), 1269–1349. https://doi.org/10.1613/jair.1.13643

[5] J. R. Dormand and P. J. Prince. 1980. A Family of Embedded Runge-Kutta Formulae. *J. Comput. Appl. Math.* 6, 1 (March 1980), 19–26. https://doi.org/10.1016/0771-050X(80)90013-3

[6] Javier Jiménez-Raboso, Alejandro Campoy-Nieves, Antonio Manjavacas-Lucas, Juan Gómez-Romero, and Miguel Molina-Solana. 2021. Sinergym: A Building Simulation and Control Framework for Training Reinforcement Learning Agents. In *Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '21).* Association for Computing Machinery, New York, NY, USA, 319–323. https://doi.org/10.1145/3486611.3488729

[7] Juha Kiili, Eero Laaksonen, Ryan Turner, David Eriksson, Serim Park, Mike Mccourt, Zhen Xu, and Isabelle Guyon. [n.d.]. BBOChallange. https://bbochallenge.com/.

[8] MathWorks Inc. [n.d.]. Tuning Kalman Filter to Improve State Estimation. https://de.mathworks.com/help/fusion/ug/tuning-kalman-filter-to-improve-state-estimation.html.

[9] Frauke Oldewurtel, Alessandra Parisio, Colin N. Jones, Dimitrios Gyalistras, Markus Gwerder, Vanessa Stauch, Beat Lehmann, and Manfred Morari. 2012. Use of Model Predictive Control and Weather Forecasts for Energy Efficient Building Climate Control. *Energy and Buildings* 45 (Feb. 2012), 15–27. https://doi.org/10.1016/j.enbuild.2011.09.022

[10] OpenAI. 2020. Openai/Gym: A Toolkit for Developing and Comparing Reinforcement Learning Algorithms. https://github.com/openai/gym.

[11] Carl Edward Rasmussen and Christopher K I Williams. 2005. *Gaussian Processes for Machine Learning.* MIT Press, London, England.

[12] Pervez Hameed Shaikh, Nursyarizal Bin Mohd Nor, Perumal Nallagownden, Irraivan Elamvazuthi, and Taib Ibrahim. 2014. A Review on Optimized Control Systems for Building Energy and Comfort Management of Smart Sustainable Buildings. *Renewable and Sustainable Energy Reviews* 34 (June 2014), 409–429. https://doi.org/10.1016/j.rser.2014.03.027

[13] Sean J Taylor and Benjamin Letham. 2017. *Forecasting at Scale.* Preprint. PeerJ Preprints. https://doi.org/10.7287/peerj.preprints.3190v2

[14] Arun Vishwanath, Vikas Chandan, Cameron Mendoza, and Charles Blake. 2017. A Data Driven Pre-cooling Framework for Energy Cost Optimization in Commercial Buildings. In *Proceedings of the Eighth International Conference on Future Energy Systems (E-Energy '17).* Association for Computing Machinery, New York, NY, USA, 157–167. https://doi.org/10.1145/3077839.3077847

[15] Greg Welch and Gary Bishop. 2006. An Introduction to the Kalman Filter. https://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf.

[16] David Wölfle, Kevin Förderer, and Hartmut Schmeck. 2019. A Concept for Standardized Benchmarks for the Evaluation of Control Strategies for Building Energy Management. *Energy Informatics* 2, Suppl 2:P3 (2019).

[17] David Wölfle, Arun Vishwanath, and Hartmut Schmeck. 2020. A Guide for the Design of Benchmark Environments for Building Energy Optimization. In *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '20).* Association for Computing Machinery, New York, NY, USA, 220–229. https://doi.org/10.1145/3408308.3427614

[18] Philipp Zwickel, Alexander Engelmann, Lutz Gröll, Veit Hagenmeyer, Dominique Sauer, and Timm Faulwasser. 2019. A Comparison of Economic MPC Formulations for Thermal Building Control. In *2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe).* 1–5.