

Regression-Based Model Error Compensation for a Hierarchical MPC Building Energy Management System

Thomas Schmitt, Jens Engel, Tobias Rodemann

2023

Preprint:

This is an accepted article published in 7th IEEE Conference on Control Technology and Applications (CCTA) 2023. The final authenticated version is available online at: <https://doi.org/10.1109/CCTA54093.2023.10252861>
Copyright 2023 IEEE

Regression-Based Model Error Compensation for a Hierarchical MPC Building Energy Management System

Thomas Schmitt¹, Jens Engel¹, Tobias Rodemann¹

Abstract—One of the major challenges in the development of energy management systems (EMSs) for complex buildings is accurate modeling. To address this, we propose an EMS, which combines a Model Predictive Control (MPC) approach with data-driven model error compensation. The hierarchical MPC approach consists of two layers: An aggregator controls the overall energy flows of the building in an aggregated perspective, while a distributor distributes heating and cooling powers to individual temperature zones. The controllers of both layers employ regression-based error estimation to predict and incorporate the model error. The proposed approach is evaluated in a software-in-the-loop simulation using a physics-based digital twin model. Simulation results show the efficacy and robustness of the proposed approach.

Index Terms—data-driven residual estimator, energy management system, digital twin, co-simulation, building control

I. INTRODUCTION

The increasing penetration of renewable energy sources (RESs) in the public power grid leads to a demand for intelligent energy management systems (EMSs) for buildings. The most popular method for controlling EMSs is Model Predictive Control (MPC). However, for MPC to be effective, an appropriate model of a building's energy behavior is necessary.

There are several approaches to building such a model, which can be categorized as white-box modeling, gray-box modeling, and black-box modeling. White-box models, mostly developed using building energy performance simulation tools such as EnergyPlus or TRNSYS, can be very accurate, but are usually too complex to be used directly in the MPC's optimal control problem (OCP). Grey-box models, such as state space or Resistor-Capacitor (RC) models, are less accurate, but can be utilized well in an OCP [1]. Both white- and gray-box modeling of buildings is very complex and requires building-specific expert knowledge, i.e. models cannot be easily transferred to other buildings [1]. Thus, data-driven black-box modeling has experienced an increase in interest [2], e.g. using Gaussian Processes (GPs) or artificial neural networks (ANNs). While the biggest advantage is the comparatively low modeling effort, they require a large amount of data, the acquisition of which is again challenging [3]. At the same time, including possibly known dynamics or behavior are difficult to incorporate directly and may also have to be approximated. Therefore, a hybrid

approach of these modeling paradigms is likely necessary to succeed in employing building EMSs in a larger scale in the real world.

One option is to replace (a part of) the building's model by a *data-driven surrogate model*. In [4], [5], a machine learning model is trained with simulation data from a physics-based model. Then, the machine learning model is included in the MPC's OCP. Data-driven surrogate models are also frequently used for real-world buildings. In [6], ANNs are trained with historical data from a test building located at the University of L'Aquila, Italy to predict both energy consumption and temperature development. The ANNs are then utilized as the sole model in the MPC. In [7], recurrent neural networks are used to approximate a nonlinear thermal model of an airport check-in hall. The check-in hall's temperature is then controlled using MPC to both follow a reference trajectory and not violate comfort boundaries by solving a linear OCP. However, ANNs can also be used as part of the objective function, instead of replacing model equations in the constraints. In [8], radial basis function (RBF)-based ANNs are used to approximate both the thermal dynamics and the occupant comfort for 4 university office rooms. For more examples of data-driven control approaches, the reader is referred to the review [9]. Notably, only very few studies consider multi-zone buildings.

A second option for a hybrid model approach is a *data-driven error estimator* (or *residual estimator*). Here, the goal is not to replace a part of the gray-box model, but to reduce the model error by augmenting it with a residual value, estimated by a data-driven regression model. However, applications in the building sector are sparse. In [10], a physics-based model of a single-office building in Stuttgart, developed in TRNSYS and MATLAB, is first simplified to a RC gray-box model. Then, a GP model is trained to predict the error of the RC model, using simulation data from the physics-based model as ground truth. However, it was not applied to any control purposes. Applications of error estimators in combination with MPC can be found in different areas. In [11], GPs are used to learn the model error for an autonomous racing car. Training data is received from simulation without MPC. The GPs are explicitly used in the OCP as part of the model dynamics. In [12], a RBF-based disturbance estimator for a nonholonomic robot is used for event-triggered MPC. The disturbance is assumed to be dependent on the system state and control input only, and could thus be interpreted as a model error. In [13], a GP based

¹Honda Research Institute Europe GmbH, Offenbach, Germany. E-mail: {thomas.schmitt, jens.engel, tobias.rodemann}@honda-ri.de

error estimation is combined with an extended Kalman filter to achieve offset-free tracking of a 6 degrees of freedom robotic arm.

In this work, we use a hierarchical setup for the MPC of the energy system of a medium-sized office building in Offenbach, Germany. An aggregator is used to control the total energy flows, which are then allocated to the 9 individual temperature zones by a distributor. Gray-box state space models are used on both levels. A physics-based digital twin serves as a surrogate model of the actual building. To compensate the model errors of both the aggregator and the distributor, we train two regression-based error estimators. As features, only signals which are easily obtainable both online and offline are used. Training data is derived from a software-in-the-loop (SiL) simulation of the digital twin with real-world measurement data. The main contributions are the development of the data-driven estimators for a multi-zone building using a digital twin and real-world measurement data, and their application for error compensation in a hierarchical MPC approach.

The rest of the paper is structured as follows. The building itself as well as its digital twin and the simplified gray-box models are described in Section II. The hierarchical MPC setup is explained in Section III. The data-driven error estimators and their training process is discussed in Section IV. The successful error compensation by combining the error estimators with the hierarchical MPC approach is shown by long-term simulation results in Section V. Finally, we conclude with a discussion on the impacts and necessary further steps in Section VI.

II. BUILDING MODELS

In this section, we will first give a brief description of the actual building. Then, we will explain the different models used in this study, i.e. 1) the digital twin, 2) a state-space model with only a single temperature zone used by the aggregator, and 3) a state-space model of the 9 temperature zones used by the distributor.

A. Building Description

The building used in this study is a medium-sized company building located in Offenbach, Germany. It has a footprint of approx. 13,000 m² and can be separated into 9 different temperature zones, which include offices, halls, some workshops and, as a peculiarity, an emissions lab. Besides the connection to the public power grid, the main energy sources are a gas-fired combined heat and power plant (CHP) for co-production of electricity and heat with 199 kW_{el}, a fairly large photovoltaic (PV) plant with 750 kW_p, which serve an average load demand of approx. 250 kW. It further has gas-fired heating boilers and an electric heating, ventilation, and air conditioning (HVAC) system. A stationary second-life battery with a capacity of 98 kWh can be used as electric storage.

B. Digital Twin

A Modelica-based simulation model implemented in SimulationX is used as a digital twin [14]. It covers the 9 different

temperature zones, their couplings, heat losses to both the ambient air and the ground, internal heat gains from electrical consumption and occupants, and the above mentioned energy producers and consumers, including various constraints on the power production. The CHP has a minimal power output of 50 %, below which it cannot be modulated. Furthermore, its power-up and power-down times, as well as nonlinear efficiencies are considered. The SimulationX model uses historic measurement data for the ambient air temperature, the electric power demand (per zone), solar irradiation, and PV power production.

C. Aggregator Model

As discussed in the introduction, the physics-based digital twin is not suited to be used in an OCP. Thus, we use a simplified state space model representing the most important entities. Note that the hierarchization, i.e. the use of an aggregator and a distributor, is done to ensure the scalability of the control approach. This also allows the integration of additional components, e.g. charging stations for electric vehicles [15]. Of the total 9 temperature zones, 7 are aggregated as a single 'building zone' with an average temperature ϑ_b (in °C). The remaining 2 zones refer to server rooms and are aggregated with an average 'server zone' temperature ϑ_s (in °C). The stationary battery's stored energy E (in kWh) completes the state vector x_{agg} . The inputs u_{agg} to the system consist of the grid power P_{grid} , the (electrical) CHP power P_{chp} , the gas heating power \dot{Q}_{rad} , and the HVAC cooling power \dot{Q}_{cool} .

As disturbances d_{agg} , PV power P_{PV} , the building's electrical power demand P_{dem} , the ambient air temperature ϑ_{air} (in °C), (constant) losses to the ground $\dot{Q}_{\text{other,b}}$, and (constant) internal heatings $\dot{Q}_{\text{other,s}}$ are considered. All powers are given in kW. The time-continuous state space model is then given by

$$\begin{aligned} \begin{bmatrix} \dot{E}(t) \\ \dot{\vartheta}_b(t) \\ \dot{\vartheta}_s(t) \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{H_{\text{air,b}} - \beta_{\text{bs}}}{C_{\text{th,b}}} & \frac{\beta_{\text{bs}}}{C_{\text{th,b}}} \\ 0 & \frac{\beta_{\text{bs}}}{C_{\text{th,s}}} & -\frac{H_{\text{air,s}} - \beta_{\text{bs}}}{C_{\text{th,s}}} \end{bmatrix} \cdot \begin{bmatrix} E(t) \\ \vartheta_b(t) \\ \vartheta_s(t) \end{bmatrix} \dots \\ &+ \begin{bmatrix} 1 & 1 & 0 & \frac{1}{c_{\text{cur}} \cdot C_{\text{th,b}}} & \frac{1}{\varepsilon_c} \\ 0 & \frac{1}{C_{\text{th,b}}} & \frac{1}{C_{\text{th,b}}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{C_{\text{th,s}}} \end{bmatrix} \cdot \begin{bmatrix} P_{\text{grid}}(t) \\ P_{\text{chp}}(t) \\ \dot{Q}_{\text{rad}}(t) \\ \dot{Q}_{\text{cool,b}}(t) \\ \dot{Q}_{\text{cool,s}}(t) \end{bmatrix} \dots \\ &+ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{H_{\text{air,b}}}{C_{\text{th,b}}} & \frac{1}{C_{\text{th,b}}} & 0 \\ 0 & 0 & \frac{H_{\text{air,s}}}{C_{\text{th,s}}} & 0 & \frac{1}{C_{\text{th,s}}} \end{bmatrix} \cdot \begin{bmatrix} P_{\text{ren}}(t) \\ P_{\text{dem}}(t) \\ \vartheta_{\text{air}}(t) \\ \dot{Q}_{\text{other,b}}(t) \\ \dot{Q}_{\text{other,s}}(t) \end{bmatrix}, \quad (1) \end{aligned}$$

where $C_{\text{th,b}}$ and $C_{\text{th,s}}$ are the thermal capacities of the building and server zone, respectively in $\frac{\text{kWh}}{\text{K}}$, $H_{\text{air,b}}$ and $H_{\text{air,s}}$ are the heat transfer coefficients to the ambient air of the building and the server zone in $\frac{\text{kW}}{\text{K}}$, respectively, β_{bs} is the heat transfer coefficient between the two zones in $\frac{\text{kW}}{\text{K}}$ and c_{cur} is the ratio of the CHP's electrical to thermal power. Numerical values are given in Table I.

In the following, we only use its discretized state space form. Furthermore, we respect the model errors for the building and the server zone temperatures, i.e.

$$x_{\text{agg}}(k+1) = A_{\text{agg}}(T_s)x_{\text{agg}}(k) + B_{\text{agg}}(T_s)u_{\text{agg}}(k) \dots + S_{\text{agg}}(T_s)d_{\text{agg}}(k) + \epsilon_{\text{agg}}(k) \quad (2)$$

with $\epsilon_{\text{agg}}(k) = [0 \quad \epsilon_b(k) \quad \epsilon_s(k)]^T$ and T_s being the sampling rate in h. Note that we can respect the model error $\epsilon_{\text{agg}}(k)$ only in the discretized form since it has to be estimated from discretely sampled data points. For more details on the modeling itself, the reader is referred to [16].

D. Distributor Model

The distributor models the 9 temperature zones individually, while neglecting the electrical part of the aggregator model. The temperature ϑ_i of a single zone i can be described by

$$\begin{aligned} \dot{\vartheta}_i(t) = & \frac{1}{C_{\text{th},i}} \left(\dot{Q}_{\text{heat},i}(t) + \dot{Q}_{\text{cool},i}(t) + \dot{Q}_{\text{other},i}(t) \right) \dots \\ & - \sum_{j \neq i} \frac{\beta_{ij}}{C_{\text{th},i}} (\vartheta_i(t) - \vartheta_j(t)) \dots \\ & - \frac{H_{\text{air},i}}{C_{\text{th},i}} (\vartheta_i(t) - \vartheta_{\text{air}}(t)) \end{aligned} \quad (3)$$

with $C_{\text{th},i}$ being the thermal capacity of zone i in $\frac{\text{kWh}}{\text{K}}$, β_{ij} the heat transfer coefficient between zones i and j in $\frac{\text{kW}}{\text{K}}$, $H_{\text{air},i}$ the heat transfer coefficient between zone i and the outside air in $\frac{\text{kW}}{\text{K}}$, and $\dot{Q}_{\text{heat},i}$ and $\dot{Q}_{\text{cool},i}$ in kW the heating and cooling powers allocated to zone i . $\dot{Q}_{\text{other},i}$ is an uncontrollable disturbance, which is assumed constant and either represents heat losses to the ground (for the building zones 1-7) or internal heat gains (for the server zones 8 and 9).

Using the 9 ϑ_i as states x_{dis} , $\dot{Q}_{\text{heat},i}$ and $\dot{Q}_{\text{cool},i}$ as inputs u_{dis} , $\dot{Q}_{\text{other},i}$ as disturbances d_{dis} , and again discrete model errors $\epsilon_{\text{dis}}(k) = [\epsilon_1(k) \quad \dots \quad \epsilon_9(k)]^T$, they are expressed as the discrete state space model

$$x_{\text{dis}}(k+1) = A_{\text{dis}}(T_s)x_{\text{dis}}(k) + B_{\text{dis}}(T_s)u_{\text{dis}}(k) \dots + S_{\text{dis}}(T_s)d_{\text{dis}}(k) + \epsilon_{\text{dis}}(k), \quad (4)$$

where A_{dis} is the system matrix, B_{dis} the input matrix and S_{dis} the disturbance matrix. Again, T_s denotes the sampling time and the numerical values are given in Table I. For brevity, the reader is referred to [16] for more details on the state space model.

III. CONTROL APPROACH

In this section, we describe the OCPs solved by the MPC on both the aggregator and distributor level.

A. Aggregator Control

The aggregator's goal is to regulate the building temperatures while minimizing the monetary costs. This is expressed as a

Table I: Numerical values of the building parameters. Note that $C_{\text{th},b} = \sum_{i=1}^7 C_{\text{th},i}$, $C_{\text{th},s} = C_{\text{th},8} + C_{\text{th},9}$, $H_{\text{air},b} = \sum_{i=1}^7 H_{\text{air},i}$, $H_{\text{air},s} = H_{\text{air},8} + H_{\text{air},9}$, $\beta_{bs} = \beta_{29} + \beta_{58} + \beta_{68}$, $\beta_{ij} = \beta_{ji}$, and all other β_{ij} not listed below are zero, e.g. $\beta_{12} = 0$.

	in kWh/K		in kW/K		in kW/K
$C_{\text{th},1}$	230.88	$H_{\text{air},1}$	3.69	β_{29}	48.40
$C_{\text{th},2}$	476.29	$H_{\text{air},2}$	9.82	β_{34}	345.60
$C_{\text{th},3}$	214.27	$H_{\text{air},3}$	3.65	β_{56}	1100.48
$C_{\text{th},4}$	103.68	$H_{\text{air},4}$	2.79	β_{58}	23.40
$C_{\text{th},5}$	330.14	$H_{\text{air},5}$	4.79	β_{68}	8.00
$C_{\text{th},6}$	330.14	$H_{\text{air},6}$	6.19		
$C_{\text{th},7}$	99.456	$H_{\text{air},7}$	3.19		
$C_{\text{th},8}$	2.40	$H_{\text{air},8}$	0.03		
$C_{\text{th},9}$	4.80	$H_{\text{air},9}$	0.04		

weighted sum of multiple cost functions. First, for the building zone temperature, the so-called comfort costs

$$J_{\text{comf,agg}}(k) = \sum_{n=1}^{N_p} (\vartheta_b(n|k) - 22^\circ\text{C})^2, \quad (5)$$

apply. The notation $\vartheta_b(n|k)$ refers to the value for $\vartheta_b(k+n)$ predicted at time step k . N_{pred} is the number of steps in the prediction horizon. Second, the monetary costs are expressed as

$$J_{\text{mon}}(k) = \sum_{n=0}^{N_p-1} \ell_{\text{mon}}(P_{\text{grid}}(n|k), P_{\text{chp}}(n|k), \dot{Q}_{\text{heat}}(n|k), T_s) \quad (6)$$

where ℓ_{mon} describes the costs arising from gas usage and buying (selling) electrical energy from (to) the public grid. Note that we consider German industry pricing, in which different prices for buying and selling as well as high peak costs apply. Details on both numerical values and how J_{mon} can be reformulated using an epigraph formulation, which results in a linear programming problem, can be found in [16, pp. 24]. Third, the server zone is only kept within an acceptable temperature range by

$$J_{s,\text{agg}}(k) = \sum_{n=1}^{N_p} \max(15^\circ\text{C} - \vartheta_s(n|k), 0) \dots + \max(\vartheta_s(n|k) - 21^\circ\text{C}, 0). \quad (7)$$

The input constraints are given by

$$-1000 \text{ kW} \leq P_{\text{grid}}(k) \leq 1000 \text{ kW} \quad (8a)$$

$$0 \leq P_{\text{chp}}(k) \leq 199 \text{ kW}, \quad (8b)$$

$$0 \leq \dot{Q}_{\text{rad}}(k) \leq 1500 \text{ kW}, \quad (8c)$$

$$-1353 \text{ kW} \leq \dot{Q}_{\text{cool},b}(k) \leq 0, \quad (8d)$$

$$-197 \text{ kW} \leq \dot{Q}_{\text{cool},s}(k) \leq 0. \quad (8e)$$

The state constraints are given by

$$0.15 \cdot 98 \text{ kWh} \leq E(k) \leq 0.85 \cdot 98 \text{ kWh}, \quad (9a)$$

$$-32.9 \text{ kW} \leq \frac{E(k+1) - E(k)}{T_s} \leq 32.9 \text{ kW}. \quad (9b)$$

Note that ϑ_b and ϑ_s are unconstrained to avoid infeasibilities in the later co-simulation without error compensation. Both are only regulated due to the respective cost functions.

Together, the aggregator's OCP is described by

$$\min_{\mathbf{u}_{\text{agg}}} w_{\text{comf}} \cdot J_{\text{comf,agg}}(k) + w_{\text{mon}} \cdot J_{\text{mon}}(k) \dots + w_{\text{s,agg}} \cdot J_{\text{s,agg}}(k), \quad (10a)$$

$$\text{s. t. (2), (8) } \forall n = 0 \dots N_{\text{pred}} - 1, \quad (10b)$$

$$(9) \forall n = 1 \dots N_{\text{pred}}, \quad (10c)$$

with $\mathbf{u}_{\text{agg}} = (u_{\text{agg}}(0|k), \dots, u_{\text{agg}}(N_{\text{pred}} - 1|k))$ being the sequence of control inputs, and a prediction horizon of $N_{\text{pred}} = 48$ steps of $T_s = 0.5$ h each, i.e. 1 day in total. The time step notation (k) and $(k+1)$ in (2), (8) and (9) are to be read as $(n|k)$ and $(n+1|k)$, respectively.

Usually, the weights are chosen such that a reasonable compromise is determined [17]. Alternatively, multi-objective optimization can be used [18], [19], since the aggregator's OCP is always solvable quickly enough due to the hierarchization. However, here we choose $w_{\text{comf}} = 0.99$, $w_{\text{mon}} = 0.01$, $w_{\text{s,agg}} = 0.99$ to ensure that the controller tries to achieve $\vartheta_b = 22^\circ\text{C}$ and $15^\circ\text{C} \leq \vartheta_s \leq 21^\circ\text{C}$ at all times. This simplifies the evaluation of the error compensation later on. Note that in the actual implementation, additional slack variables are used due to the reformulation of J_{mon} and of the max-terms of $J_{\text{s,agg}}$.

Since we want to assess the compensation of the model error, we simulate with no prediction error. Namely, we assume perfect predictions for the PV power, the building's load and the ambient air temperature. For an assessment of the influence of real predictions for the facility under study, the reader is referred to [20] and [21].

B. Distributor Control

In the distributor, the total heating and cooling powers determined by the aggregator are split (distributed) between the individual zones. To this end, we use individual weights $\sum_{i=1}^9 w_{\text{th},i} = 1$ proportional to the thermal capacities, i.e.

$$w_{\text{th},i} = \frac{C_{\text{th},i}}{\sum_{j=1}^9 C_{\text{th},j}} \quad \forall i = 1, \dots, 9. \quad (11)$$

The temperature goals are the same as in the aggregator, i.e. we punish temperature deviations from 22°C in the 7 building zones by

$$J_{\text{comf,dis}}(k) = \sum_{n=1}^{N_p} \sum_{i=1}^7 w_{\text{th},i} \cdot (\vartheta_i(n|k) - 22^\circ\text{C})^2. \quad (12)$$

For the 2 server zones, the same temperature range applies as in the aggregator. Outside of these, we punish temperature deviations by

$$J_{\text{s,dis}}(k) = \sum_{n=1}^{N_p} \sum_{i=8}^9 \left(\max(15^\circ\text{C} - \vartheta_i(n|k), 0) \dots + \max(\vartheta_i(n|k) - 21^\circ\text{C}, 0) \right). \quad (13)$$

The inputs are subject to box constraints which stem from the building's internal infrastructure,

$$0 \leq \dot{Q}_{\text{heat},i}(k) \quad \forall i = 1, \dots, 7 \leq 893.95 \text{ kW}, \quad (14a)$$

$$0 \leq \dot{Q}_{\text{heat},i}(k) \quad \forall i = 8, \dots, 9 \leq 0, \quad (14b)$$

$$-800 \text{ kW} \leq \sum_{i \in \{1,2,3,4,7\}} \dot{Q}_{\text{cool},i}(k) \leq 0, \quad (14c)$$

$$-330 \text{ kW} \leq \dot{Q}_{\text{cool},5}(k) + \dot{Q}_{\text{cool},6}(k) \leq 0, \quad (14d)$$

$$-53 \text{ kW} \leq \dot{Q}_{\text{cool},8}(k) \leq 0, \quad (14e)$$

$$-144 \text{ kW} \leq \dot{Q}_{\text{cool},9}(k) \leq 0. \quad (14f)$$

Note that the server zones 8 and 9 have no heating systems, since they have to be cooled all the time. Furthermore, the total powers are constrained by the powers allocated by the aggregator,

$$\sum_{i=1}^7 \dot{Q}_{\text{heat},i}(k) = \dot{Q}_{\text{heat}}(k) + \frac{P_{\text{chp}}(k)}{c_{\text{cur}}}, \quad (15a)$$

$$\sum_{i=1}^7 \dot{Q}_{\text{cool},i}(k) = \dot{Q}_{\text{cool,b}}(k), \quad (15b)$$

$$\sum_{i=8}^9 \dot{Q}_{\text{cool},i}(k) = \dot{Q}_{\text{cool,s}}(k). \quad (15c)$$

As in the aggregator, the zone temperatures have no hard constraints to avoid infeasibilities in the co-simulation with no error compensation.

Together, the distributor's OCP is described by

$$\min_{\mathbf{u}_{\text{dis}}} J_{\text{comf,dis}}(k) + J_{\text{s,dis}}(k) \quad (16a)$$

$$\text{s. t. (4), (14), (15) } \forall n = 0 \dots N_{\text{pred}} - 1 \quad (16b)$$

with $\mathbf{u}_{\text{dis}} = (u_{\text{dis}}(0|k), \dots, u_{\text{dis}}(N_{\text{pred}} - 1|k))$ being the sequence of control inputs, and the same prediction horizon as in the aggregator. Again, the time step notation (k) and $(k+1)$ in (4), (14) and (15) are to be read as $(n|k)$ and $(n+1|k)$, respectively.

IV. ERROR COMPENSATION METHODOLOGY

As previously described, the control approach is aware of a model error $\epsilon(k)$ in both the aggregator and distributor. We aim to perform error compensation, i.e. we want to find an estimator $\tilde{\epsilon}(k)$ that can approximate this error, such that $\tilde{\epsilon}(k) \approx \epsilon(k)$. Incorporating the estimator to approximate the model error should improve control performance. We use machine learning regression models to build these estimators. We train 9 estimators $\tilde{\epsilon}_i(k)$, $i = 1, \dots, 9$, i.e. one for each temperature zone in the distributor. The estimators in the aggregator are the weighted sum of the individual zone estimators, i.e.

$$\tilde{\epsilon}_b(k) = \frac{\sum_{i=1}^7 w_{\text{th},i} \cdot \tilde{\epsilon}_i(k)}{\sum_{i=1}^7 w_{\text{th},i}}, \quad (17a)$$

$$\tilde{\epsilon}_s(k) = \frac{\sum_{i=8}^9 w_{\text{th},i} \cdot \tilde{\epsilon}_i(k)}{\sum_{i=8}^9 w_{\text{th},i}}. \quad (17b)$$

This is analogous to ϑ_b and ϑ_s being the weighted averages of the individual zone temperatures ϑ_i . The estimators will predict the model error for only one time step at a time, i.e. N_p separate predictions will be made to calculate $\tilde{\epsilon}_i(n|k)$ over the horizon $n = 0, \dots, N_p - 1$ at each time step k .

A. Feature selection

The first step to training a regression model is feature selection. The target variable (i.e. labels) of the regression model are the measured model errors. In principle, these can be calculated as the difference between observed state and predicted state, i.e. $\epsilon(k) = x(k+1) - x(1|k)$. Generally, the resulting difference may also include prediction errors of the disturbances. This can be circumvented by recalculating $x(1|k)$ using the state space model and measurements of disturbances $d(k)$ and inputs $u(k)$. For selecting the features, we want to consider that the resulting estimators should be easily (re)trainable and employable. This means that we should only use features that are readily available and both measurable and predictable. Therefore, all disturbances of the MPC controller are good candidate features, as they are both measurable and predictable in the case of the proposed EMS. From a brief correlation analysis between measured errors $\epsilon(k)$ and measured disturbances (omitted for brevity), we deduced that a set of 5 features should provide good a basis for training, i.e.

- 1) the current ambient temperature $\vartheta_{\text{air}}(k)$, for unaccounted heat flows to/from the environment (e.g. inaccurate heat transfer coefficients; warm/cold air from ventilation)
- 2) the past values of the ambient temperature $\vartheta_{\text{air}}(k-1), \dots, \vartheta_{\text{air}}(k-n_{\text{hist}})$, for heat diffusion from other zones
- 3) the total building load $P_{\text{dem}}(k)$, as electrical consumption is transformed into heat and is correlated to occupant behavior
- 4) the time of day in h, i.e. $\text{ToD}(k) : \mathbb{N} \rightarrow [0, 24)$, for regular occupant behavior and ventilation schedule
- 5) the day of the year $\text{DoY}(k) : \mathbb{N} \rightarrow \{1, \dots, 365\}$, for mapping seasonal effects.

B. Models

Based on these features, we propose two candidate regression models as estimators: 1) A linear regression model and 2) an XGBoost regression model. XGBoost (eXtreme Gradient Boosting) is an open-source software library that provides an efficient and effective implementation of the gradient boosting framework for machine learning [22]. It uses gradient boosting [23] to improve the performance of decision trees, which can be used for both regression and classification problems.

For the first estimator, we propose the linear model

$$\begin{aligned} \tilde{\epsilon}_i^{\text{lin}}(k) = & + \gamma_{i,1} \sin\left(\frac{2\pi}{24}\text{ToD}(k)\right) + \gamma_{i,2} \cos\left(\frac{2\pi}{24}\text{ToD}(k)\right) \\ & + \delta_{i,1} \sin\left(\frac{2\pi}{365}\text{DoY}(k)\right) + \delta_{i,2} \cos\left(\frac{2\pi}{365}\text{DoY}(k)\right) \\ & + \alpha_i P_{\text{dem}}(k) + \sum_{j=0}^{n_{\text{hist}}} \beta_{i,j} \vartheta_{\text{air}}(k-j) + \kappa_i \end{aligned} \quad (18)$$

for each zone i , where $n_{\text{hist}} = 2$. The parameters $\alpha_i, \beta_{i,j}, \gamma_{i,l}, \delta_{i,l}, \kappa_i$ are fitted through least-squares regression. The features $\text{ToD}(k)$ and $\text{DoY}(k)$ are transformed using a cyclical transformation to normalize them uniquely to values between -1 and 1, preserving the cyclical nature of day time and seasons.

For the second estimator, $\tilde{\epsilon}_i^{\text{xgb}}(k)$, we train an XGBoost regressor for each zone i . For this estimator, we use all aforementioned features and $n_{\text{hist}} = 2$. Contrary to the linear model, we do not apply a cyclical transformation to the time features, as this is not needed with tree-based regression models and can actually be detrimental.

C. Training and evaluation

To generate the training data for training the estimators, we use the digital twin model described in Section II-B. We simulate a full calendar year using the digital twin in a SiL setup together with the described control approach and without compensation, i.e. $\tilde{\epsilon}(k) = 0$. In this setup, the digital twin running in SimulationX is connected through an FMU (functional mock-up unit) to a Python bridge, linking it to the MPC controller implemented in MATLAB using the PARODIS framework [24]. At each discrete time step k , the controller receives the updated system states from the digital twin model, determines the control input $u(k)$ and applies it to the digital twin. For the simulation in the digital twin and the predictions for the MPC controller, we use measurement data for the weather and electrical demands collected for the year 2021 at the Honda R&D facility in Offenbach, Germany. During the simulation, we collect both the states predicted by the MPC as well as the realized (i.e. measured) states. From these we calculate the model error $\epsilon(k)$ for training. In a real world setting, one would use a baseline controller to run in parallel to a controller with an estimator pre-trained in a digital twin setting, to be able to calculate raw model errors to retrain estimators on new data.

One of the main error sources between digital twin models of buildings and reality, next to occupant behavior, are the estimated heat capacities of the building zones [25]. Therefore, we benchmark the robustness of our proposed error compensation against this error, by creating additional simulation scenarios, where we change the heat capacities in the model of the MPC, while keeping them the same in the digital twin model. Overall, we examine four scenarios, namely

- 1) heat capacities in MPC model are exact,
- 2) heat capacities in MPC are 50 % of digital twin,
- 3) heat capacities in MPC are 150 % of digital twin,
- 4) total heat capacity of the building is exact, individual capacities are shifted randomly according to Algorithm 1.

To train the estimators, we use a 70/30 train-test split on the collected data, and use scikit-learn [26] to fit the linear regressor as well as the scikit-learn interface of the XGBoost Python library for training the XGBoost estimator, respectively.

Algorithm 1 Shifting of capacities between adjacent zones

for all zone coupling pairs (i, j) **do**
 Draw capacity shift fraction $p_{\text{shift}} \leftarrow \mathcal{U}[0, 0.5]$
 Randomly decide shift direction $d_{\text{shift}} \leftarrow \{-1, 1\}$
 Calculate $C_{\text{shift}} = \min(C_i, C_j) \cdot p_{\text{shift}} \cdot d_{\text{shift}}$
 Update $C_i = C_i + C_{\text{shift}}$
 Update $C_j = C_j - C_{\text{shift}}$
end for

Table II shows the performance of the trained estimators on the training and test data sets in terms of mean absolute error (MAE) of the residual model error $\tilde{\epsilon}_i(k) - \epsilon_i(k)$. We calculate the overall MAE as the weighted sum of the MAE of each temperature zone over the data set, i. e.

$$\text{MAE} = \sum_{i=1}^9 w_i \left(\frac{1}{N} \sum_{k=0}^{N-1} |\tilde{\epsilon}_i(k) - \epsilon_i(k)| \right). \quad (19)$$

The linear estimator shows fair performance on both data sets. The XGBoost estimator shows very good performance on the training set and similar performance on the test set. This suggests that the estimator is not overfitting.

Table II: Performance of trained estimator models under different scenarios in terms of mean absolute error (MAE). The baseline performance column refers to the measured model error in the baseline simulations without error compensation.

Estimator	Scenario	MAE in 10^{-3} K			
		Baseline	Train	Test	SiL
Linear	2021	38.766	15.853	15.836	16.904
	2022	42.137	—	—	20.461
XGBoost	2021	38.766	4.846	5.379	7.440
	2022	42.137	—	—	15.571
	2021 50%	76.596	7.893	8.718	12.820
	2021 150%	27.544	5.616	6.291	9.532
	2021 shifted	39.473	6.073	6.653	8.521

V. SIMULATION RESULTS

To evaluate the performance of the proposed error compensation approach, we applied the trained estimators in the previously described SiL simulation. First, we simulated the baseline 2021 simulation with active compensation. Figure 1 shows the resulting average temperature of the building zone for the baseline case compared to linear compensation and XGBoost compensation. This shows that the control performance regarding the comfort costs in the aggregator (i. e. deviation from the setpoint of 22°C) is significantly improved with active error compensation.

Figure 2 shows the residual errors, i. e. left over model error, for zones 1 and 9 in the baseline case compared to the active compensation using the XGBoost estimator. This suggests that the estimator manages to approximate the actual model error also in the SiL simulation. This is confirmed by looking at the overall MAE of the residual errors, as shown in Table II. The MAE of both the linear estimator and the XGBoost

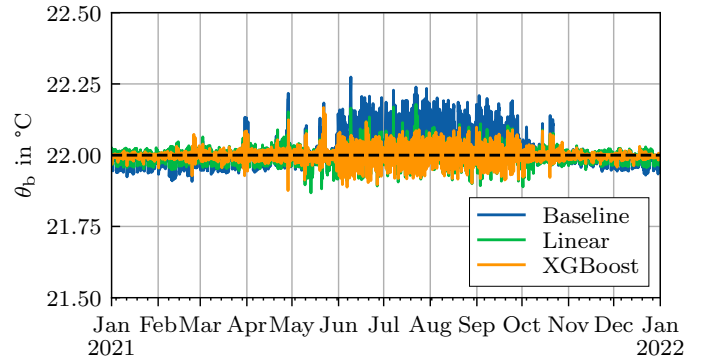


Figure 1: Comparison of overall building temperature ϑ_b for the year 2021 between no error compensation (Baseline), the linear estimator model (Linear) and the XGBoost estimator model (XGBoost).

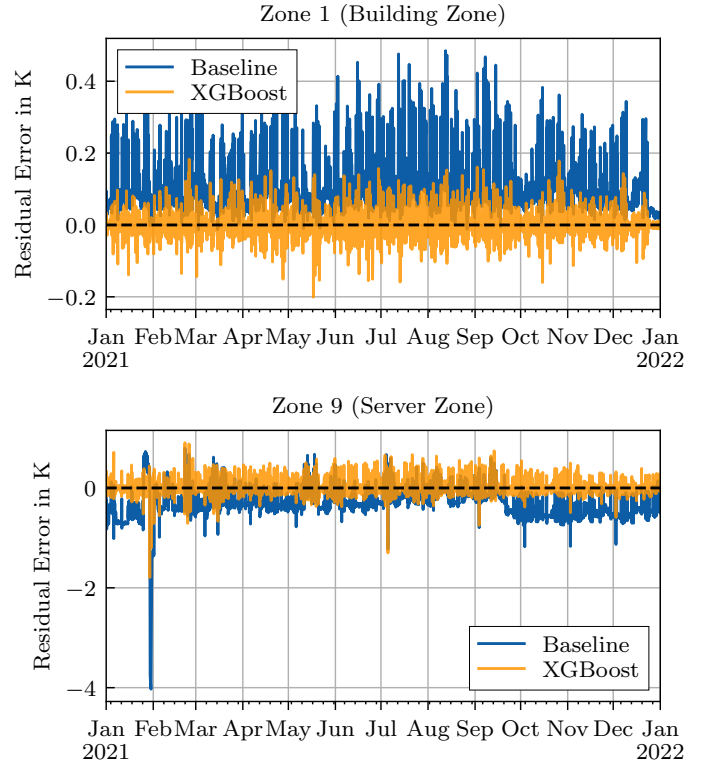


Figure 2: Comparison of the residual model error in zones 1 and 9 without compensation (Baseline) and with the XGBoost estimator model for the year 2021.

estimator are significantly lower than in the baseline case, with XGBoost clearly outperforming the linear model. The linear estimator reduces the MAE by 56 %, the XGBoost estimator by 80 %. Both the linear and XGBoost estimator have a slightly decreased performance in the SiL setting.

The performance of the estimators is further illustrated in Figure 3, where the MAE of each of the 9 temperature zones is shown and compared between the three cases. This again shows that both estimators manage to approximate the model error well in all zones with XGBoost exhibiting best performance.

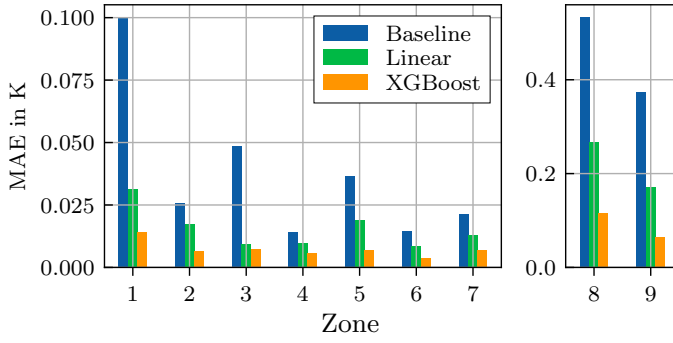


Figure 3: Mean absolute error (MAE) of the residual model error in all temperature zones with no error compensation (Baseline), with the linear estimator model (Linear), and with the XGBoost estimator model (XGBoost).

To evaluate the generalizability of the proposed approach, we tested the performance of the estimators trained on 2021 data in a SiL simulation with unseen data of the year 2022. This way, we can implicitly test robustness against changing occupant behavior and weather. We simulated the year 2022 analogously to the 2021 base scenario, i.e. with exact capacities in the controller, both without compensation and with each of the estimators. The results are again shown in Table II. Both estimators exhibit decreased performance on 2022 data. The linear estimator reduces the MAE by 51 %, the XGBoost estimator by 63 %. The results suggest that periodical retraining of these data-driven estimators may be necessary. For example, the facility under study experienced significant change in occupant behavior between 2021 and 2022 due to COVID-19 policies.

As motivated in Section IV-C, we want to test the robustness of the approach against errors in estimated heat capacities in the controller. We therefore simulated the three described scenarios, i.e. 1) heat capacities at 50 %, 2) heat capacities at 150 %, and 3) randomly shifted heat capacities in the MPC, with error compensation. We only simulated with the better performing XGBoost estimator. The results are again shown in Table II. In all cases, the estimators manage to significantly reduce the residual model error, while yielding only slightly worse performance than with exact heat capacities. Overall, the results suggest that the approach is reasonably robust against this type of error.

VI. CONCLUSION & OUTLOOK

We have shown that our proposed data-driven error compensation approach can significantly reduce the residual model error between the proposed hierarchical MPC controller and a digital twin building model in a SiL simulation. We have proposed two simple regression-based error estimator models, which achieve an error reduction of up to 56 % (linear model) and 80 % (XGBoost) in a baseline full calendar year simulation. We have shown that the proposed approach is robust against model errors of heat capacities in the controller. Furthermore, we have shown that the regression-based estimators generalize reasonably well by applying the estimators trained on 2021 measurement data to a simulation based on 2022 measurement data. Despite significant change in occupant behavior between 2021 and 2022, both estimators exhibit good performance.

While the proposed error compensation approach achieves significant model error reduction in all zones and improved control performance of the overall building temperature, as shown in Figure 1, the control performance in individual zones is still lacking. This is illustrated in Figure 4, where the temperature of zone 1 over the course of the year 2021 is shown, with and without error compensation. The control performance is only marginally better in the case with active error compensation. This is due to the structure of the hierarchical control approach: The aggregator derives a heating and cooling budget by considering the weighted sum of the individual zone errors of the distributor. Thereby, positive and negative components cancel out. In turn, not enough heating and cooling budget is allocated for compensation in the distributor. This problem could be resolved in future work by extending the control scheme by introducing additional communication between the two layers.

For this paper, we have used data from a full calendar year for training. However, further investigation could be conducted to understand how the amount of training data relates to the performance of the error compensation, to determine how much data is needed to (re)train compensators in a real life setting.

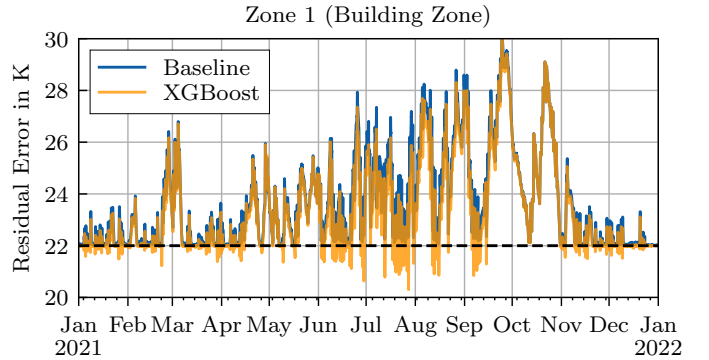


Figure 4: Temperature in zone 1 in the year 2021 without error compensation (Baseline) and with compensation using the XGBoost estimator (XGBoost).

REFERENCES

- [1] S. Prívvara, J. Cigler, Z. Váňa, F. Oldewurtel, C. Sagerschnig, and E. Žáčeková, “Building modeling as a crucial part for building predictive control,” *Energy and Buildings*, vol. 56, pp. 8–22, 2013. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378778812005336>
- [2] Z. Wang and Y. Chen, “Data-driven modeling of building thermal dynamics: Methodology and state of the art,” *Energy and Buildings*, vol. 203, p. 109405, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378778819303913>
- [3] E. T. Maddalena, Y. Lian, and C. N. Jones, “Data-driven methods for building control — a review and promising future directions,” *Control Engineering Practice*, vol. 95, p. 104211, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0967066119301832>
- [4] P. Stoffel, C. Löffler, S. Eser, A. Kumpel, and D. Müller, “Combining data-driven and physics-based process models for hybrid model predictive control of building energy systems,” in *2022 30th Mediterranean Conference on Control and Automation (MED)*, 2022, pp. 121–126.
- [5] J. Tang, C. Fan, F. Zeng, and W. Feng, “Data-driven model predictive control for power demand management and fast demand response of commercial buildings using support vector regression,” in *Building Simulation*, vol. 15, no. 3. Springer, 2022, pp. 317–331.
- [6] A. Jain, F. Smarra, E. Reticcioli, A. D’Innocenzo, and M. Morari, “Neuropt: Neural network based optimization for building energy management and climate control,” in *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, ser. Proceedings of Machine Learning Research, A. M. Bayen, A. Jadbabaie, G. Pappas, P. A. Parrilo, B. Recht, C. Tomlin, and M. Zeilinger, Eds., vol. 120. PMLR, 10–11 Jun 2020, pp. 445–454. [Online]. Available: <https://proceedings.mlr.press/v120/jain20a.html>
- [7] H. Huang, L. Chen, and E. Hu, “A hybrid model predictive control scheme for energy and cost savings in commercial buildings: Simulation and experiment,” in *2015 American Control Conference (ACC)*, 2015, pp. 256–261.
- [8] P. Ferreira, A. Ruano, S. Silva, and E. Conceição, “Neural networks based predictive control for thermal comfort and energy savings in public buildings,” *Energy and Buildings*, vol. 55, pp. 238–251, 2012, cool Roofs, Cool Pavements, Cool Cities, and Cool World. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S037877881200388X>
- [9] A. Kathirgamanathan, M. De Rosa, E. Mangina, and D. P. Finn, “Data-driven predictive control for unlocking building energy flexibility: A review,” *Renewable and Sustainable Energy Reviews*, vol. 135, p. 110120, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1364032120304111>
- [10] F. Massa Gray and M. Schmidt, “A hybrid approach to thermal building modelling using a combination of gaussian processes and grey-box models,” *Energy and Buildings*, vol. 165, pp. 56–63, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378778817332942>
- [11] A. Jain, M. O’Kelly, P. Chaudhari, and M. Morari, “Bayesrace: Learning to race autonomously using prior experience,” *arXiv preprint arXiv:2005.04755*, 2020.
- [12] J. Yoo, A. Molin, M. Jafarian, H. Esen, D. V. Dimarogonas, and K. H. Johansson, “Event-triggered model predictive control with machine learning for compensation of model uncertainties,” in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Dec. 2017, pp. 5463–5468.
- [13] A. Carron, E. Arcari, M. Wermelinger, L. Hewing, M. Hutter, and M. N. Zeilinger, “Data-driven model predictive control for trajectory tracking with a robotic arm,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3758–3765, Oct. 2019.
- [14] R. Unger, B. Mikoleit, T. Schwan, B. Bäker, C. Kehrler, and T. Rodemann, “Green building - modeling renewable building energy systems with eMobility using Modelica,” in *Proceedings of Modelica 2012 conference*. Munich, Germany: Modelica Association, September 2012.
- [15] J. Engel, T. Schmitt, T. Rodemann, and J. Adamy, “Hierarchical economic model predictive control approach for a building energy management system with scenario-driven EV charging,” *IEEE Transactions on Smart Grid*, pp. 1–1, 2022.
- [16] T. Schmitt, “Multi-objective building energy management optimization with model predictive control,” Ph.D. dissertation, Technische Universität Darmstadt, Darmstadt, 2022. [Online]. Available: <http://tuprints.ulb.tu-darmstadt.de/22344/>
- [17] T. Schmitt, J. Engel, T. Rodemann, and J. Adamy, “Application of Pareto optimization in an economic model predictive controlled microgrid,” in *2020 28th Mediterranean Conference on Control and Automation (MED)*. IEEE, 2020, pp. 868–874. [Online]. Available: <https://www.honda-ri.de/pubs/pdf/4341.pdf>
- [18] T. Schmitt, T. Rodemann, and J. Adamy, “Multi-objective model predictive control for microgrids,” *at - Automatisierungstechnik*, vol. 68, no. 8, pp. 687–702, 2020. [Online]. Available: <https://www.honda-ri.de/pubs/pdf/4361.pdf>
- [19] T. Schmitt, M. Hoffmann, T. Rodemann, and J. Adamy, “Incorporating human preferences in decision making for dynamic multi-objective optimization in Model Predictive Control,” *Inventions*, vol. 7, no. 3, 2022. [Online]. Available: <https://www.mdpi.com/2411-5134/7/3/46>
- [20] C. Wang, T. Bäck, H. H. Hoos, M. Baratchi, S. Limmer, and M. Olhofer, “Automated machine learning for short-term electric load forecasting,” in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2019, pp. 314–321.
- [21] T. Schmitt, T. Rodemann, and J. Adamy, “The cost of photovoltaic forecasting errors in microgrid control with peak pricing,” *Energies*, vol. 14, no. 9, 2021. [Online]. Available: <https://www.mdpi.com/1996-1073/14/9/2569>
- [22] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, aug 2016. [Online]. Available: <https://doi.org/10.1145/2939672.2939785>
- [23] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” *The Annals of Statistics*, vol. 29, no. 5, pp. 1189 – 1232, 2001. [Online]. Available: <https://doi.org/10.1214/aos/1013203451>
- [24] T. Schmitt, J. Engel, M. Hoffmann, and T. Rodemann, “PARODIS: One MPC framework to control them all. Almost,” in *2021 IEEE Conference on Control Technology and Applications (CCTA)*, 2021.
- [25] W. Tian, Y. Heo, P. de Wilde, Z. Li, D. Yan, C. S. Park, X. Feng, and G. Augenbroe, “A review of uncertainty analysis in building energy assessment,” *Renewable and Sustainable Energy Reviews*, vol. 93, pp. 285–301, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S136403211830368X>
- [26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.