**13th IEA Heat Pump Conference**
April 26-29, 2021 Jeju, Korea

# Model Calibration of an Air Source Heat Pump System for Transient Simulations in Modelica

Philipp Mehrfeld[a,*], Markus Nürenberg[a], Dirk Müller[a]

RWTH Aachen University, E.ON Energy Research Center, Institute for Energy Efficient Buildings and Indoor Climate,
Mathieustrasse 10, 52074 Aachen, Germany

## Abstract

Within the scope of HVAC and building performance simulations, 74 % perform manual calibrations whilst the remaining 26 % aim for a fully automated process. Nonetheless, with focus on detailed grey-box simulation models for complex HVAC systems, such as a heat pumps (HP), a full automation might lead to several assumptions and, thus, to inaccuracies regarding the calibration results in terms of its misfit measures. Therefore, we define clear requirement specifications for an experimental design in combination with a semi-automated calibration procedure for an electrically driven air source HP system. The device under test is located in a climatic chamber and coupled to a hydraulic test bench, representing the source and sink of HP, respectively. As software toolchain, we use a combination of Python-based framework and Modelica-based simulation model library. We define the objective function as a chained and weighted KPI consisting of the normalized misfit measure CV(RMSE) of supply temperature and consumed electrical power. In a three-day long test with realistic dynamic boundary conditions, covering a comprehensive spectrum of inputs, we validate the calibrated models. The validation misfit measures show a deviation of 1.6 %.

*Keywords: Type your keywords here, separated by semicolons ;*

## 1. Introduction

Electrically driven heat pumps (HPs) are a main rising technology in the building energy sector. With a growing share of renewable energy sources in the current mixture of electricity, HPs are the most promising energy conversion system to convert electricity efficiently into thermal power. This work will focus mainly on air source heat pumps as they have a higher market share than ground source HP [1].

As the computing power has massively increased over time, HVAC plant simulation models represent an ever-increasing approach to predicting system performance using transient simulation. There are different approaches, ranging from white-box to grey-box to black-box modelling. Anyway, in every case adequate parameterization is necessary so that the simulation model is able to represent a real plant. In addition to parameterization with the aid of manufacturer data sheets, a semi-automatic calibration of a grey-box air-source HP (ASHP) model is used in this paper. Semi-automatic in this case means that besides a manual aggregation of existing model classes in the modelling language Modelica, an individual preprocessing of used input time series data takes place. The rest of the calibration process is fully automatic.

A well calibrated model of the HVAC system is necessary, for example, to:

- predict system performance for e.g. one year using a simulation with a suitable sink model (building + e.g. underfloor heating and dynamic user boundary conditions)
- derive a model for inclusion in an MPC algorithm [2, 3]

---

[*] Corresponding author. Tel.: +49 241 80 49776; fax: +49 241 80 49779.
*E-mail address:* pmehrfeld@eonerc.rwth-aachen.de.

- test functionalities of the system and the controller virtually (model in the loop) and thus save cost-intensive hardware tests

For this reason, measurement data is extracted from a 3-day dynamic test for an ASHP. About 60 % of the time series data is used for training and 40 % for validation purposes.

This paper deals with a developed framework consisting of a python-based and a modelica-based part. Figure 2 shows an overview of the process. The use case applied represents the calibration of an ASHP from existing time series data. In order to keep further influences to a minimum, the domestic hot water preparation and the control system are left out, although both were part of the experiment that provides the measurement data.

## 2. State of the art and initial condition

### *2.1. Calibration*

Calibration is the variation of model parameters so that the difference between simulated and measured data is as small as possible [4]. A challenge in the field of modelling is the divergence between simulated and practical behavior [5]. In order to make building energy simulations a reliable tool for analyses regarding energy saving potentials or operational optimizations, there is a great interest in a correspondence between simulated and measured values of technical systems [6].

Within this framework, a model calibration in its core describes the parameterization of a model so that generated simulation results fit the measurement results of real systems. Before this, sensitive model parameters can be identified by sensitivity or uncertainty analyses and afterwards applied to the calibration process. [6]. However, we skip this step in the context of this work, since we know the model sufficiently well to determine the tuner parameters ourselves.

The mathematical methods differentiated Coakley et al. [7] in optimization methods and alternative modeling techniques. The optimization procedures are based on an objective function that evaluates the difference between measured and simulated data in the form of a statistical indicator or misfit measure. The optimization problem is formulated as a minimization problem, since this difference has to be reduced. The practical implementation typically takes place via the coupling of a simulation software with an optimization algorithm.

### *2.2. Measurement Data*

To calibrate a model, it is vital to have extensive measurement data. In the scope of our case study, we obtained the dataset from a hardware-in-the-loop experiment [8, 9]. Figure 1 shows test bench arrangement of the ASHP system mentioned above and locations where the monitoring equipment are mounted. The HP is inverter controlled. While the outdoor unit of the split HP was positioned in a climatic chamber, the indoor unit was connected to a hydraulic test bench in order to emulate the outside air conditions and heat sink, respectively. With heat sink it is meant that simulation models of the hydronic network, the radiator heat transfer system and the building shell of a single family house were running during the experiment and values were interchanged online. Since a feedback of the simulation influences the hardware setup and vice versa, this explains the origin of the term hardware-in-the-loop. Tests with a monitoring frequency of 1 s represent the measurement database for the later performed calibration.

Uncertainties of measurements are evaluated for the supplied thermal energy and electrical energy consumption from a set of 11 reproducibility tests [10]. Results show a maximum uncertainty of 1.85 % for the thermal and 0.87 % for the electric energy.

Furthermore, we apply highly dynamic user profiles to the building model during the hardware-in-the-loop experiment. In terms of user profile we mean internal Gains up to 700 W, air change rates up to 0.8 1/h and room set temperatures up to 24 °C in the bathroom. The combination of all these boundary conditions and general heat losses of HVAC components like the buffer storage lead to a certain heat demand even during warmer days.

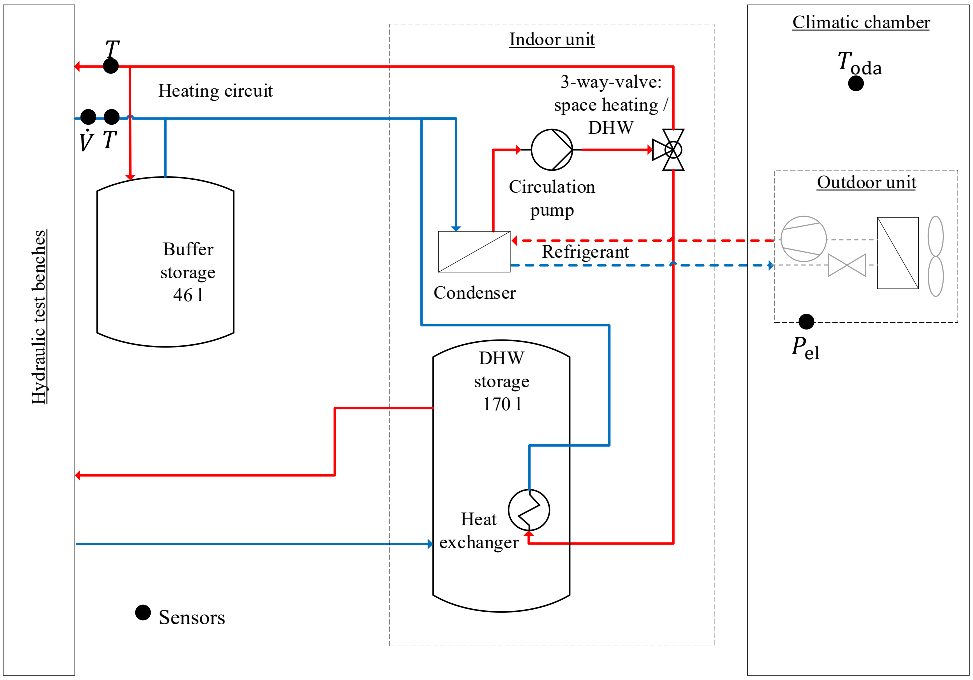

Figure 1: Experimental setup of hardware-in-the-loop experiment

### *2.3. Models*

Throughout this work, we use the open-source and object-oriented language Modelica to model and run building performance and HVAC simulations. Modelica is specially designed for modeling physical systems [11]. To perform the required simulations for this work we use the freely model library AixLib [12] [13], which is a part of the IBPSA Project 1 [14] (former IEA EBC Annex 60 [15], therefore some models will be found as well in the main library IBPSA [14]. The HP model depends on tabulated data as stated by EN 14511 [16]. The model uses both black-box and grey-box modeling approaches. The first is used for the refrigerant circuit, whilst the latter is used for the remaining model parts. This means that thermal capacities, heat losses and pressure drops of the heat exchangers at both sides of the heat pump, condenser (con) and evaporator (eva), are implemented as grey-box approach in the model. A variable compressor speed signal represents one mandatory and time variable input to the model. A PT1 element of the model characterizes the dynamic behavior of start-up and shut-off phases of the compressor. The heat losses at the condenser to the atmosphere is parametrized by the combination of a thermal resistance-capacity model. Furthermore, in order to be able to model the temperature change due to heat losses in connected piping – even when there is no circulating fluid – a time constant in the temperature sensor can be defined in the corresponding Modelica model.

Furthermore, the simulation setup, consists of the HP model and the following three main components. The models are a 46 l buffer tank for hydraulic decoupling between the loading and the heating circuit (HC), a loading pump that allows the heating water to flow through the condenser, and a fan on the evaporator side. The storage model is discretized into three volumes.

### *2.4. Framework*

As described above, this work is primarily concerned with structuring a calibration process based on Modelica models. For this reason, a two-part internal institute framework consisting of a Python-based and a

Modelica-based part was set up. The Python part is called AixCaliBuHA and stands for “**Aix** (from French Aix-la-Chapelle) **Cali**bration for **Bu**ilding and **H**VAC Systems”. The Modelica part is a library of templates called MoCaTe and stands for “**Mo**delica **Ca**libration **Te**mplates”. The template models of the library allow to be filled with arbitrary models, but designate certain inputs and outputs, which are needed for each calibration procedure (compare Figure 2). In the general pre-processing phase e.g. low pass filters or moving averages can be applied to the time series data. In terms of variable declaration, a fixed naming specifies the following three vectors, which contain the variables necessary for calibration and occur in every calibration: „Measured inputs“ ($mi$), „Target values measured“ ($tv_{\text{meas}}$) und „Target values simulated“ ($tv_{\text{sim}}$). The latter two are compared in the objective function to be minimized and a misfit measure is formed. AixCaliBuHA allows the following misfit measures by default: Mean-absolute error (MAE), mean-square error (MSE), root-mean-square error (RMSE), coefficient of variation of the RMSE (CV(RMSE)), normalized RMSE, coefficient of determination ($R^2$). In the case of the present work it is the weighted chaining of CV(RMSE) [17]. The CV(RMSE) is defined as follows:

$$CV(RMSE) = \frac{\sqrt{1/n \cdot \sum_{j=1}^{n} (tv_{meas} - tv_{sim})^2}}{\overline{tv}_{meas}} \tag{1}$$

With $n$ as the number of data points in the time series and $\overline{tv}_{\text{meas}}$ as the arithmetic mean of the measured time series.

The parameterization of AixCaliBuHA provides for the definition of time segments and the definition of tuner parameters for these time classes with associated constraints in the form of minimal and maximal values. Start values can also be assigned. The core of the framework is the optimization of an objective function under consideration of the mentioned constraints. This objective function always calls the simulation necessary for the aimed period, extracts the simulated target values and compares them with the measured target values as calculation of an integral misfit measure. In the scope of this work this is the CV(RMSE).

The currently implemented optimization algorithms are dlib.find_min_global [18] and scipy.optimize.minimize [19], which in turn allows multiple algorithms. The non-linear optimizer dlib was used in the work. It is important to note that the optimizer cannot only optimize linear problems, since the behavior of the models used is usually non-linear.

In order to describe the dlib library, which provides a Python application programming interface, I quote the description from the website [18]: “Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems. It is used in both industry and academia in a wide range of domains including robotics, embedded devices, mobile phones, and large high performance computing environments. Dlib's open source licensing allows you to use it in any application, free of charge.”

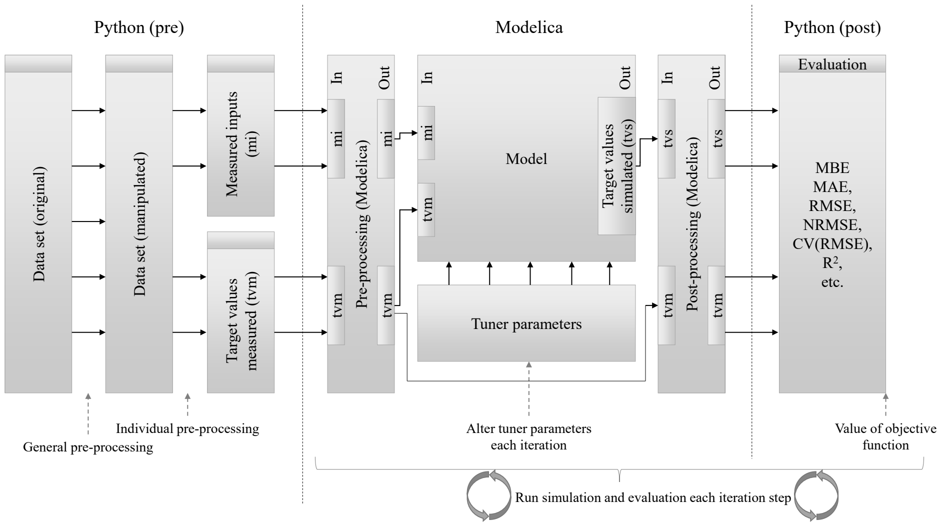

Figure 2: Overview and temporal representation of toolchain

In Modelica, the explicit statement “Evaluate=false” must be in the annotation of all tuner parameters. This ensures that the compiler excludes the parameter from the symbolic pre-processing. In some cases, however, this is not possible, namely when so-called structural parameters are involved which change the system of equations in a structural way. In these cases, the entire model must be recompiled each time in each iteration step of the optimization, which results in a significantly greater expenditure of time. The framework checks if all tuner parameters can be excluded from the symbolic pre-processing of the compilation and informs the user if this is not possible.

Finally, it should be said that the partial framework AixCaliBuHA can be fed with arbitrary forms of models. The use of Modelica models is only the choice during this work. It is therefore conceivable to use Functional Mock-Up Units [20] or models written in other programming languages that can be placed in the AixCaliBuHA framework via defined inputs and outputs.

## 3. Carrying out the calibration

The target values ($tv$) are on the one hand the **heating circuit flow temperature** $\boldsymbol{T_{\mathrm{HC,flow}}}$ and on the other hand the total **electrical power consumption** $\boldsymbol{P_{\mathrm{el}}}$ of the system including powers consumed by fan, pump, controller unit and compressor. The total objective function to be minimized is composed of the chain approach:

$$min \sum f_{tv_i} \cdot CV(RMSE)(tv_i)\ \forall\ tv_i\ \in \{T_{HC,flow}, P_{el}\} \qquad (2)$$

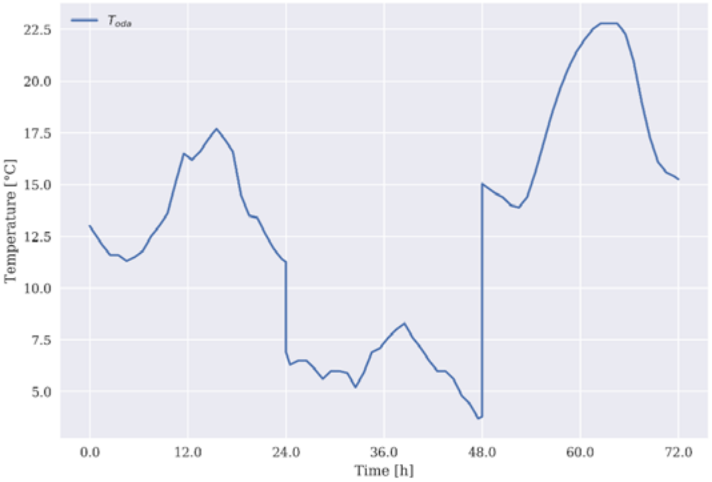

Figure 3: Outdoor air temperature $T_{oda}$ that was emulated by the climatic chamber

Under the premise that the sum of all $f_{tv_i}$ equals 1. The aim is to weight the two targets equally. For this reason, both terms must be in the same order of magnitude, for which a sensible choice of the factors $f_{tv_i}$ is done. For this purpose, we conducted a preliminary investigation that the order of magnitude of $\mathrm{CV(RMSE)}(T_{\mathrm{HC,flow}})$ lies at approximately 0.01 and that of $\mathrm{CV(RMSE)}(P_{\mathrm{el}})$ is about 0.2. Therefore, the factors are $f_{T_{\mathrm{HC,flow}}} = 19/20$ and $f_{P_{\mathrm{el}}} = 1/20$, in order to ensure approximately equal weighting.

The time series data refer to three experimental days in the hardware-in-the-loop test mentioned above. Figure 3 displays the curves of the outside temperature, emulated by the climatic chamber. The test days come from an autumn, winter and summer day and are representative and realistic.

The following variables represent time series variables as „Measured inputs“ given with a brief description each:

- `nSpeedFanEva_1_0`: Relative fan speed (either 0 or 1)
- `TEva_K`: Outdoor air temperature emulated and measured in climatic chamber
- `THCRet_K`: Return temperature of heating circuit measured at hydraulic test bench
- `nSpeedPumpCon_1_0`: Relative pump speed of supply / loading pump near condenser (either 0 or 1)
- `m_flowHC_kg_s`: Mass flow rate through heating circuit measured by hydraulic test bench
- `nSpeedHP_invCtrld_0_1`: Relative and variable compressor speed with a minimum threshold of 25 %

The following variables represent the „Tuner parameters“:

- `scaleHPQCon_nominal`: Scaling factor for tabulated data of default thermal powers for the HP model
- `scaleHPPEl_nominal`: Scaling factor for tabulated data of default electrical powers for the HP model
- `mCon_flow_nominal`: Nominal mass flow rate on water side of the condenser (condenser model derives nominal pressure drop and heat transfer capabilities from this value)
- `mHC_flow_nominal`: Nominal mass flow rate on air side of the evaporator (evaporator model derives nominal pressure drop and heat transfer capabilities from this value)
- `fQWarm`: Additional scaling factor for thermal powers in the range of $T_{\mathrm{oda}} \geq 12$ °C
- `fQMed`: Additional scaling factor for thermal powers in the range of $2\ °\mathrm{C} < T_{\mathrm{oda}} < 12\ °\mathrm{C}$
- `fQCold`: Additional scaling factor for thermal powers in the range of $T_{\mathrm{oda}} \leq 2$ °C
- `fPWarm`: Additional scaling factor for electrical powers in the range of $T_{\mathrm{oda}} \geq 12$ °C
- `fPMed`: Additional scaling factor for electrical powers in the range of $2\ °\mathrm{C} < T_{\mathrm{oda}} < 12\ °\mathrm{C}$
- `fPCold`: Additional scaling factor for electrical powers in the range of $T_{\mathrm{oda}} \leq 2$ °C

- `T_fixAmbTestHall`: Assumed temperature in the test hall, wo which indoor unit, most piping and sensors were exposed to and transferred heat to as losses
- `lambdaIns`: Thermal conductivity of buffer storage envelope
- `tauVolCon`: Time constant that scales the assumed condenser volume of the HP model

In terms of the measured inputs, we conduct chosen individual pre-processing like in the step in Figure 2 states. This is necessary since we do not have all the necessary knowledge of all operational states of the HP For example, we derive the fan speed `nSpeedFanEva_1_0` as well as the pump speed `nSpeedPumpCon_1_0` from one phase of the monitored electrical power consumption. This means the fan and pump is enabled when the power consumption exceeds a threshold of 0.5 % of maximal power. Furthermore, we gain the relative and modulating compressor speed `nSpeedHP_invCtrld_0_1` by normalizing the total electrical power between its maximum and minimum for each day. It is necessary to distinguish between the days and, thus, in a certain way between the outdoor temperature since this mainly determines the level of power consumption.

## 4. Results

### *4.1. Training*

The plot of the objective function's values in Figure 4 shows that already from approximately iteration 50 onwards the deviation mainly lies below 0.02.

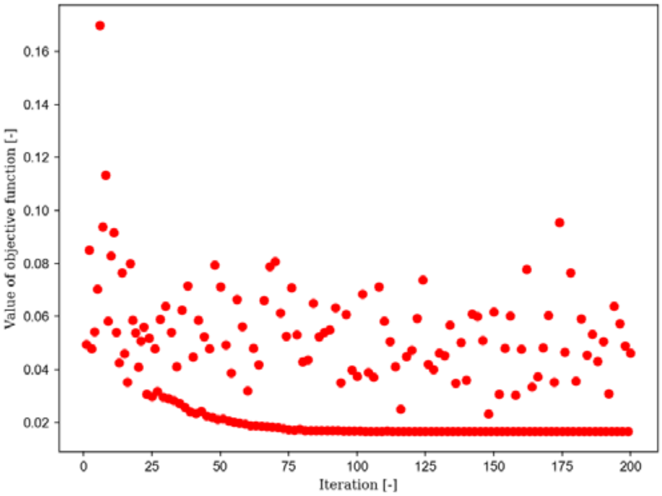

Figure 4: Progress of the objective function's values over iteration steps (training phase)

Figure 5 shows the measured and simulated target values for the 1.5 day-long training phase. The behavior of the dynamics fits predominantly well. In particular with regard to $T_{\mathrm{HC,flow}}$, in some places deviations become apparent, which can be explained with a high degree of probability by the control that was not transferred from real experiment into the simulation domain. In contrast, during the experiments the control of the HP system was active, including pump's pre- and post-run times, heating rod control, partial DHW tank charging. For the calibration, a speed control for the inverter-controlled compressor was derived from the power consumption only. The following target value function results from a quantitative point of view:

$$\left[f_{T_{HC,flow}} \cdot CV(RMSE)\left(T_{HC,flow}\right) + f_{P_{el}} \cdot CV(RMSE)(P_{el})\right]_{Training} \tag{3}$$

$$= \frac{19}{20} \cdot 0.0090 + \frac{1}{20} \cdot 0.1463 = \mathbf{0.0158} \tag{4}$$

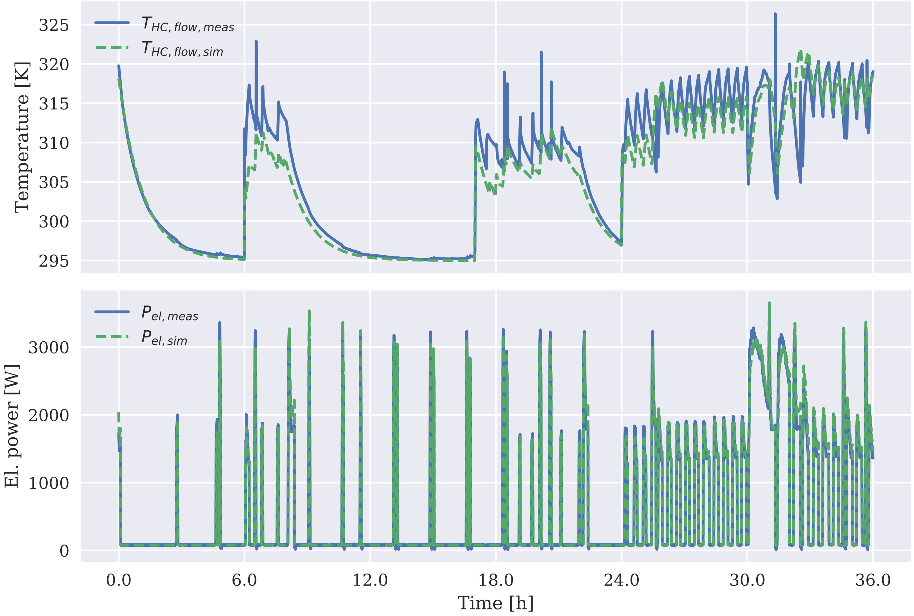

Figure 5: Curves of target values plotted for training data (day 1 and first half of day 2)

Table 1 shows the finally selected tuner parameters' values that the optimization algorithm found. The development of the tuner parameters' value search displays Figure 6.

Table 1: Final values of optimized tuner parameters

| **scaleHPQCon_nominal [-]** | **scaleHPPEl_nominal [-]** | **fQWarm [-]** | **fQMed [-]** | **fQCold [-]** | **T_fixAmbTestHall [K]** | |
|---|---|---|---|---|---|---|
| 0.82 | 0.70 | 0.77 | 1.31 | 1.76 | 295.45 | |
| **mCon_flow_nominal [kg/s]** | **mHC_flow_nominal [kg/s]** | **fPWarm [-]** | **fPMed [-]** | **fPCold [-]** | **lambdaIns [W/m/K]** | **tauVolCon [s]** |
| 0.34 | 0.49 | 1.97 | 1.77 | 1.94 | 0.15 | 150.00 |

### *4.2. Validation*

The data set for the validation consists of two partial days. On the one hand from the second half of the winter day (day 2) and on the other hand a part from the summer day (day 3), since this has a similar behavior to the transition day (day 1). However, the first 25 % of the summer day were excluded from the validation data as there is an incomprehensible discrepancy between measured and simulated values of the target values. This probably stems from a control strategy such as the use of the heating rod, which we could no longer clearly prove from the experimental documentation at the time of calibration. The hardware-in-the-loop tests were performed several months before calibration and the system was already dismantled. The experiments were selected because of their very high measurement resolution of 1 s, as they provide a good basis for calibrating transient models. However, we did not conduct the experiments with this aim in mind.

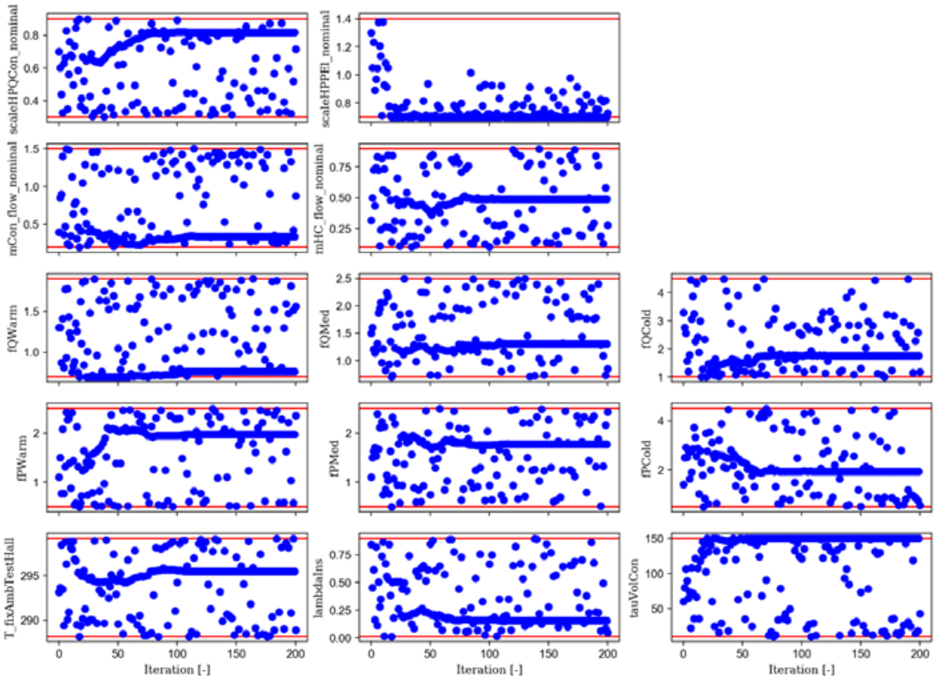

Figure 6: Process of the tuner parameters' value during optimization (red lines: minimal and maximal boundaries)

Figure 7 shows the measured and simulated target values for the two periods mentioned above. Hereby, the time series parts of day 2 and day 3 are concatenated.

The calculation of the value of the combined objective function looks as follows:

$$\left[f_{T_{HC,flow}} \cdot CV(RMSE)\left(T_{HC,flow}\right) + f_{P_{el}} \cdot CV(RMSE)(P_{el})\right]_{Validation} \tag{5}$$

$$= \frac{19}{20} \cdot 0.0085 + \frac{1}{20} \cdot 0.1343 = \mathbf{0.0148} \tag{6}$$

## 5. Conclusions

In the scope of this work, we develop and present a toolchain to calibrate transient simulation models of HVAC plants. This framework is preliminary based on the programming language and can be interfaced to a sub-framework of the simulation model. In the case of this work this is the open-source and object-oriented modelling language Modelica. The calibration framework provides general as well as individual pre-processing possibilities and multiple evaluation strategies with focus on time series data. These evaluation strategies create key performance indicators by comparing the measured to the simulated time series of defined target values. These key performance indicators are meant to be minimized by one of the implemented optimizers.

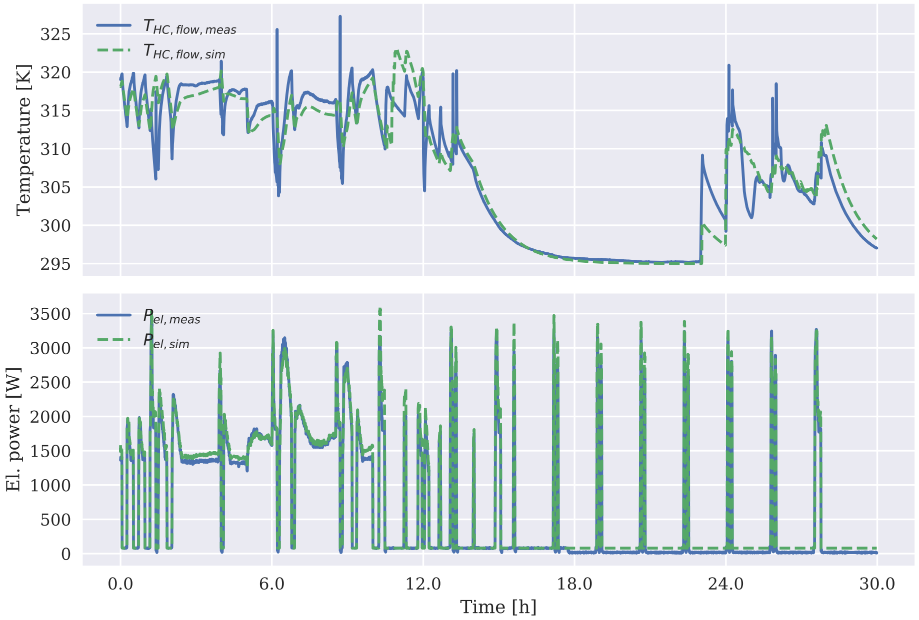

Figure 7: Curves of target values plotted for validation data (part of day 2 and 3 concatenated)

The use case is the calibration of an air-to-water heat pump, of which we gained high-frequent measurement data of a hardware-in-the-loop test that we conducted in our institute's laboratory. Since the boundary conditions of this 3-day long experiment are volatile and covering a broad range, e.g. in terms of outdoor air temperature, the data suits well for the application in a calibration process.

To calibrate the air-source heat pump we split the measurement time series into 60 % training and 40 % validation data. With 200 Iterations for calibrating the model on the basis of the chosen training set, we achieve a coefficient of variation of the root-mean-square error CV(RMSE) = 0.0158. For this purpose, we use 13 tuner parameters. The validation reaches an even better value of CV(RMSE) = 0.0148. Whilst the target value $P_{\mathrm{el}}$ (electrical power consumption) shows a satisfying congruence in terms of dynamics and level of the values, the other target value $T_{\mathrm{HC,flow}}$ (heating circuit's flow temperature) displays in the time series plot solely in terms of dynamics good results. Offsets within certain time spans are probably due to neglecting control phenomena during the calibration process.

Especially the latter mentioned effect asks for further improvement of the process. The suggestion is to have individual calibrations for individual components like the heat pump unit itself, the hot water storages and the controller. For this reason, monitoring equipment must be installed with focus for a feasible calibration. Furthermore, the calibration of a controller is a challenging task due to lots of binary states like mode for domestic hot water or space heating.

## Acknowledgements

We gratefully acknowledge the financial support by Federal Ministry for Economic Affairs and Energy (BMWi), promotional reference 03ET1595A.

## References

[1] T. Nowak und P. Westring, *European Heat Pump Market and Statistics Report 2018,* v1.6 Hrsg., The European Heat Pump Association AISBL, Hrsg., Brussels, Belgium, 2018.

[2] R. W. Wimmer, „Regelung einer Wärmepumpenanlage mit Model Predictive Control," Eidgenössiche Technische Hochschule, Zürich, 2004.

[3] T. Storek, A. Esmailzadeh, P. Mehrfeld, M. Schumacher, M. Baranski und D. Müller, „Applying Machine Learning to Automate Calibration for Model Predictive Control of Building Energy Systems," *Building Simulation 2019: 16th Conference of IBPSA,* 2019.

[4] T. G. Trucano, L. P. Swiler, T. Igusa, W. L. Oberkampf und M. Pilch, „Calibration, validation, and sensitivity analysis: What's what," *Reliability Engineering & System Safety,* Bd. 91, Nr. 10-11, p. 1331–1357, 2006.

[5] G. Dall'O', L. Sarto, N. Sanna und A. Martucci, „Comparison between predicted and actual energy performance for summer cooling in high-performance residential buildings in the Lombardy region (Italy)," *Energy and Buildings,* Bd. 54, p. 234–242, 2012.

[6] E. Fabrizio und V. Monetti, „Methodologies and Advancements in the Calibration of Building Energy Models," *Energies,* Bd. 8, Nr. 4, p. 2548–2574, 2015.

[7] D. Coakley, P. Raftery und M. Keane, „A review of methods to match building energy simulation models to measured data," *Renewable and Sustainable Energy Reviews,* Bd. 37, p. 123–141, 2014.

[8] M. Nürenberg, P. Mehrfeld, K. Huchtemann und D. Müller, „Hardware-in-the-Loop test bench setup and its application to determine seasonal performance of heat pump systems," 2017.

[9] P. Mehrfeld, M. Nürenberg, M. Knorr, L. Schinke, M. Beyer, M. Grimm, M. Lauster, D. Müller, J. Seifert und K. Stergiaropoulos, „Dynamic evaluations of heat pump and micro combined heat and power systems using the hardware-in-the-loop approach," *Journal of Building Engineering,* Bd. 28, p. 101032, 2020.

[10] A. Miehlich, „Studies on the repeatability of Hardware-in-the-loop experiments," RWTH Aachen University, Aachen, 2017.

[11] Modelica Association, „Modelica," 2019. [Online]. Available: https://modelica.org/.

[12] AixLib, *AixLib - A Modelica model library for building performance simulations,* 2018.

[13] D. Müller, M. Lauster, A. Constantin, M. Fuchs und P. Remmen, „AixLib - An Open-Source Library within the IEA-EBC Annex60 Framework," 2016, p. 3–9.

[14] IBPSA, *IBPSA Project 1: BIM/GIS and Modelica Framework for building and community energy system design and operation,* 2018.

[15] International Energy Agency, *IEA EBC Annex 60: New generation computational tools for building and community energy systems based on the Modelica and Functional Mockup Interface standards,* 2018.

[16] DIN EN 14511-3, *Air conditioners, liquid chilling packages and heat pumps for space heating and cooling and process chillers, with electrically driven compressors - Part 3: Test methods; EN 14511-3:2018,* Bd. 27.080; 91.140.30, 2018.

[17] R. J. Hyndman und A. B. Koehler, „Another look at measures of forecast accuracy," *International Journal of Forecasting,* Bd. 22, Nr. 4, p. 679–688, 2006.

[18] D. E. King, „Dlib," 2019. [Online]. Available: http://dlib.net/.

[19] Scipy, „SciPy: Scientific Library for Python (Function: optimize.minimize)," 2019. [Online]. Available: https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html.

[20] T. Blockwitz, M. Otter, J. Akesson, M. Arnold, C. Clauss, H. Elmqvist, M. Friedrich, A. Junghanns, J. Mauss, D. Neumerkel, H. Olsson und A. Viel, „Functional Mockup Interface 2.0: The Standard for Tool independent Exchange of Simulation Models," Linköping University Electronic Press, 2012, p. 173–184.