Optimizing operational control of a compression chiller system with photovoltaics and an electrical battery

Alex Hobé[a], Neha Dimri[a], Mikel Arenas-Larrañaga[a], Daniel Carbonell[b]

[a]SPF Institute for Solar Technology, Ostschweizer Facchochschule, Rapperswil 8640, Switzerland
[b]DCarbo Energy Consulting SA, Barcelona, Spain

Abstract

Cooling currently accounts for ~20% of the world's total energy consumption in buildings. This is projected to increase by 2050, when two-thirds of the global households will likely have air conditioners. As a result, the pressure on the electricity grids will also increase (especially during peak load hours). Reducing electricity peaks in cooling systems is, therefore, crucial and could be achieved using storage technologies to shift the demand. To this aim, we are investigating a compression-chiller-based cooling systems with photovoltaics (PV) coupled with an electrical battery. In addition to reducing the peak load, such a system would also benefit from storing surplus PV or grid electricity at lower tariffs. An electrical battery could also shift the non-cooling electricity load of the building. As the economics and energy efficiency of this system is tied to the combination of electricity prices, and weather and cooling demands, it has many opportunities for control optimization. To show the impact of this optimization, this paper aims to compare results using a rule-based control to results using model-predictive control. The system will be modelled within a detailed dynamic simulation environment for an office building in Chur, Switzerland. The model-predictive controller will interact with the dynamic simulations using control signals from a MILP-based mathematical optimizer. The optimized control aims to minimize the total operation cost, i.e. the cost of electricity consumption, assuming dynamic electricity prices. In the initial results, the application of model-predictive control demonstrated a significant reduction in operation costs compared to the rule-based control, highlighting its potential to enhance economic performance in PV-battery cooling systems.

Selection and/or peer-review under the responsibility of the organizers of the 15th IEA Heat Pump Conference 2026.

Keywords: Cooling; Model-Predictive Control; Rule-Based Control, TRNSYS, optimization

1. Introduction

Chillers provide cooling around the world. ~20% of the total energy used in buildings worldwide is in the form of cooling. As temperatures increase, the importance of air-conditioning (AC) will increase further [1]. This provides both a challenge and an opportunity. The increase in electrical consumption for AC will put additional pressure on the electricity grids. Additionally, as the energy transition progresses, more and more intermittent sources of electricity are integrated into Europe's electricity grids, which also complicates the production side. This has already led to changing electricity costs throughout the day, which in turn has changed electricity tariffs for customers in many countries. Where before a high/low split during the week and on weekends was mainly based on customer behavior, novel pricing schemes currently range from multi-window fixed schemes (e.g. Spain) [2], to fully dynamic schemes (e.g. Netherlands) [3].

The increased variability in prices opens an opportunity for optimized control, which will benefit both the customers and the grid operators. The customers get reduced costs by using the flexibilities in their systems, while the grid operators use the price to lower energy peaks, which is crucial to ensure grid stability.

Previous studies have investigated the potential for Model-Predictive Control (MPC) to optimize the operation of cooling systems [4-6]. The control of a thermal energy storage with a chiller run only using grid electricity is quite straightforward and could be achieved using a Rule-Based Control (RBC). The problem becomes considerably more complex when adding PV and a battery. When analyzing the benefits of MPC algorithms, such systems could see cost reductions in the order of 10% [4,6]. However, in real systems there will be many sources of discrepancies between the MPC model and the actual system operation.

This investigation takes a step towards applying MPC to real systems, by coupling MPC with a detailed simulation environment (TRNSYS [7]) using a novel co-simulation framework. The detailed simulation closely mirrors the behavior of a real system, distinguishing it significantly from the simplified MPC model. The focus lies on a cooling system for an office building with PV, a battery, and a thermal energy storage.

2. Approach

This Section describes the system under investigation, the logic behind the rule-based control (RBC), the implementation of the MPC, the co-simulation framework, and the required timeseries data.

2.1. Cooling system for an office building

Fig. 1 shows the cooling system under investigation. It includes: i) a chiller to meet the cooling demands, ii) a mixing valve to ensure a reliable temperature is provided to the distribution system, iii) a small thermal energy storage to further stabilize the supply temperature and to meet any demands below the minimum capacity of the chiller, iv) PV as a decentral source of electricity, v) a battery to shift electricity demands, and vi) the connection to the grid. Electricity from the grid is never used to charge the battery.

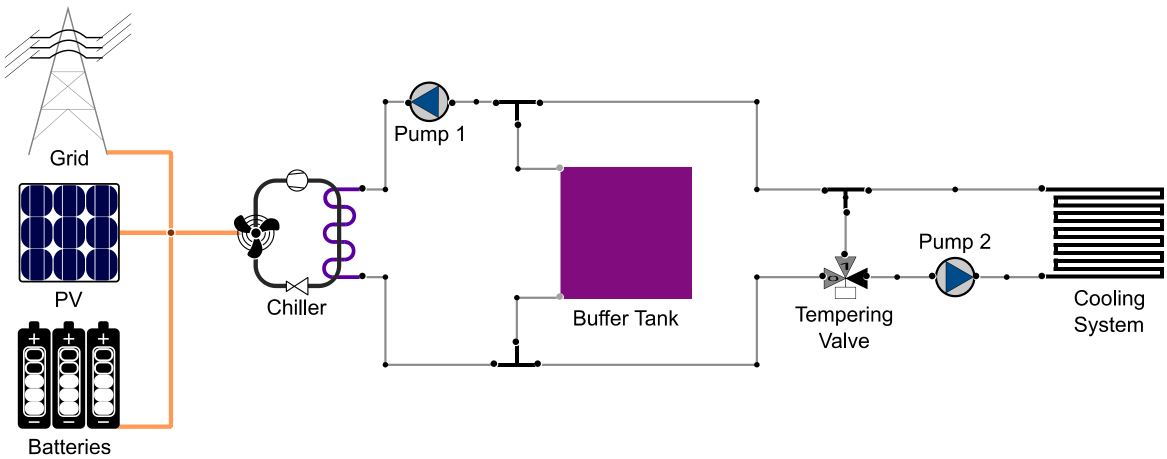

Figure 1. Simplified diagram of a cooling system for an office building. Visualized using pytrnsys [8].

2.2. Rule-Based Control

The rule-based control (RBC) serves as a baseline and was implemented directly in TRNSYS. It uses simple heuristics to maximize the self-consumption of the PV electricity. The chiller is run both when there is cooling demand, and/or when the temperature of the thermal storage falls below the setpoint temperature. The battery is charged whenever there is excess PV electricity and discharged whenever the load goes beyond the current PV generation. When the battery is full, the surplus PV electricity is sent to the grid, while any missing load is met by drawing electricity from the grid. The RBC is purely reactive. Therefore, it cannot think ahead and use the available flexibilities to prepare for upcoming demands.

2.3. Model-predictive Control

The MPC uses new extensions to the optihood optimization framework [9]. This framework uses Mixed-Integer Linear Programming (MILP) models of energy-system components to optimize energy systems in different ways. When designing a new system, optihood can simultaneously optimize i) the optimal system components given certain demands, profiles, and constraints, ii) the optimal sizing of those components, and iii) the optimal operation of the chosen components. The meaning of "optimal" depends on the desired objective, which can be, e.g., total costs and environmental impact. The end results provide an optimized energy system, which takes into account the efficiencies gained by optimal control. Options ii) and iii) can be run together without option i), and the MPC of this study is an example of option iii) being run by itself.

The primary objective of the MPC is to minimize the total operational cost, given the following constraints: a) The physical limitations of the system's components; b) The total building load must be met during each timestep by a combination of PV generation, battery discharge, and grid import; c) The chiller's cooling output is constrained by its maximum capacity and its part-load performance, which is modelled using a linearized efficiency curve; d) The batteries operational limits and efficiencies of charging and discharging; e) The total power calculated for the PV array cannot be exceeded by the total combined power supplied to the building, chiller, battery, and grid. Eqs. (1)-(10) present the mathematical formulation of the optimization problem, defining the objective function, as well as the constraints listed above.

$$\text{minimize} \sum_{t=1}^{N} C_{\text{grid}}(t) \cdot P_{\text{grid}}(t)\, \Delta t \tag{1}$$

where N is the prediction horizon length, $C_{\text{grid}}(t)$ is the grid electricity price in CHF/kWh, $P_{\text{grid}}(t)$ is the imported power from grid (kW), and Δt is the timestep resolution in hours.

$$\mathrm{P_{PV}(t)} \leq \mathrm{P_{PV}^{max}} \tag{2}$$

where $P_{\text{PV}}(t)$ is the power generated by PV and P_{PV}^{max} is the capacity of the PV array (i.e. the maximum power the PV array can produce).

$$E_{\text{bat}}(t) \leq E_{\text{bat}}^{max} \tag{3}$$

where $E_{\text{bat}}(t)$ is the state-of-charge of the battery (kWh) and E_{bat}^{max} represents the maximum limit based on its capacity (kWh).

$$Q_{\text{chiller}}(t) \leq Q_{\text{chiller}}^{max} \tag{4}$$

where $Q_{\text{chiller}}(t)$ is the chiller's cooling output (kW) and Q_{chiller}^{max} is the maximum chiller cooling capacity.

$$E_{\text{tes}}(t) \leq E_{\text{tes}}^{max} \tag{5}$$

where $E_{\text{tes}}(t)$ is the state-of-charge of the thermal energy storage tank (kWh) and E_{tes}^{max} represents its capacity.

$$P_{\text{load}}(t) = P_{\text{PV,used}}(t) + P_{\text{bat,dis}}(t) + P_{\text{grid}}(t) \tag{6}$$

where $P_{\text{load}}(t)$ is the building's electrical load (kW), $P_{\text{PV,used}}(t)$ is PV power allocated to the building's load (kW), and $P_{\text{bat,dis}}(t)$ is battery's discharge power (kW).

$$Q_{\text{chiller}}(t) = EER(t) \cdot P_{\text{chiller}}(t) \tag{7}$$

where *EER(t)* is the chiller's energy efficiency ratio.

$$E_{\text{bat}}(t+1) = E_{\text{bat}}(t) + \eta_{\text{ch}} \cdot P_{\text{bat,ch}}(t)\, \Delta t - \frac{1}{\eta_{\text{dis}}} \cdot P_{\text{bat,dis}}(t)\, \Delta t \tag{8}$$

where η_{ch} is battery charge efficiency, η_{dis} is battery discharge efficiency, and $P_{\text{bat,ch}}(t)$ is battery's charging power (kW).

$$0 \leq P_{\text{bat,ch}}(t) \leq P_{\text{bat,ch}}^{max} \tag{9}$$

where $P_{\text{bat,ch}}^{max}$ is the maximum battery charging power (kW). Similar constraint is defined to limit the battery's discharge power $P_{\text{bat,dis}}(t)$ to a maximum $P_{\text{bat,dis}}^{max}$.

$$P_{\text{PV}}(t) = P_{\text{PV,used}}(t) + P_{\text{PV,ch}}(t) + P_{\text{PV,grid}}(t) \tag{10}$$

where $P_{\text{PV,used}}(t)$ is PV power used for battery charging (kW) and $P_{\text{PV,grid}}(t)$ is PV power fed into the grid (kW).

The MPC uses a rolling time-window. For each call to the MPC, this time-window is shifted by the temporal distance between calls (e.g. 20 minutes). This allows the optimizer to take into account relevant upcoming loads, weather predictions, and electricity prices, while making regular adjustments based on the actual system state. Because of the MILP implementation, as well as predictive nature of the weather and price forecasts, these discrepancies are expected to grow with time. However, for the sake of this study, both the weather and price forecasts are perfectly known (i.e. perfect forecasts), and the feed-in tariff for PV electricity is zero. This limits the variables in the study, allowing us to focus on implementing the optihood extensions and the novel co-simulation framework.

2.4. Novel co-simulation framework

The novel co-simulation framework (Fig. 2) connects the TRNSYS simulations with the optihood optimization framework. The co-simulation framework additionally includes both an MPC Handler and a Python Interface. The Python Interface is made up of a custom Python script and a TRNSYS type (3157 [10]). This type uses the CFFI protocol [11], with a specified Python environment to connect to the custom Python script. The MPC Handler handles the MPC requirements of the co-simulation framework. These requirements include: i) setting up the optimization scenario, ii) updating the optimization scenario for every call to the optimizer, iii) calling the optimizer, iv) writing the results of the optimizer to file, v) translating the energy flows to control decisions, and vi) communicating with the Python interface.

Whereas the simulations have a timestep of 2 minutes, the MPC is called once every 20 minutes, to predict over a horizon of 24 hours. A full MPC call, using the described components, takes the following steps: a) The current state of the system (state-of-charge of both the battery and the thermal storage) is passed through the Python Interface to the MPC Handler. The MPC Handler updates the optimization scenario with i) the current system state, ii) the new prediction horizon, and iii) the associated profiles for the new prediction horizon. Next, the MPC Handler calls optihood to optimize the updated scenario. The optimized energy flows between components are passed back to the MPC Handler, which saves them to file and converts the flows to control decisions (e.g. charge/discharge battery, HP on/off, pump flows, electricity flows to/from grid). These decisions are passed back to the simulation through the python interface, which adds additional information about the state of the interface, e.g. "updated results received" vs "failure due to": e.g. convergence issues, error encountered. This latter feature allows the simulation to continue when the MPC has failed to make a prediction.

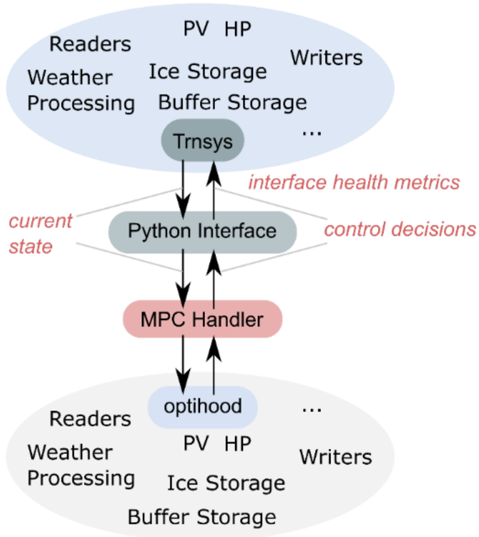

Figure 2. Architectural overview of the co-simulation framework.

2.5. Price tariffs, weather data, and cooling demand

Both TRNSYS and optihood are provided with the same electricity price and weather profiles. The weather profiles and cooling demands correspond to an office building in Chur, Switzerland. The weather data (solar radiation and ambient temperature) is a Typical Meteorological Year [12]. The electricity price profile (Fig. 3) is a dynamic electricity tariff, with varying prices for peak, off-peak, and mid-peak hours.

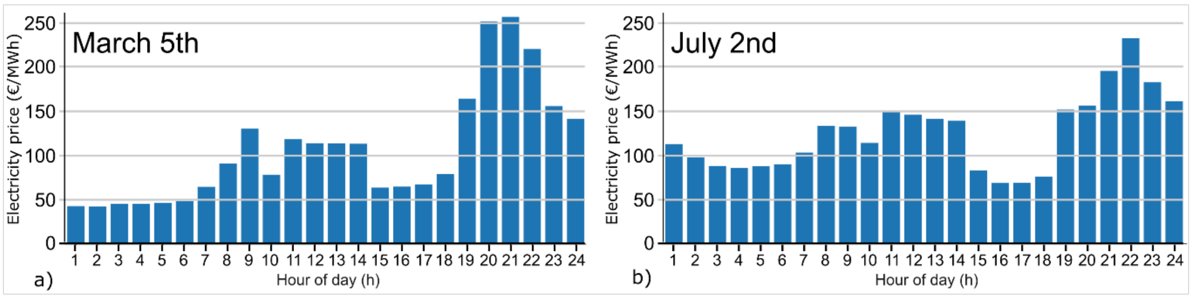

Figure 3. Hourly electricity prices for a single day in a) March, and b) July.

3. Results and discussion

3.1. Initial results

Figs. 4 and 5 show the initial results of both the RBC and MPC simulations for a single day in March (Fig.4) and July (Fig 5.), respectively. These hourly energy balances show how much energy flows in and out of a given component during that hour.

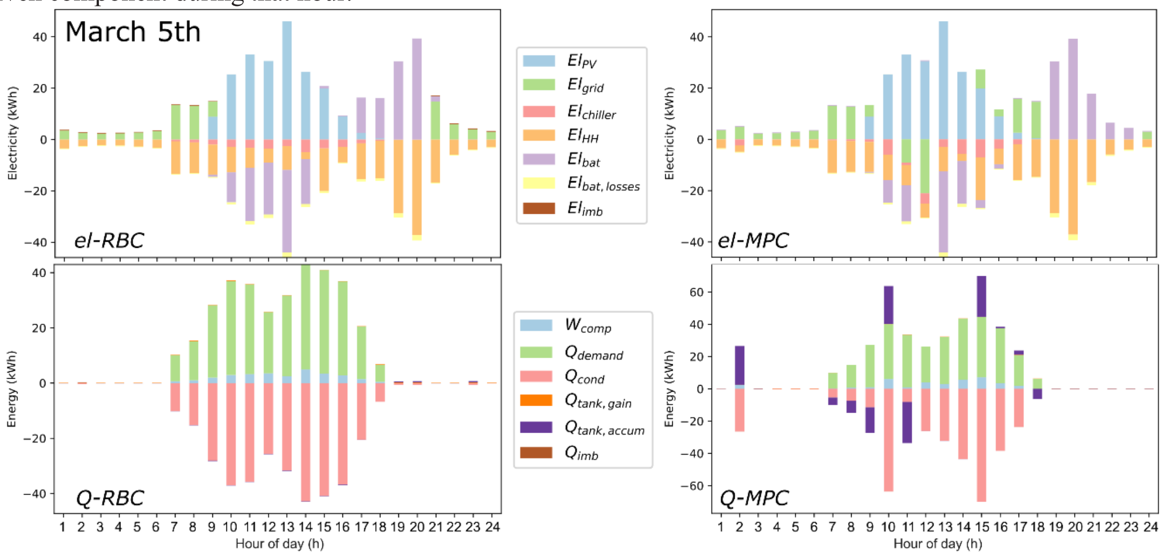

Figure 4. Initial comparison (March) between the rule-based control (RBC) and the model-predictive control (MPC) results. Shown are hourly energy balances for both electric (el) and cooling (Q) energy.

The constant values between the two simulations are the cooling demands, the household-electricity demands, and the produced PV electricity. The biggest differences on the 5th of March (Fig. 4) occur in the usage of the thermal storage and the grid feed-in. The RBC does not find surplus energy to feed to the grid, whereas the MPC chooses to feed to the grid during two hours (11:00 and 12:00). Though the results indicate that the storage is oversized, the MPC makes better use of the flexibility the storage provides. This has consequences on the usage of the chiller, which in turn impacts the electricity fluxes to/from both the battery and the grid. Lastly, the RBC depletes the battery earlier, leading to electricity being pulled from the grid at the most expensive hour (21:00 in Fig. 3a). The MPC instead pulls electricity from the grid before and after the electricity price peak between 19:00 and 23:00, while meeting this peak fully using the battery. The peak hourly power consumed from the grid, which was not optimized for, is 16 kW for the RBC and 18 kW for the MPC. When comparing the two simulations, the MPC reduces the total operational cost by 11%.

On the 2nd of July (Fig. 5), very similar behaviors can be observed. The main difference to March: both the MPC and the RBC do not feed into the grid. The thermal storage is also used by the MPC, yet to a lesser extent. The battery is again used more effectively by the MPC to cover the most expensive hours (23:00 and 24:00). The MPC also pulls electricity from the grid at the cheapest times (4:00 and 16:00 – 19:00). For this case, the peak hourly power consumed from the grid is 25 kW for RBC, and 20 kW for MPC, while the MPC reduces the total operational costs by 6%.

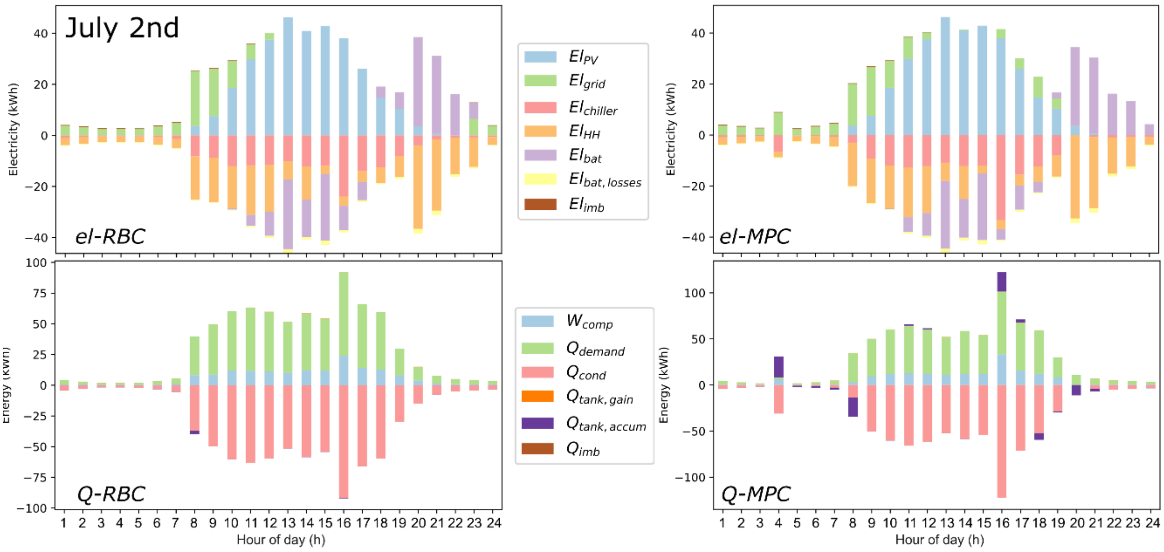

Figure 5. Initial comparison (July) between the rule-based control (RBC) and the model-predictive control (MPC) results. Shown are hourly energy balances for both electric (el) and cooling (Q) energy.

3.2. Discussion

This initial investigation indicates the potential for MPC to reduce cooling costs by making optimal use of a cooling system's flexibilities. However, further work is needed to quantify i) this potential, ii) the sensitivity of this potential to the many variable profiles (e.g. weather and prices), iii) the trade-off between computational cost, prediction frequency, and prediction accuracy, and iv) discrepancies between forecasts of, e.g., weather and electricity prices, and reality.

Most of these will be investigated in the SolIceTes project. Additionally, the design of a new cooling system could take into account the optimal operation using MPC. Though this is not the focus of this study, optihood provides this option to optimize the sizing of a system's components with optimal operation in mind.

4. Conclusions and outlook

This initial investigation compares the operation of the cooling system for an office building from two perspectives: rule-based control (RBC), and model-predictive control (MPC). To simulate differences between the idealized MPC and real-world systems, we developed a novel co-simulation framework and added extensions to the optihood optimization framework. Though the initial results cannot yet be used to quantify the full potential of MPC to optimize the operation of cooling systems, the presented cost reduction already shows what an MPC may be capable of using very little flexibility.

Adding more flexibility to the system would provide more options for the MPC to work with, which would have to be warranted with an associated variability in electricity prices. As both the energy transition and climate change progress, the variable factors impacting cooling (and other systems) will increase in their variability. Because of this, RBCs will be less and less likely to cope effectively with the requirements put upon them, while both optimal control and appropriate flexibilities in each system will be ideally suited to meet these new requirements.

Acknowledgements

The authors acknowledge the financial support received from the Swiss Federal Office of Energy (SFOE) within the scope of the research project SolIceTES to finalize the work presented in this article.

References

[1] EA (2018). The Future of Cooling, IEA, Paris, License: CC BY 4.0, https://www.iea.org/reports/the-future-of-cooling.
[2] Today’s electricity prices by hour, Octopus Energy: https://octopusenergy.es/en/precio-luz-hoy (accessed on December 1st, 2025)
[3] Current dynamic energy prices ANWB-Energy: https://www.anwb.nl/energie/actuele-tarieven (accessed on December 1st, 2025)
[4] Hao, K. (2020). COMPARING THE ECONOMIC PERFORMANCE OF ICE STORAGE AND BATTERIES FOR BUILDINGS WITH ON-SITE PV THROUGH MODEL PREDICTIVE CONTROL. thesis, Purdue University Graduate School.
[5] Henze, G. P., Dodier, R. H., and Krarti, M. (1997). Development of a Predictive Optimal Controller for Thermal Energy Storage Systems. HVAC&R Research, 3(3):233–264.
[6] Thiem, S., Bom, A., Danov, V., Sch¨afer, J., and Hamacher, T. (2016a). Optimized cold storage energy management: Miami and los angeles case study. In 2016 5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS), pages 1–8. IEEE.
[7] Beckman, W. A., Broman, L., Fiksel, A., Klein, S. A., Lindberg, E., Schuler, M., and Thornton, J. (1994). Trnsys the most complete solar energy system modeling and simulation software. Renewable energy, 5(1-4):486–488.
[8] Institute for Solar Technology (SPF)-Eastern Switzerland University of Applied Sciences (OST). Pytrnsys: The Python TRNSYS Tool Kit. 2025. Available online: https://pytrnsys.readthedocs.io/en/latest/ (accessed on December 1st, 2025).
[9] Dimri, N., Zenhäusern, D., Carbonell, D., Jobard, X., Jacquot, V., Francois, A., Capezzali, M., and Fesefeld, M. (2023). Optihood – a multi-objective analysis and optimization framework for building energy systems at neighborhood scale. Journal of Physics: Conference Series, 2600(8):082027.
[10] Bernier, N., Marcotte, B., and Kummert, M. (2022). Calling python from trnsys with cffi. Polytech. Montréal.
[11] Rigo, A., & Fijalkowski, M. (2018). CFFI documentation. *CFFI 2.0.0 documentation*: https://cffi.readthedocs.io/en/latest/index.html
[12] Lukasczyk, C., Amsler, A., Fuhrer, O., & Biegger, S. (2025). The Introduction of Open Data at MeteoSwiss: Development, Set up, and Lessons Learned (No. EMS2025-620). Copernicus Meetings.