A dynamic bandwidth tariff assessment in a Dutch distribution network using a novel scalable distributed simulation framework

van der Holst, Bart ; Verhoeven, Gijs; Matthijssen, Edwin; Vrijlandt, Mark ; van der Meer, Arjen; Plug, Ruduan; Kok, Koen

DOI
10.1049/icp.2023.0722

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the 27th International Conference on Electricity Distribution (CIRED 2023)

Citation (APA)
van der Holst, B., Verhoeven, G., Matthijssen, E., Vrijlandt, M., van der Meer, A., Plug, R., & Kok, K. (2023). A dynamic bandwidth tariff assessment in a Dutch distribution network using a novel scalable distributed simulation framework. In *Proceedings of the 27th International Conference on Electricity Distribution (CIRED 2023)* (6 ed., Vol. 2023, pp. 3779-3783). (IET Conference Proceedings). IET. https://doi.org/10.1049/icp.2023.0722

A DYNAMIC BANDWIDTH TARIFF ASSESSMENT IN A DUTCH DISTRIBUTION NETWORK USING A NOVEL SCALABLE DISTRIBUTED SIMULATION FRAMEWORK

Bart van der Holst*[1], Gijs Verhoeven[1], Edwin Matthijssen[3], Mark Vrijlandt[3], Ruduan Plug[3], Arjen van der Meer[2], and Koen Kok[1]

[1]*Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands*
[2]*Department of Electrical Engineering, Mathematics, and Computer Science, Delft University of Technology*
[3]*TNO, Den Haag, The Netherlands*
**b.v.d.holst@tue.nl*

ABSTRACT

Due to the increasing penetration of distributed energy resources, congestion problems are already emerging in Dutch distribution grids. The available flexibility of assets in the built environment could have the potential to reduce congestion if prosumers are properly incentivized by distribution grid operators (DSOs). However, it is not yet clear what (combinations of) flexibility a ctivation mechanisms will be effective for congestion management in Dutch Distribution grids. To shed light on this issue, the GO-e consortium aims at performing large-scale agent-based simulations of up to 120 low-voltage networks and a large variety of possible instruments and scenarios. For this reason, we developed a novel scalable time-discrete simulation framework for distributed agent-based simulations of energy systems. We demonstrate the framework on a case-study in which we assess the effectiveness of a dynamic bandwidth tariff instrument on overloading problems in a low-voltage network containing solar panels, batteries, and heat pumps. It was shown that a dynamic bandwidth tariff can successfully resolve forecasted congestion if the associated costs are high enough compared to the day-ahead prices. However, the resulting load shifting can cause new congestion intra-day aswell.

INTRODUCTION

The energy transition results in an increasing penetration of (renewable-based) distributed energy resources (DERs) such as photovoltaics (PV), heat pumps (HP) and battery energy storage systems (BESS) in the lower levels of the distribution networks. In the Netherlands, this increase already causes congestion and voltage challenges in considerable parts of the distribution networks. The traditional approach from distribution system operators (DSOs), preventing congestion problems by expanding and reinforcing the network, is not always a feasible solution due to high costs, limited manpower, and long lead times. Within the GO-e consortium, consisting of the three largest Dutch DSOs, energy suppliers, market platforms, universities, research institutes, and engineering companies [1], the potential of flexibility from DERs in the built environment for congestion management in the distribution networks is explored.

In theory, a DSO could use various instruments to incentivize controllers of flexible resources to allocate flexibility to help resolve the congestion (e.g. tariff structures, market-based solutions, non-market based (re)dispatching). To assess the effectiveness of different combinations of these so-called flexibility activation mechanisms, interactions between assets and prosumers in low-voltage grids, DSOs, electricity markets, and the instruments will be studied in large-scale simulations. The results could contribute to the legalization of effective (combinations of) instruments for Dutch DSOs. Future research will entail analyzing different combinations of flexibility activation mechanisms in 120 Dutch low voltage (LV) networks under various market and weather conditions and various penetrations of flexible assets. These simulations will contain networks with over 100 houses, possibly containing smart controllers optimizing their dispatch. An agent-based simulation could be a good approach, but the high computational burden of the simulation task requires the possibility to perform the simulations in a distributed manner on external machines.

Some existing off-the-shelf software for agent-based simulations of energy systems are Mosaik [2], Gridlab-D [3], and Anylogic [4]. However, none of these solutions were designed for tailored distributed simulations on external machines. While Mosaik Docker allows for model containerization, the containerizing options are very limited and all model communication goes through a central component, limiting scalability. For Gridlab-D, distributed simulation is not discussed in the available documentation. Anylogic provides cloud computing in Anylogic cloud, but the computational resources and their usage are abstracted away from the user, limiting the control of the user. Other academic studies present frameworks for distributed agent-based simulations but are not built for scalability [5][6], or were never made open-source and are not accessible to other researchers [7].

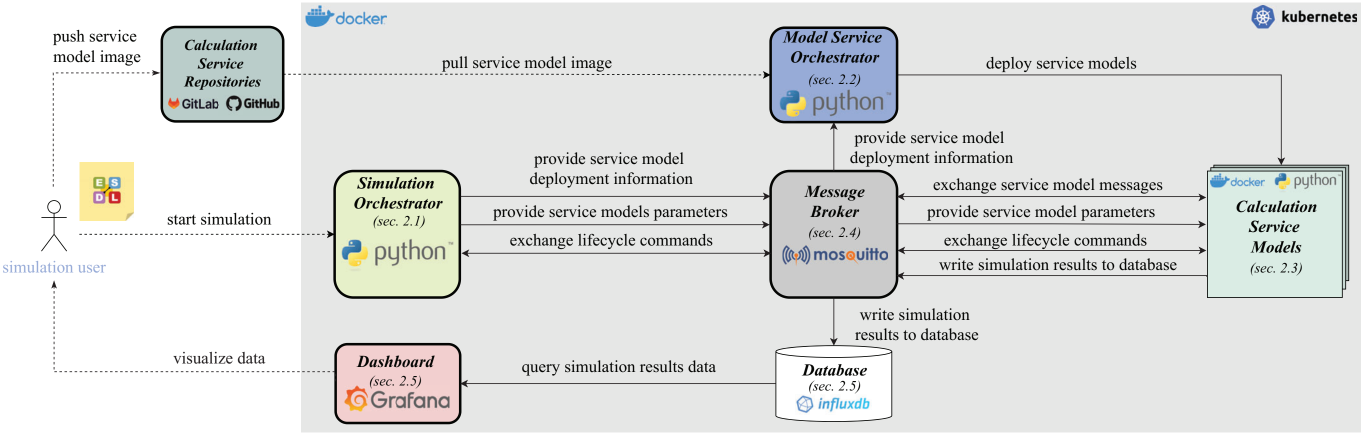

Fig. 1. Overview of the developed scalable distributed simulation framework for time-discrete simulations of energy systems.

In this study, we therefore develop a new open-source simulation framework for time-discrete distributed agent-based simulations of flexibility activation mechanisms in LV-voltage networks. The contributions of this framework are:

- The ability to flexibly distribute the computational load of the simulated agents over Kubernetes clusters, which can be acquired from any large cloud provider.
- It facilitates distributed and transparent development of the agent models.
- All simulation components are described in 1 file written in a uniform energy system description language.

The framework is demonstrated in a case study in which we investigate the effectiveness of a dynamic bandwidth tariff on cable/transformer overloading in a low-voltage network. The developed framework is presented in Section 2. The case study is introduced in Section 3. The paper concludes with the simulation results in Section 4, which demonstrate the framework in the case study and the conclusions 5.

2 Simulation framework

Fig. 1 presents our new scalable distributed simulation framework for time-discrete simulations of energy systems. It consists of a Simulation Orchestrator (SO), a Model Service Orchestrator (MSO), several calculation service models, a message broker, and a database and dashboard. All simulation components are deployed on a Kubernetes cluster. Kubernetes is a container orchestration platform that automatically manages containers over 1 or more machines. These clusters are offered by several cloud providers and the presented framework does not depend on a particular cloud provider. In our simulation framework, calculation service models represent the agent types in our simulation (assets, controllers, etc.) and their instances can be distributed over one or more Docker containers, depending on the desire to parallelize the computations of the instances of the calculation services. Before every simulation, the user needs to 1) publish the docker images of the Python code of the calculation service models to a GitHub/GitLab repository, and 2) provide all relevant simulation information to the Simulation Orchestrator via a FastAPI REST interface. Using a template, the user provides: the simulation name, the simulation start date, the number of simulation time steps, a list of calculation service models in the simulation, the desired number of calculation models per container, the calculation service image locations (on the GitHub/GitLab), and a description of the energy system via an HTTP post-request. The energy system is described in the Energy System Description Language (ESDL), which is an open-source modeling language developed by TNO [8]. Within this language, not only physical assets and their connections can be described, but also information on user profiles, weather information, electricity markets, and parties like DSOs can be stored in a uniform format. The various simulation components and their functioning will now be described one by one.

2.1 Simulation Orchestrator

The SO provides a FastAPI REST interface to the user and is the coordinator of the simulation. After obtaining the post request via the API, it will first deduce the type and number of calculation models to be deployed and sends this information to the MSO. After the models are deployed on the Kubernetes clusters, it will parameterize all the calculation models based on the ESDL file. From that moment on, it continuously sends life-cycle commands to the calculation models and checks if all performed their calculations.

2.2 Model Service Orchestrator

The Model Service Orchestrator (MSO) is a python application that makes use of a python Kubernetes client to deploy calculation models onto the Kubernetes cluster. It receives model information from the SO and deploys the models using the docker images pulled from the GitHub/GitLab. The multiple instances of a calculation service model (e.g. a Home Energy Management System) can be distributed flexibly over 1 or more containers. Using many containers for a service allows for a higher degree of parallelization, and thus greater speed, but increases the memory use of the simulation on the cluster. When a model is successfully deployed, it will send a message to the SO. The SO waits until all models are successfully deployed.

2.3 Calculation Service Models

The calculation service models represent the agent types to be modeled in the energy system (e.g. PV-panels, HPs, low-voltage grids, a DSO, the weather, etc.). The services need to be written in Python in a specific template, of which the boilerplate code can be automatically generated (not shown in Fig. 1). After a model is deployed by the MSO, it will receive parameterization information from the SO, containing the ESDL-file of the energy system. From the file, the model instances deduce from which other models they require input and subscribe themselves to messages sent by these models.

2.4 Message Broker

All communication between the SO, MSO, the calculation service models, and the database on the Kubernetes cluster is done via a message broker. This way, the communication between the models is not coordinated at a central point, thereby greatly improving the scalability of the framework. In this work, the open-source Eclipse Mosquitto message broker is used, which uses the lightweight MQTT-protocol for communication [9]. The messages are serialized using Google's Protocol Buffer.

2.5 Database and Dashboard

All simulation results in this study can be represented by time series. For this reason, an InfluxDB database is set up on the Kubernetes cluster. The calculation models can push stored state variables like battery state of charge, house temperatures, or voltage and current magnitudes in the network to this database. To visualize the stored simulation results, an instance of the dashboarding tool Grafana is also set up on the Kubernetes cluster.

3 Simulation Case Study

To demonstrate the workings of the proposed simulation framework, we consider a case study in which we assess the effectiveness of a dynamic bandwidth (BW) tariff on cable/transformer overloadings in a LV network. An overview of the agents and their interactions in the simulation is presented in Fig. 2. The physical system consists of the LV network, households having a connection to the network, and the local weather. The households all contain an inflexible base load, and possibly a selection of flexible PVs, HPs and BESSs. Each of the connections will also have a Home Energy Management System (HEMS) which will optimize the dispatch of the connected flexible assets based on the BW tariff information from a DSO, day-ahead (DA) prices from the DA-market, and weather forecasts.

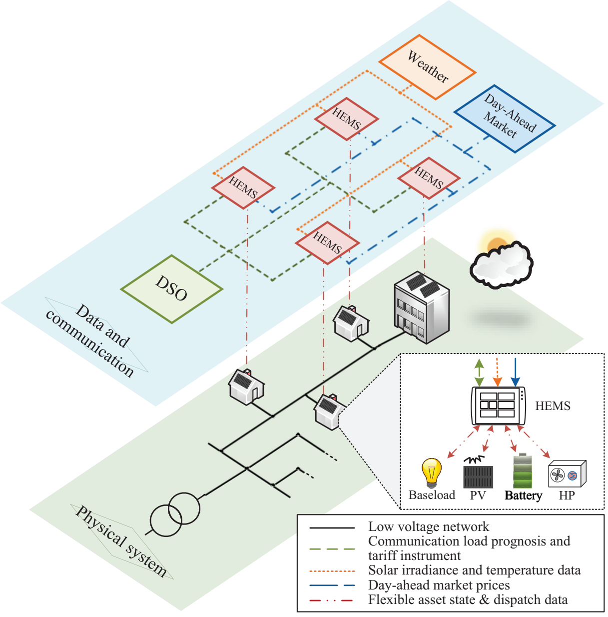

Fig. 2 Overview of the agents and interactions in the simulation case study systems.

To model the DSO's knowledge about the future state of the network, we assume that all HEMSs send a DA load prognosis to the DSO once a day at 12 PM, right after the DA market closes. Based on these prognoses, the DSO performs a load flow calculation to determine possible cable/transformer overloading the next day. Whenever the prognosis load flow calculation from the DSO indicates that overloading problems are expected at certain time steps, a BW network tariff will be active at these moments. Whenever this BW network tariff model is active, consumers will pay their normal network tariff (assumed to be 0 in this study) up to a pre-defined maximal capacity ("the bandwidth"). If they use additional capacity above the BW, a pre-defined higher tariff will be charged for the amount of power that was used above the BW limit. The DSO will communicate which time periods the BW tariff will be active for the next day. We assume that the HEMSs optimize monetary value in terms of DA-market revenue/costs and the extra costs due to the BW network tariff. We consider the HEMSs to have perfect knowledge of the future DA prices and weather conditions. We furthermore assume that the representations the HEMSs use for the assets in its optimization perfectly model the asset behavior. The horizon of the rolling optimization is chosen to be $H = 12$ hours. Only for the prognosis of the day-ahead schedule the horizon is 36 hours. The resulting Mixed-Integer Linear Program (MILP) solved by the HEMSs at every time step is then:

$$\min_{\substack{P_{t,\mathrm{PV}},P^{\mathrm{ch}}_{t,\mathrm{BESS}},P^{\mathrm{disch}}_{t,\mathrm{BESS}} \\ ,z^{\mathrm{ch}}_{t,\mathrm{BESS}}Q^{\mathrm{tank}}_{t,\mathrm{HP}},Q^{\mathrm{house}}_{t,\mathrm{HP}},z_{t,\mathrm{HP}}}} \quad \sum_{t=0}^{H} \pi_{t,\mathrm{DA}}(E_{t,\mathrm{buy}} - E_{t,\mathrm{sell}}) + \Pi_{t,\mathrm{BW}}, \tag{1}$$

where the index t denotes time, E denotes energy for buying/selling in a PTU, $\pi_{t,\mathrm{DA}}$ is the DA-price per unit of energy and $\Pi_{t,\mathrm{BW}}$ is the total costs due to the BW model. The continuous decision variables represent the dispatch for the PV $P_{t,\mathrm{PV}}$, dispatches of the BESS $P^{\mathrm{ch}}_{t,\mathrm{BESS}}$, $P^{\mathrm{disch}}_{t,\mathrm{BESS}}$, the heat provided by the HP to the buffer tank $Q^{\mathrm{tank}}_{t,\mathrm{HP}}$, and the heat provided from the tank to the house $Q^{\mathrm{house}}_{t,\mathrm{HP}}$. We have two binary decision variables, $z^{\mathrm{ch}}_{t,\mathrm{BESS}}$ denoting that the battery is charging (1) or discharging (0), and $z_{t,\mathrm{HP}}$ denoting that the heat pump is on (1) or off (0). For BW costs, we have the following constraints:

$$\pi_{\mathrm{BW}}(-(E_{t,\mathrm{buy}} - E_{t,\mathrm{sell}}) - P_{t,\mathrm{BW}}\Delta t) \leq \Pi_{t,\mathrm{BW}} \tag{2}$$

$$0 \leq \Pi_{t,\mathrm{BW}} \tag{3}$$

$$\pi_{t,\mathrm{BW}}((E_{t,\mathrm{buy}} - E_{t,\mathrm{sell}}) - P_{t,\mathrm{BW}}\Delta t) \leq \Pi_{t,\mathrm{BW}}, \tag{4}$$

where $P_{t,\mathrm{BW}}$ and $\pi_{t,\mathrm{BW}}$ denote the BW height and the BW price per unit of energy. The energy balance can now be written as:

$$\begin{aligned} E_{t,\mathrm{buy}} = {} & (P_{t,\mathrm{base}} + P^{\mathrm{ch}}_{t,\mathrm{BESS}} + P_{t,\mathrm{HP}})\Delta t \\ & - E_{t,\mathrm{PV,use}} - E_{t,\mathrm{BESS,use}} \end{aligned} \tag{5}$$

$$E_{t,\mathrm{sell}} = E_{t,\mathrm{PV,sell}} + E_{t,\mathrm{BESS,sell}} \tag{6}$$

$$E_{t,\mathrm{PV,use}} + E_{t,\mathrm{PV,sell}} \leq P_{t,\mathrm{PV}}\Delta t \tag{7}$$

$$E_{t,\mathrm{BESS,use}} + E_{t,\mathrm{BESS,sell}} = \eta^{\mathrm{disch}} P^{\mathrm{disch}}_{t,\mathrm{BESS}}\Delta t, \tag{8}$$

where we introduced the discharging efficiency of the BESS system η^{disch}. For the PV, we simply assume that its produced power is the product of the global horizontal irradiance, the surface area, and the efficiency of the solar panels. For the BESS, we impose standard constraints for updating the state of charge with a charge efficiency η^{ch}, and conditions for maximum/minimum charge/discharge power and state of charge. We assume that the HP is only used for space heating and that it has a buffer tank. For the thermal models of the houses, we use one of the lumped resistance-capacitance models (RC-models) from the work of Leprince [10]. More information on this model, and on the modeling of the HP in general (the coefficient of performance and operational constraints) we refer to our earlier work [11]. For the sizing of the PV, BESS, and HP systems, random values in a range of typical sizes are selected. Weather data was obtained by linearly interpolating Dutch hourly temperature and irradiance data measured in 2020 to a 15-minute resolution [12], and we used anonymized smart-meter data as baseline profiles $P_{t,\mathrm{base}}$. The resulting MILP is solved using the Coin-or branch and cut solver [13] using a 0.5% tolerance gap for the dispatch optimization and a 5% tolerance gap for creating the DA prognoses. For the load flow calculations in the DSO and LV network agents, we use the python package dss-python, which is a wrapper around the OpenDSS software [14].

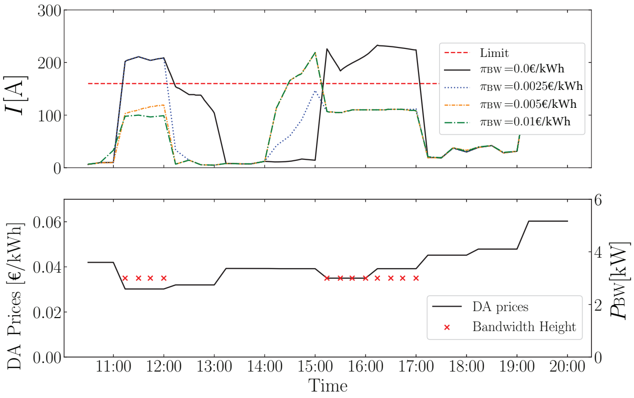

Fig. 3 The effect of different network tariff costs for the dynamic bandwidth on the overloading of one of the main cables of the study network, the bandwidth height when active and corresponding DA prices for the 9th of September 2020.

4 Simulation Results

The computational load of the simulations performed in this study is currently distributed over two Kubernetes nodes, where the Kubernetes cluster is obtained via the Azure Kubernetes Service of Microsoft. For the case study, we use a Dutch LV network, consisting of 2 MV/LV transformers, 74 network nodes, 107 distribution cables, and 37 residential connections. It is assumed that 50% of all residential connections have an HP system installed and 60% of the connections have a PV system installed of which 50% also have a BESS. Whenever congestion problems are expected by the DSO agent, a BW network tariff with a maximal capacity of 3kW will be active at these moments. To access the effectiveness of the dynamic BW network tariff, the pre-defined higher tariff, which will be charged for the amount of power that is used above the BW limit, is varied from a baseline of 0.0, to values of 0.025, 0.005, and 0.01 €/kWh. The simulation is run for a period of three days, using a sliding window and load and weather data for the 8th, 9th, and 10th of September 2020, where the first day is to initialize the data, the second day (September 9th) is the day used to evaluate the dynamic bandwidth tariff, and the third day is used to continue the optimization horizon after the considered study day.

Fig. 3, shows the effect of different network tariff costs for the dynamic bandwidth of 3kW on the overloading of one of the main cables of the study network, as well as the corresponding DA prices for the 9th of September 2020. It can be seen that, in line with expectations, when a higher amount is charged for the amount of power above the BW limit, the dispatch of power during congestion moments is minimized below the BW limit. For lower amounts, the dispatch will be lowered, but not fully minimized depending on the asset state and market prices. For the lowest tested amount of costs, 0.025 €/kWh, the dispatch optimization even neglected the active bandwidth and followed the same dispatch that was seen in the baseline simulation. Furthermore, it can be seen that a dynamic bandwidth network tariff, as applied in this study, is not always able to ensure that congestion problems are resolved. Due to a rebound effect in the dispatch of flexible assets and the influence of market prices, it can also cause a shift in the moment of congestion.

5 Conclusion

This study introduced and demonstrated a new open-source simulation framework for time-discrete distributed agent-based simulations of energy systems. This scalable framework is able to flexibly distribute the computational load of the simulated agents within a simulation and enables distributed development of the simulation models by multiple users. Within this paper, the framework was demonstrated for the analysis of the effect of a dynamic bandwidth network tariff on overloading congestion problems in a Dutch LV network. However, this framework can also be applied to all kinds of energy systems simulations that are far larger and more complex than demonstrated in this paper. In future work, the developed framework will be used to conduct more elaborate research to assess the effectiveness of different combinations of flexibility activation mechanisms, interactions between assets and prosumers in low-voltage grids, DSOs, and electricity markets for 120 Dutch LV networks under various market and weather conditions and various penetrations of flexible assets.

Acknowledgement

The first two authors contributed equally to this work. The authors would like to thank Phuong Nguyen (Eindhoven University of Technology), Johan Morren (Eindhoven University of Technology/Enexis Groep), and Arun Subramanian (Siemens Gamesa/TNO) for the insightful discussions. We furthermore thank the Netherlands Enterprise Agency for partially funding this research through the GO-e project (MOOI32001).

References

[1] GOe-Consortium. Go-e. Accessed on: Jan 22, 2023. [Online]. Available: http://www.projectgo-e.nl/

[2] S. Schütte, S. Scherfke, and M. Tröschel, "Mosaik: A framework for modular simulation of active components in smart grids," in *2011 IEEE First International Workshop on Smart Grid Modeling and Simulation (SGMS)*. IEEE, 2011, pp. 55–60.

[3] D. P. Chassin, K. Schneider, and C. Gerkensmeyer, "Gridlab-d: An open-source power systems modeling and simulation environment," in *2008 IEEE/PES Transmission and Distribution Conference and Exposition*. IEEE, 2008, pp. 1–5.

[4] Anylogic. Anylogic simulation software. Accessed on: Jan 16, 2023. [Online]. Available: https://www.anylogic.com/

[5] H. Çakmak, A. Erdmann, M. Kyesswa, U. Kühnapfel, and V. Hagenmeyer, "A new distributed co-simulation architecture for multi-physics based energy systems integration," *at-Automatisierungstechnik*, vol. 67, no. 11, pp. 972–983, 2019.

[6] L. Gomes, P. Faria, H. Morais, Z. Vale, and C. Ramos, "Distributed, agent-based intelligent system for demand response program simulation in smart grids," *IEEE Intelligent Systems*, vol. 29, no. 1, pp. 56–65, 2013.

[7] M. Lees, B. Logan, and G. Theodoropoulos, "Distributed simulation of agent-based systems with hla," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 17, no. 3, pp. 11–es, 2007.

[8] E. Matthijssen. ESDL documentation. Accessed on: Jan 22, 2023. [Online]. Available: https://energytransition.gitbook.io/esdl/

[9] R. A. Light, "Mosquitto: server and client implementation of the mqtt protocol," *Journal of Open Source Software*, vol. 2, no. 13, p. 265, 2017. [Online]. Available: https://doi.org/10.21105/joss.00265

[10] J. Leprince, "Grey-brick buildings," https://github.com/JulienLeprince/greybrickbuildings, 2022.

[11] G. Verhoeven, B. Van der Holst, and S. C. Doumen, "Modeling a domestic all-electric air-water heat-pump system for discrete-time simulations," in *2022 57th International Universities Power Engineering Conference (UPEC)*. IEEE, 2022, pp. 1–6.

[12] Royal Netherlands Meteorological Institute (KNMI). (2021) Uurgegevens van het weer in nederland. Source in Dutch. [Online]. Available: https://www.knmi.nl/nederland-nu/klimatologie/uurgegevens

[13] J. Forrest and R. Lougee-Heimer, "Cbc user guide," in *Emerging theory, methods, and applications*. INFORMS, 2005, pp. 257–277.

[14] DSS Extensions: multi-platform OpenDSS extensions. DSS-Python: Python interface. Accessed on: Jan 20, 2023. [Online]. Available: https://dss-extensions.org/