SoftwareX 24 (2023) 101580

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

PyECOM: A Python tool for analyzing and simulating Energy Communities

Eduardo Gomes [a,b,c,*], Lucas Pereira [a,b], Augusto Esteves [a,b], Hugo Morais [a,c]

[a] *Instituto Superior Técnico-IST, Universidade de Lisboa, Portugal*
[b] *ITI/LARSyS, Portugal*
[c] *INESC-ID, Portugal*

ARTICLE INFO

Keywords:
Energy Communities
Optimization

ABSTRACT

Python Energy Communities (PyECOM) is a tool designed to facilitate the research into Energy Communities (EC) through the ability to create, simulate and test various EC settings. Key components of EC functioning are implemented and provide a flexible basis for the integration of new components. Optimization algorithms from libraries such as Pymoo can be accessed and are facilitated through the use of the proposed data model. In future versions, PyECOM will also feature Reinforcement Learning as a method available to the users.

Code metadata

Current code version	1.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00547
Permanent link to Reproducible Capsule	https://zenodo.org/record/8395454
Legal Code License	MIT License
Code versioning system used	Git
Software code languages, tools, and services used	Python
Compilation requirements, operating environments & dependencies	requirements.txt available
If available Link to developer documentation/manual	https://ecgomes.github.io/pyecom/ (doxygen generated)
Support email for questions	eduardo.c.gomes@tecnico.ulisboa.pt

1. Motivation and significance

The integration of Renewable Energy Sources (RES), coupled with the emergence of Energy Communities (ECs), has brought new challenges to the way resources are used [1,2]. Scheduling of energy resources on ECs can be challenging due to the various technologies now integrated with it [3]. Along with technological developments, new interactions with the grid and business models are now considered [4]. The latter requires systems that can correctly estimate resource availability and how much energy will be needed for executing transactions [5,6].

Previous research in this field has tackled the problem of resource utilization in ECs, from focusing on single assets of a community such as storage [7,8] to Electric Vehicles (EVs) [9,10]. Several works have also tackled the optimization of ECs as a whole [11–13].

Optimization of ECs resources requires an understanding of how to make the best use of all of them, considering their technical characteristics. To this end, different strategies have been proposed, including Linear Programming (LP) [14], Mixed-Integer Linear Programming (MILP) [15,16], Metaheuristics (MH) [12,17,18] and Machine Learning (ML) approaches [19].

However, designing an EC an optimizing its operation is not a straightforward process, requiring code proficiency [20,21], which creates barriers to developing and deploying solutions. This work aims to offer a tool to facilitate the implementation pipeline and provide a user-friendly end-to-end framework. Our framework circumvents the high-level coding barriers required for solution handling and provides simple commands that are easily interpreted and implemented.

To this end, this paper proposes Python Energy Community (PyECOM), a Python package that attempts to streamline the entirety of the process of optimizing an EC, from the data loading to the extraction and analysis of the optimization results. PyECOM allows loading EC scenarios built in multiple formats and includes a data engineering component to prepare the data for usage and a constraint-based solution-fixing mechanism intended for metaheuristics

* Corresponding author at: Instituto Superior Técnico-IST, Universidade de Lisboa, Portugal.
E-mail address: eduardo.c.gomes@tecnico.ulisboa.pt (Eduardo Gomes).

https://doi.org/10.1016/j.softx.2023.101580
Received 14 August 2023; Received in revised form 5 November 2023; Accepted 6 November 2023
2352-7110/

Table 1
Available Metrics.

Metric	Description
Self Consumption	Measure of production consumed internally.
Total Produced	Sum of total produced power.
Import and Export Balance	Indication of the import and export required.

and neural network generated values. PyECOM is available on GitHub as an open-source repository in [22].

The remainder of this paper is organized as follows: Section 2 provides an in-depth overview of the tool and the functionality provided, presenting a visual representation of the pipeline and how the PyECOM was constructed. Section 3 provides a use case of the proposed software, with examples of usage, including a benchmark between two well-established MHs. Section 4 discusses the proposed work's potential impact and implications, including the comparison with previously proposed tools. Finally, Section 5 provides some remarks on PyECOM and future research directions.

2. Software description

PyECOM is a Python library that offers a unified approach to studying Energy Communities, providing users with data processing, optimization, and result analysis capabilities. It consolidates the management of ECs by utilizing a data model that streamlines the integration of components. The proposed software interfaces with other libraries such, as Pymoo [23], making use of its implemented metaheuristics. PyECOM also incorporates a result analysis component, available through the implementation of relevant metrics in the context of ECs, as listed in Table 1.

2.1. Software architecture

PyECOM is comprised by seven main components: algorithms, metrics, parsers, repairs, resources, markets, and scenes. Each of these components was defined using interfaces to maximize code reuse. By using said components, an Energy Community test scenario can be generated with the desired inputs, algorithms and evaluated according to an user defined metrics.

Data Parsing — PyECOM provides an abstract BaseParser class that can be extended to support multiple data formats and inputs. The parser is responsible for of accurately reading necessary data and creating the appropriate resources for the correct functioning of the system.

Data Model — The input/parsed data is preserved in the form of a Resource, storing the information required for the functioning of the community, such as cost values and forecasts. Depending on the resource analyzed, a typical resource used for optimization purposes possesses a value and its cost (can be over a time period), and a lower and upper bound accepted. Basic operations for the resource instances were defined, such as sum and multiplication, as to facilitate the implementation process. Should a resource require additional information, a class can be created that extends the existing BaseResource or another class that extends from it. Taking advantage of this formulation, PyECOM provides a subclass for some resources that contemplate the use of probabilistic methods, accepting lower, and upper confidence bounds.

Optimization — The algorithmic component of the tool, especially the metaheuristics (MH), relies on the Pymoo package [23], which implements several MH algorithms and allows custom solution fixing. The definition of custom algorithms is also supported, requiring user and coding expertise. The provided BaseOptimizer provides the main directives for optimization algorithms, while its extensions aim to facilitate the use of several algorithms such as Artificial Neural Networks (OptimizerNeuralNet) and mentioned MH (OptimizerMetaheuristic).

Markets — A compilation of entities that make up the interactions of a market. It is, at a base level, comprised of items, participants, transactions, transaction handler, pricing system, and the market itself. When defining a market, it is required to define the pricing system, responsible for managing the evolution of participant and item interactions, as well as the transaction handling system, used for certifying transaction viability. This part of PyECOM allows resource trading between agents of a simulation.

Analysis — PyECOM is capable of providing a result analysis component. With user inputs, it is possible to analyze particular aspects of the specified EC, such as self-consumption, and self-sufficiency. The CommunityMetrics class implements the metrics listed in Table 1, while BaseMetric defines the metric general behavior and facilitates function calls.

```python
#  Imports
from src.parsers import HMParser
from src.repairs import HMRepairPymoo
from src.scenes import HMProblemPymoo
from src.metrics import CommunityMetrics

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# Ignore the warnings (only demonstration purposes!)
import warnings
warnings.filterwarnings("ignore")

# Data parsing
data = HMParser(file_path='data/EC_V4.xlsx', ec_id=1)
data.parse()

# Pymoo imports for using DE
from pymoo.algorithms.soo.nonconvex.de import DE
from pymoo.operators.sampling.lhs import LHS
from pymoo.optimize import minimize
from pymoo.termination.default import DefaultSingleObjectiveTermination

# Instantiate the DE algorithm
algorithm = DE(pop_size=30,
               sampling=LHS(),
               variant="DE/rand/1/bin",
               CR=0.3,
               dither="vector",
               jitter=False,
               repair=HMRepairPymoo(data=data))


# Set the termination criteria
termination = DefaultSingleObjectiveTermination(xtol=1e-8,
                              cvtol=1e-6, ftol=1e-6, period=1000,
                              n_max_gen=2e3, n_max_evals=5e9)

# Instantiate the problem
customProblem = HMProblemPymoo(data=data)

# Run the optimization
temp_res = minimize(customProblem,
                    algorithm,
                    termination=termination,
                    save_history=True,
```

```
verbose=True)
```

Listing 1: Pymoo integration example

Scene — A Scene plays the role of an aggregator, where all the parts of the package come together to form a pipeline. The Scenes are at least comprised by the components of the Community and should handle the calls to the individual parts. An example of a Scene implementation can be idealized as loading and parsing the data to resources, passing the resources to an algorithm, and analyzing the results with proposed metrics, making use of the established directives of the `BaseScene` class.

2.2. Software functionalities

`PyECOM` facilitates the study of Energy Communities, providing a centralized end-to-end pipeline that incorporates data loading and organization, optimization and result analysis. Through the proposed data model, `PyECOM` aims to facilitate resource integration, providing implementations for resources such as generators, loads, storage solutions and electric vehicles.

`PyECOM` provides a parser implemented for an example Excel file that contains information of a sample EC. Said file is composed of nine spreadsheets — general information, network, peers, load profiles, generator profiles, available storage, electric vehicles, and additional generator type and contract details. General Information contains information regarding the simulation settings, such as Simulation Periods, Periods duration (min). Additionally, it contains information on maximum import and exports, as well as the grid's energy buy and sell prices. The Network Information sheet contains data pertaining to the physical grid and connections among the several resources, and was not considered in this stage. Peers' sheet holds data regarding the peers participating on the simulation, presenting information on the buy and sell prices of the individual resources. The Load and Generator sheets contain information on the forecasting values of the components, upper limits, owner and type of contract. Storage and Charging Station sheets provide specifications of battery solutions and charging stations used in the simulation, including maximum capacity, charge and discharge limits and efficiencies, complemented by the costs of charge and discharge costs. Finally, the Electric Vehicles' sheet considers information on the scheduling of each EV use, as well as information regarding each EV battery. New users can use this file as a starting point reference, coupled with its dedicated README page that contains description, examples on how to add resources and events, and current limitations.

To facilitate the process of studying ECs, a parser for `PROCSIM` was also implemented. `PROCSIM` is a `Python` package that allows the creation of Energy Community datasets, more specifically demand and generation at applicance-level granularity [21]. It was developed with the aim of providing a simple and reliable way of creating datasets that can be used for optimization and scheduling, as well as testing control strategies on particular resources such as energy storage system technologies.

`PyECOM` provides an implementation of the HyDE-DF algorithm, as per [24]. We have opted against the implementation of optimization functions, as these should be written considering the problem at hand and the objectives of the user.

3. Illustrative example

This section provides an example use of `PyECOM`, for benchmarking two single-objective metaheuristics readily available in `Pymoo`: Differential Evolution (DE) [25], and Genetic Algorithm (GA) [26]. The Excel data source used in this example can be found in the tool's repository.

The Energy Community considered in this example has information over a 24-hour period at an hourly resolution, composed of Generators, Loads, Profiles, Storage options, Charging Stations, main grid imports/exports, and Electric Vehicles. Complementary information is also included in the Excel file, such as indicators of whether the Generators are renewable or not, vehicle trip schedules and load types. The code for the optimization process can be found in Listing 1. The purposed of the presented code is the generation of an optimal resource usage, through the minimization of community's costs, defined in the `HMProblemPymoo` class. having as purpose generating an optimal 24 h schedule of resource usage.

For completion, we provide a visualization of the results achieved by both algorithms over 2000 iterations, using the package `matplotlib`. Fig. 1 allows the visualization of the solution provided, separated in terms of consumption and production, to better distinguish the impact of each individual component. No modifications were made to the algorithms used.

4. Impact

Similarly to the various definitions of an Energy Community in the literature [27], there are also multiple approaches to tackle the optimization of ECs resources [20]. This can be extended to the many research directions related to ECs, such as proposing and exploring algorithms [28] and policymaking [27,29,30]. Although EC software implementations have emerged in other related works [31,32], and other related areas such as energy in buildings [33] and grid [34], the overall landscape of the energy sector suffers from the availability of open-source tools [35,36].

`PyECOM` aims to provide an open-source, centralized and unified approach to Energy Community analysis in a way that users – i.e. researchers and system planners – can interact with and facilitate research and deployment of solutions [20,35]. One of `PyECOM`'s main contribution lies in its facilitation of research and solution deployment within the EC domain. Through its design and functionality implementations, `PyECOM` allows researchers and system planners to address interactions with growing energy data. By streamlining the process, it not only offers enhanced efficiency to proficient developers, but also facilitates the interaction to less experienced audiences. Example notebooks can be found in the tool's repository, providing sample code for running scheduling tasks, and market simulations. An early version of this tool was utilized for benchmarking of HyDE-DF against a MILP baseline, on a similar use case [37].

Furthermore, by presenting a standardized methodology for EC analysis, `PyECOM` addresses a crucial concern in the realm of reproducibility and transparency, promoting consistent practices across the field. This unified approach not only ensures that results can be easily reproduced, but also facilitates the comparison of different approaches, and allow sharing of methodologies among the research community. The extensive design of `PyECOM` positions it as a tool capable of accommodating emerging algorithms and strategies. The integration of Reinforcement Learning as an optimization method is planned for future updates, further solidifying `PyECOM`'s capabilities and potential adoption.

5. Conclusions

This work presented `PyECOM`, a tool intended to streamline and standardize the process of creating, deploying, and evaluating Energy Community operations. Our tool was designed to be flexible, allow users to design Energy Communities in the desired configuration, and study the behavior of different optimization strategies facilitated by the use of provided metrics. Furthermore, some implementations, such as HyDE-DF, are already part of the tool and can be used out of the box. `PyECOM` can be further optimized and extended to suit the needs of the user. Reinforcement Learning approaches are currently being implemented, taking advantage of the established data modeling. More methods and algorithms will be considered in future versions, furthering the capabilities of the proposed tool.

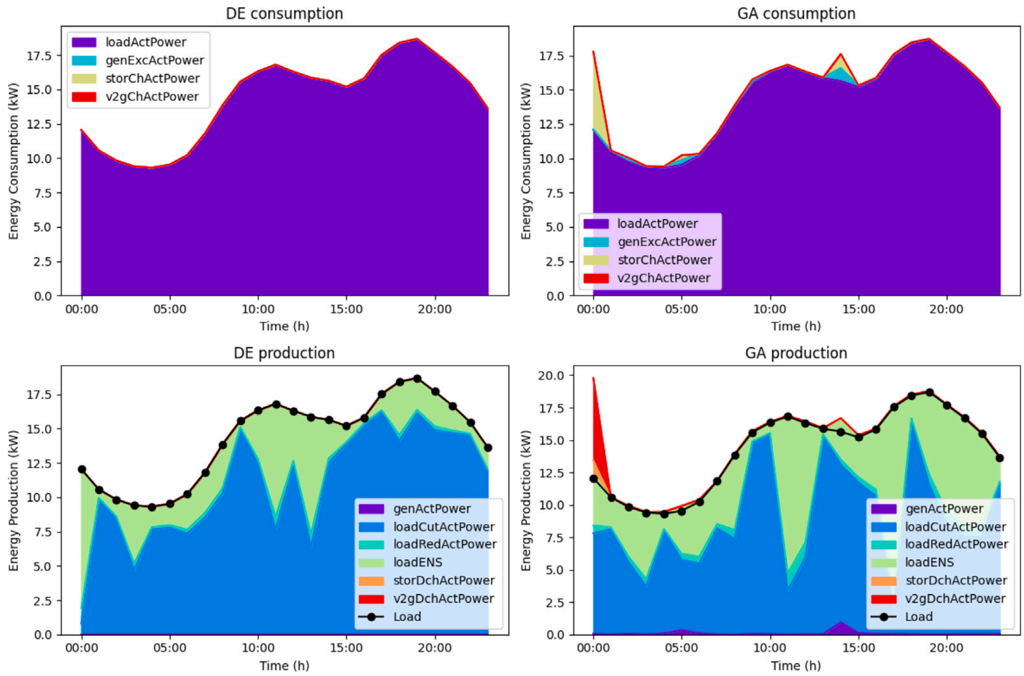

Fig. 1. Consumption and Production groups of benchmarked algorithms.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Excel file included in the compressed file.

Acknowledgments

This work received funding from the European Union's Horizon Europe research and innovation programme under grant agreement no. 101056765. Views and opinions expressed in this document are however those of the authors only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the grating authority can be held responsible for them. The authors received funding from the Portuguese Foundation for Science and Technology (FCT) under grant 2021.07754.BD (E.G.), CEECIND/01179/2017 (L.P.), UIDB/50009/2020 (L.P., A.E., and E.G.) and UIDB/50021/2020 (H.M.).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.softx.2023.101580.

References

[1] Caramizaru E, Uihlein A. Energy communities: an overview of energy and social innovation. In: Scientific analysis or review, Policy assessment KJ-NA-30083-EN-N (online). Luxembourg: Publications Office of the European Union; 2020, http://dx.doi.org/10.2760/180576(online).

[2] Gjorgievski VZ, Cundeva S, Georghiou GE. Social arrangements, technical designs and impacts of energy communities: A review. Renew Energy 2021;169:1138–56. http://dx.doi.org/10.1016/j.renene.2021.01.078, URL https://www.sciencedirect.com/science/article/pii/S0960148121000859.

[3] Ahmed SD, Al-Ismail FSM, Shafiullah M, Al-Sulaiman FA, El-Amin IM. Grid integration challenges of wind energy: a review. IEEE Access 2020;8:10857–78. http://dx.doi.org/10.1109/ACCESS.2020.2964896, Conference Name: IEEE Access.

[4] Reis I, Gonçalves I, Lopes M, Henggeler Antunes C. Business models for energy communities: A review of key issues and trends. Renew Sustain Energy Rev 2021;144:111013. http://dx.doi.org/10.1016/j.rser.2021.111013, URL https://www.sciencedirect.com/science/article/pii/S1364032121003038.

[5] Sousa T, Soares T, Pinson P, Moret F, Baroche T, Sorin E. Peer-to-peer and community-based markets: A comprehensive review. Renew Sustain Energy Rev 2019;104:367–78. http://dx.doi.org/10.1016/j.rser.2019.01.036, URL http://www.sciencedirect.com/science/article/pii/S1364032119300462.

[6] Gomes L, Morais H, Gonçalves C, Gomes E, Pereira L, Vale Z. Impact of forecasting models errors in a peer-to-peer energy sharing market. Energies 2022;15(10):3543. http://dx.doi.org/10.3390/en15103543, Number: 10 Publisher: Multidisciplinary Digital Publishing Institute, URL https://www.mdpi.com/1996-1073/15/10/3543.

[7] Koirala BP, van Oost E, van der Windt H. Community energy storage: A responsible innovation towards a sustainable energy system? Appl Energy 2018;231:570–85. http://dx.doi.org/10.1016/j.apenergy.2018.09.163, URL https://www.sciencedirect.com/science/article/pii/S0306261918314740.

[8] Manso-Burgos A, Ribó-Pérez D, Gómez-Navarro T, Alcázar-Ortega M. Local energy communities modelling and optimisation considering storage, demand configuration and sharing strategies: A case study in Valencia (Spain). Energy Rep 2022;8:10395–408. http://dx.doi.org/10.1016/j.egyr.2022.08.181, URL https://www.sciencedirect.com/science/article/pii/S2352484722016249.

[9] Ma W, Hu J, Yao L, Fu Z, Morais H, Marinelli M. New technologies for optimal scheduling of electric vehicles in renewable energy-oriented power systems: A review of deep learning, deep reinforcement learning and blockchain technology. Energy Convers Econ 2022;3(6):345–59. http://dx.doi.org/10.1049/enc2.12071, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1049/enc2.12071, URL https://onlinelibrary.wiley.com/doi/abs/10.1049/enc2.12071.

[10] Morais H. New approach for electric vehicles charging management in parking lots considering fairness rules. Electr Power Syst Res 2023;217:109107. http://dx.doi.org/10.1016/j.epsr.2022.109107, URL https://www.sciencedirect.com/science/article/pii/S0378779622011567.

[11] Denysiuk R, Lilliu F, Recupero D, Vinyals M. Peer-to-peer energy trading for smart energy communities:. In: Proceedings of the 12th international conference on agents and artificial intelligence. Valletta, Malta: SCITEPRESS - Science and Technology Publications; 2020, p. 40–9. http://dx.doi.org/10.5220/0008915400400049, URL http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0008915400400049.

[12] Papadimitrakis M, Giamarelos N, Stogiannos M, Zois EN, Livanos NAI, Alexandridis A. Metaheuristic search in smart grid: A review with emphasis on planning, scheduling and power flow optimization applications. Renew Sustain Energy Rev 2021;145:111072. http://dx.doi.org/10.1016/j.rser.2021.111072, URL https://www.sciencedirect.com/science/article/pii/S1364032121003609.

[13] Faia R, Soares J, Pinto T, Lezama F, Vale Z, Corchado JM. Optimal model for local energy community scheduling considering peer to peer electricity transactions. IEEE Access 2021;9:12420–30. http://dx.doi.org/10.1109/ACCESS.2021.3051004, Conference Name: IEEE Access.
[14] Rabe M, Bilan Y, Widera K, Vasa L. Application of the linear programming method in the construction of a mathematical model of optimization distributed energy. Energies 2022;15(5):1872. http://dx.doi.org/10.3390/en15051872, Number: 5 Publisher: Multidisciplinary Digital Publishing Institute, URL https://www.mdpi.com/1996-1073/15/5/1872.
[15] Volpato G, Carraro G, Cont M, Danieli P, Rech S, Lazzaretto A. General guidelines for the optimal economic aggregation of prosumers in energy communities. Energy 2022;258:124800. http://dx.doi.org/10.1016/j.energy.2022.124800, URL https://www.sciencedirect.com/science/article/pii/S0360544222017030.
[16] Schiera DS, De Vizia C, Zarri A, Borchiellini R, Lanzini A, Patti E, et al. Modelling and techno-economic analysis of Peer-to-Peer electricity trading systems in the context of Energy Communities. In: 2022 IEEE international conference on environment and electrical engineering and 2022 IEEE industrial and commercial power systems Europe (EEEIC/I&CPS Europe). 2022, p. 1–6. http://dx.doi.org/10.1109/EEEIC/ICPSEurope54979.2022.9854537.
[17] Mohseni S, Khalid R, Brent AC. Metaheuristic-based isolated microgrid sizing and uncertainty quantification considering EVs as shiftable loads. Energy Rep 2022;8:11288–308. http://dx.doi.org/10.1016/j.egyr.2022.08.254, URL https://www.sciencedirect.com/science/article/pii/S2352484722016985.
[18] Fathy A, Alanazi TM, Rezk H, Yousri D. Optimal energy management of micro-grid using sparrow search algorithm. Energy Rep 2022;8:758–73. http://dx.doi.org/10.1016/j.egyr.2021.12.022, URL https://www.sciencedirect.com/science/article/pii/S2352484721014608.
[19] Ahmad T, Madonski R, Zhang D, Huang C, Mujeeb A. Data-driven probabilistic machine learning in sustainable smart energy/smart energy systems: Key developments, challenges, and future research opportunities in the context of smart grid paradigm. Renew Sustain Energy Rev 2022;160:112128. http://dx.doi.org/10.1016/j.rser.2022.112128, URL https://www.sciencedirect.com/science/article/pii/S1364032122000569.
[20] Kazmi H, Munné-Collado I, Mehmood F, Syed TA, Driesen J. Towards data-driven energy communities: A review of open-source datasets, models and tools. Renew Sustain Energy Rev 2021;148:111290. http://dx.doi.org/10.1016/j.rser.2021.111290, URL https://www.sciencedirect.com/science/article/pii/S1364032121005773.
[21] Velosa N, Gomes E, Morais H, Pereira L. PROCSIM: an open-source simulator to generate energy community power demand and generation scenarios. Energies 2023;16(4):1611. http://dx.doi.org/10.3390/en16041611, Number: 4 Publisher: Multidisciplinary Digital Publishing Institute, URL https://www.mdpi.com/1996-1073/16/4/1611.
[22] ECGomes/pyecom, URL https://github.com/ECGomes/pyecom.
[23] Blank J, Deb K. pymoo: multi-objective optimization in python. IEEE Access 2020;8:89497–509.
[24] Lezama F, Soares J, Faia R, Vale Z. Hybrid-adaptive differential evolution with decay function (HyDE-DF) applied to the 100-digit challenge competition on single objective numerical optimization. In: Proceedings of the genetic and evolutionary computation conference companion. New York, NY, USA: Association for Computing Machinery; 2019, p. 7–8. http://dx.doi.org/10.1145/3319619.3326747.
[25] Storn R, Price K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 1997;11(4):341–59. http://dx.doi.org/10.1023/A:1008202821328.
[26] Holland JH. Genetic algorithms. Sci Am 1992;267(1):66–73, Publisher: Scientific American, a division of Nature America, Inc., URL https://www.jstor.org/stable/24939139.
[27] de São José D, Faria P, Vale Z. Smart energy community: A systematic review with metanalysis. Energy Strategy Rev 2021;36:100678. http://dx.doi.org/10.1016/j.esr.2021.100678, URL https://www.sciencedirect.com/science/article/pii/S2211467X2100064X.
[28] Lazzari F, Mor G, Cipriano J, Solsona F, Chemisana D, Guericke D. Optimizing planning and operation of renewable energy communities with genetic algorithms. Appl Energy 2023;338:120906. http://dx.doi.org/10.1016/j.apenergy.2023.120906, URL https://www.sciencedirect.com/science/article/pii/S0306261923002702.
[29] Boulanger SOM, Massari M, Longo D, Turillazzi B, Nucci CA. Designing collaborative energy communities: a European overview. Energies 2021;14(24):8226. http://dx.doi.org/10.3390/en14248226, URL https://www.mdpi.com/1996-1073/14/24/8226.
[30] Fernandes DV, Silva CS. Open Energy Data — A regulatory framework proposal under the Portuguese electric system context. Energy Policy 2022;170:113240. http://dx.doi.org/10.1016/j.enpol.2022.113240, URL https://www.sciencedirect.com/science/article/pii/S0301421522004591.
[31] El Kontar R, Polly B, Charan T, Fleming K, Moore N, Long N, et al. URBANopt: an open-source software development kit for community and Urban district energy modeling: preprint. Tech. rep. NREL/CP-5500-76781, Golden, CO (United States): National Renewable Energy Lab. (NREL); 2020, URL https://www.osti.gov/biblio/1677416.
[32] Lazzeroni P, Lorenti G, Moraglio F, Repetto M. Modeling of renewable energy communities: the RECoupled approach. In: 2022 IEEE 46th annual computers, software, and applications conference. 2022, p. 1349–54. http://dx.doi.org/10.1109/COMPSAC54236.2022.00213.
[33] Vázquez-Canteli JR, Kämpf J, Henze G, Nagy Z. CityLearn v1.0: an OpenAI gym environment for demand response with deep reinforcement learning. In: Proceedings of the 6th ACM international conference on systems for energy-efficient buildings, cities, and transportation. New York, NY, USA: Association for Computing Machinery; 2019, p. 356–7. http://dx.doi.org/10.1145/3360322.3360998, URL https://dl.acm.org/doi/10.1145/3360322.3360998.
[34] Pigott A, Crozier C, Baker K, Nagy Z. GridLearn: multiagent reinforcement learning for grid-aware building energy management. 2021, http://dx.doi.org/10.48550/arXiv.2110.06396, URL http://arxiv.org/abs/2110.06396.
[35] Pfenninger S, DeCarolis J, Hirth L, Quoilin S, Staffell I. The importance of open data and software: Is energy research lagging behind? Energy Policy 2017;101:211–5. http://dx.doi.org/10.1016/j.enpol.2016.11.046, URL https://www.sciencedirect.com/science/article/pii/S0301421516306516.
[36] Nicoli M, Gracceva F, Lerede D, Savoldi L. Can we rely on open-source energy system optimization models? The TEMOA-Italy case study. Energies 2022;15(18):6505. http://dx.doi.org/10.3390/en15186505, URL https://www.mdpi.com/1996-1073/15/18/6505.
[37] Gomes E, Pereira L, Morais H. Energy resources scheduling in energy communities: A comparison between mixed integer linear programming and hybrid-adaptive differential evolution with decay function. [in press] in 2023 IEEE PES Innovative Smart Grid Technologies Europe (ISGT EUROPE).