PyPSA-PL

Version 2.0. User guide

Patryk Kubiczek (patryk.kubiczek@instrat.pl)

2023-03-28

instrat

Table of Contents

1 Introduction **1**

2 Model inputs **2**
2.1 Run parameters 2
2.2 Input data 6

3 Model outputs **12**
3.1 Raw PyPSA inputs and outputs 12
3.2 Output statistics 12

4 Selected methodological details **12**
4.1 Dispatch optimisation 12
4.2 Investment optimisation 13
4.3 Rescaling of the wind power utilisation factors 14

5 Annex: flowchart representation of the optimisation pipeline **14**

6 Changelog **15**

7 References **16**

1 Introduction

PyPSA-PL is an implementation of the PyPSA framework (Brown, Hörsch and Schlachtberger, 2018) shipped with a use-ready dataset tailored for the Polish energy system. PyPSA-PL can be used to plan optimal investments in the energy sector – given the energy demand together with capital and operation costs for assets – or just to optimise the hourly dispatch of the utility units – given the energy demand and operation costs only. That makes it a useful tool to investigate the feasibility of decarbonisation scenarios for the Polish energy system in which a large share of energy will come from intermittent sources like wind and solar.

As of now (PyPSA-PL version 2.0) only the electricity sector is represented. The components of the PyPSA-PL model are presented in Figure 1. All (or a subset) of those components are connected to each electricity bus of the model (i.e. grid node). The electricity flows and energy conversions between the components are modelled with high temporal resolution under the cost-optimality assumption.

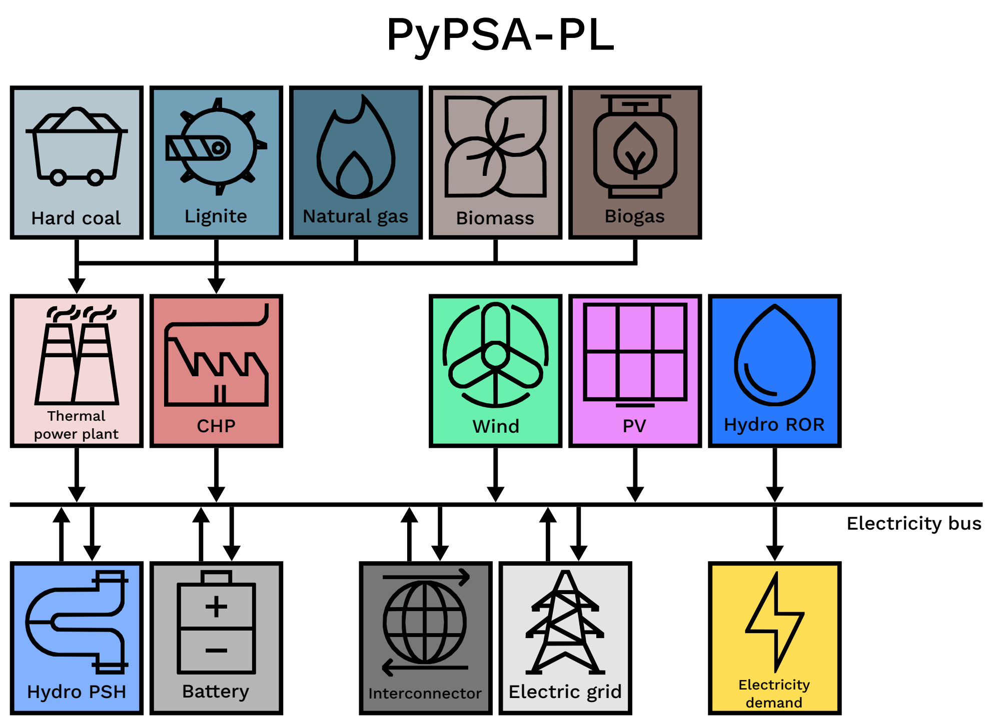

Figure 1: Energy flows and conversions between PyPSA-PL components

2 Model inputs

2.1 Run parameters

The parameters that specify each model run are listed in Table 1. All parameters are optional – in case the parameter is not specified, the default value is used.

Table 1: Parameters specifying a model run

Parameter	**Meaning**	**Default value**
`mode`	mode of PyPSA simulation (`lopf` – linear power flow optimisation)	`lopf`
`solver`	solver to be used internally by PyPSA (e.g. `highs`, `cplex`, `gurobi`)	`highs`
`scenario`	user-selected scenario name – can be used to conveniently refer to the whole set of parameter selections	`pypsa_pl_v2`
`years`	years to be simulated	`[2020]`
`extendable_technologies`	list of technologies whose capacities should be optimised or `None` for pure dispatch optimisation	`None`
`extend_from_zero`	for extendable technologies, ignore the capacities defined by `capacity_investments` dataset?	`False`
`combustion_units`	source of thermal power plant data	`energy.instrat.pl`
`renewable_units`	source of renewable power plant data	`pypsa_pl_v2`
`storage_units`	source of energy storage unit data	`pypsa_pl_v2`
`decommission_year_inclusive`	is the power plant active in the decommissioning year?	`True`
`capacity_investments`	source of data on existing and planned capacity investments per technology and voivodeship	`pypsa_pl_v2_copper_plate`
`virtual_dsr`	add virtual DSR capacity to avoid potential solution infeasibilities?	`True`
`capacity_potentials`	source of data on capacity potentials per technology and voivodeship	`instrat_2021`
`capacity_max_growth`	source of data on maximum capacity growth assumptions	`instrat_2022`

Parameter	Meaning	Default value
renewable_utilization_profiles	source of data on maximum hourly utilisation of wind and solar energy	PECD3+EMHIRES
correction_factor_wind_old	rescaling the hourly wind power utilisation for turbines built until 2020	0.91
correction_factor_wind_new	rescaling the hourly wind power utilisation for turbines built after 2020	1.09
hydro_utilization	source of data on annual utilisation of hydropower	entsoe_2020
chp_utilization_profiles	source of data on hourly utilisation of CHP plants	regression
enforce_bio	minimum annual capacity utilisation factor for biomass and biogas power plants (expressed as a share of maximum technically available capacity factor)	0
industrial_utilization	fixed hourly utilisation of industrial power plants	0.5
demand	source of annual demand data	PSE_2022
demand_correction	rescaling annual demand	1
load_profile	source of hourly demand profile data	entsoe
load_profile_year	historical year used to create hourly load and capacity utilisation profiles (2012, 2013, or 2015)	2012
temporal_resolution	length of a single timestep	1H
nodes	source of grid node data	pypsa_pl_v1_75
grid_resolution	spatial resolution of the electricity grid	copper_plate
links	source of cross-border flow capacity data	pypsa_pl_v2
trade_factor	rescaling cross-border flow capacities	1
imports	allow electricity imports?	True
exports	allow electricity exports?	True

Parameter	Meaning	Default value
`dynamic_trade_prices`	simulate neighbours' energy systems dynamically?	`True`
`neighbors_capacity_demand`	source of installed capacity and demand data for neighbouring countries (only if `dynamic_trade_prices` is `True`)	`TYNDP_2022`
`neighbors_load_profile`	source of hourly demand profile data for neighbouring countries (only if `dynamic_trade_prices` is `True`)	`PECD3`
`trade_prices`	source of static import/export price assumption data (only if `dynamic_trade_prices` is `False`)	`pypsa_pl_v1`
`technology_data`	source of technology data	`instrat_2023`
`discount rate`	discount rate used for annuitisation of capital costs	`0.03`
`prices`	source of price assumption data	`instrat_2023`
`srmc_wind`	marginal cost of wind energy (PLN/MWh_e) – overrides value from `technology_data` dataset	`8`
`srmc_pv`	marginal cost of solar energy (PLN/MWh_e) – overrides value from `technology_data` dataset	`1`
`srmc_dsr`	marginal cost of electricity demand reduction (PLN/MWh_e)	`1200`
`random_seed`	random seed used for adding random noise from range [-1, 1] to marginal costs or `None` if no randomisation is desired	`0`
`reserve_margin`	required reserve capacity (expressed as share of hourly load)	`0.09`
`reserve_technologies`	list of technologies that can be used to provide reserve capacity	`["JWCD", "Hydro PSH", "Battery large", "Battery large 1h", "Battery large 4h"]`

Parameter	Meaning	Default value
max_r_over_p	maximum reserve to active power ratio for thermal power plants (used to enforce running reserves)	1

⚠ Only copper plate spatial resolution is supported now

A voivodeship-level spatial resolution of the electricity grid might be implemented in the future.

Total installed capacity

The total installed capacity in a simulation is an aggregation of data defined by combustion_units, renewable_units, storage_units, and capacity_investments, together with optimal investments for technologies listed in extendable_technologies.

Weather years

It is important to preserve all the realistic correlations between load and weather time series in each simulation. For that purpose, we select a single historical year as a basis to create all the hourly profiles that are used to drive the model. In the context of the utilization of variable renewable energy sources, we identify 2012 as typical, 2013 as bad, and 2015 as a good weather year. The choice of the weather year is done by setting the load_profile_year parameter.

Marginal costs determine curtailment order

From the perspective of system cost optimisation, marginal costs for PV and wind are negligible. However, the parameters srmc_pv and srmc_wind can be used to determine the preferred curtailment order – higher marginal cost for wind turbines means their output will be curtailed first.

2.2 Input data

Input data are stored in the data/input directory and are organised in several *.xlsx spreadsheets. Users can substitute any of the default inputs with their own data by setting the corresponding run parameter (shown in the following in curly braces) to the source keyword of their custom input filename.

2.2.1 Existing and planned utility units

Files:

- `combustion_units;source={combustion_units}.xlsx`
- `renewable_units;source={renewable_units}.xlsx`
- `storage_units;source={storage_units}.xlsx`

Those spreadsheets contain the data on individual utility units, specifying their properties including commissioning and decommissioning years.

Spreadsheet structure:

- rows: individual units
- columns: unit properties
- sheets: unit categories/technologies

2.2.2 Existing and planned aggregate capacity investments

File: `capacity_investments;source={capacity_investments}.xlsx`

The remaining utility units are aggregated at the national or voivodeship level and are attributed to an investment year (more precisely, an investment period; investment year Y encompasses a 5-year period $[Y$-4, $Y]$). Only capacities defined in this file can be subject to investment optimisations.

It is also possible to specify the installed capacity at a given year rather than derive the capacity from cumulative investments, which is useful for capacities that are never subject to investment optimisation (e.g. industrial power plants).

Spreadsheet structure:

- rows: voivodeships, type (investment or installed capacity)
- columns: capacities per year (in MW)
- sheets: categories/technologies

2.2.3 Capacities in neighbouring countries

File: `neighbors_capacities;source={neighbors_capacity_demand}.xlsx`

This spreadsheet contains the installed capacities of neighbouring countries.

Spreadsheet structure:

- rows: technologies
- columns: capacities per year (in GW)
- sheets: countries

2.2.4 Capacity potentials

File: `capacity_potentials;source={capacity_potentials}.xlsx`

This spreadsheet contains maximum capacity potentials for each technology, taking into account physical, technical, and legal constraints.

Spreadsheet structure:

- rows: voivodeships
- columns: potentials per year (in MW)
- sheets: technologies

2.2.5 Maximum capacity investments per year

File: `capacity_max_growth;source={capacity_max_growth}.xlsx`

This spreadsheet contains the upper bound on new capacity investments per year for each technology, taking into account technical, material, and labour constraints.

Spreadsheet structure:

- rows: technologies
- column: maximum annual investment (in MW)

2.2.6 Utilisation profiles of wind and solar power plants

Files: `{technology}_utilization_profile;source={renewable_utilization_profiles}.xlsx`

Those spreadsheets contain the utilisation profiles for wind and solar power plants. The profiles are hourly time series of available-to-installed power ratios, taking into account the effects of weather, season, and time of day. The utilisation profiles determine the upper bound on the utilisation of variable renewable energy sources.

Spreadsheet structure:

- rows: hours
- columns: utilisation factor per country/voivodeship
- sheets: weather years

2.2.7 CHP utilisation profiles

File: `chp_utilization_profile;source={chp_utilization_profiles}.xlsx`

Those spreadsheets contain the utilisation profiles for CHP plants. The utilisation of CHP plants follows the input utilisation profiles exactly. The utilisation of CHP plants is enforced (rather than optimised) to reflect the fact that CHP utilisation is mostly driven by heat demand which in turn is driven by the weather and season.

Spreadsheet structure:

- rows: fuel (Coal or Natural gas), hours
- columns: utilisation factor per voivodeship
- sheets: weather years

2.2.8 Annual utilisation of run-of-river hydro power plants

File: `hydro_utilization;source={hydro_utilization}.xlsx`

This spreadsheet contains the annual utilisation of run-of-river hydropower plants.

Spreadsheet structure:

- rows: countries
- column: annual utilisation factor

2.2.9 Annual energy demand

File: `demand;source={demand}.xlsx`

This spreadsheet contains the annual energy demand for each energy type.

Spreadsheet structure:

- rows: demand type (currently only Electricity)
- columns: demand per year (in TWh)

2.2.10 Annual electricity demand in neighbouring countries

File: `neighbors_demand;source={neighbors_capacity_demand}.xlsx`

This spreadsheet contains the annual electricity demand in neighbouring countries.

Spreadsheet structure:

- rows: countries
- columns: electricity demand per year (in TWh)

2.2.11 Load profiles

Files:

- `load_profile;source={load_profile}.xlsx`
- `neighbors_load_profile;source={neighbors_load_profile}.xlsx`

Those spreadsheets contain the load profiles for each country. The load profiles are hourly time series of the fraction of the annual electricity demand and they are based on historical load data. The load year has to match the weather year used to derive the utilisation profiles of vRES and CHP plants.

Spreadsheet structure:

- rows: hours
- columns: load factor per country
- sheets: load years

2.2.12 Nodes of the electricity grid

Files:

- `nodes;source={nodes}.xlsx`
- `neighbors.xlsx`

Those spreadsheets contain the nodes of the electricity grid. As the model supports the copper plate runs only, these input files do not influence the results at the moment.

Spreadsheet structure:

- rows: nodes
- columns: node coordinates

2.2.13 Cross-border flow capacities

File: `links;source={links}.xlsx`

This spreadsheet contains the cross-border flow capacities between Poland and neighbouring countries. It lists all individual interconnectors. It is possible to specify direction-dependent capacities.

Spreadsheet structure:

- rows: interconnectors
- columns: origin/destination nodes, capacity (in MW)
- sheets: years

2.2.14 Price assumptions

File: `prices;source={prices}.xlsx`

This spreadsheet contains the assumptions on fuel prices and CO_2 emission costs. The prices should be given in real values for a selected base year. Only rows with prices expressed in PLN are used by the model.

Spreadsheet structure:

- rows: prices
- columns: years

2.2.15 Technology data

File: `technology_data;source={technology_data}.xlsx`

This spreadsheet contains the capital costs, lifetimes, and other relevant technical parameters for each technology dependent on the investment period. Investment periods are defined by their last years.

Spreadsheet structure:

- rows: parameters
- columns: investment years
- sheets: technologies

2.2.16 Import/export prices

File: `trade_prices;source={trade_prices}.xlsx`

This spreadsheet is used only if `dynamic_trade_prices` parameter is set to `False`. It defines the prices at which electricity can be imported and exported.

Spreadsheet structure:

- rows: price type (import/export)
- columns: years

3 Model outputs

3.1 Raw PyPSA inputs and outputs

The internal representation of the PyPSA network (i.e. Python object storing information on the grid, generators, loads, power time series etc.) for each model run is stored in `data/runs` directory as a collection of `*.csv` files. Both input and output networks are stored.

3.2 Output statistics

Postprocessed output statistics (based on raw output networks) are stored in `data/output` directory, both as figures and `*.csv` files. So far the following statistics are available:

- total installed capacity per technology,
- annual production per technology,
- annual capacity utilisation per technology,
- annual CO_2 emissions per fuel,
- annual fuel consumption per fuel,
- annual curtailment of wind and PV energy,
- annual average operational reserve per technology.

4 Selected methodological details

4.1 Dispatch optimisation

The principle of dispatch optimisation is to minimise the total short-run marginal cost of the energy system while satisfying the electricity demand at each timestep. Dispatch optimisation can be performed jointly with investment optimization or under the assumption of fixed capacities. The marginal cost M_s of a utility unit s is composed of fuel, CO_2 emission, and operating cost

$$M_s = \frac{P_s}{\eta_s} + \frac{P_{CO_2}\,\varepsilon_s}{\eta_s} + V_s,$$

where P_s is the fuel cost per MWh_t of the primary energy, η_s is the net electrical efficiency of the unit s, P_{CO_2} is emission cost of 1 tonne of CO_2, ε_s is CO_2 emission coefficient (tCO_2 per MWh_t of the primary energy), and V_s is the variable operating cost per MWh_e of electricity produced. For hard coal, we include transportation expenses in the fuel cost.

4.2 Investment optimisation

In the PyPSA family of models, the total investment cost (including the cost of credit) is distributed over all the lifetime years of the asset (Brown, 2020). Since our model simulates only the last year within the given multi-year investment period, the yearly capital cost C_s of creating an asset s is identified with an annuitized total investment cost I_s plus the fixed annual cost of operation F_s.

$$C_s = \frac{I_s}{\sum_{y=1}^{L_s} \left(\frac{1}{1+r}\right)^{-y}} + F_s, \tag{1}$$

where L_s is the lifetime of the asset in years and r is the annual discount rate. At the moment a uniform discount rate is used for all assets.

Each investment period in PyPSA-PL has by default $N_p = 5$ years and its last year is identified as a reference year for which simulation is done (i.e., the investment period of the year 2030 encompasses the years 2026-2030). This is where PyPSA-PL differs from core PyPSA models (like PyPSA-Eur) where the first year of the investment period is used as a reference year. Our motivation for this change is that by taking the last year of the investment period as a reference year, we can interpret the investments more easily as those that are needed for the "target" demand at the end of the investment period.

Investment optimisation is always performed jointly with dispatch optimisation.

> ⚠ Only myopic optimisation is supported now
>
> At the moment only myopic optimisation has been fully implemented and tested, i.e. future investment periods are not considered in the investment optimisation for a given investment period. This should not be viewed as a major limitation of the model, as it can be argued that the perfect-foresight assumption could lead to overly optimized investment pathways – the real-world investors can only behave in a myopic way, as the future is uncertain.

4.3 Rescaling of the wind power utilisation factors

To enable distinguishing between the old and new onshore wind turbines, we allow the user to specify correction factors for wind power utilisation profiles. The starting point for the corrected utilisation profiles is provided by the profiles specified by the relevant input file, which are assumed valid for an "average" wind turbine. The annual correction factor is defined as the ratio of the assumed annual utilisation for the specific turbine type to the annual utilisation of an average wind turbine.

It would be physically incorrect to rescale the utilisation profiles just by multiplying all the hourly values by the annual correction factor, as this would lead to utilisation factors exceeding 1. For that reason, we transform the utilisation profiles in a non-linear way, which ensures that the values remain always below 1 and that the annual utilisation is modified according to the assumed annual correction factor. Specifically,

$$u'_t = [a + (1 - a)(2u_t - 1)]\, u_t,$$

where u_t is the original utilisation factor at hour t, u'_t is the corrected utilisation factor, and a is related to the annual correction factor A as follows:

$$a = \frac{A \langle u_t \rangle_t - \langle u_t(2u_t - 1) \rangle_t}{\langle u_t \rangle_t - \langle u_t(2u_t - 1) \rangle_t}.$$

The brackets $\langle \cdot \rangle_t$ denote the time average over the whole year:

$$\langle u_t \rangle_t = \frac{1}{8760} \sum_{t=1}^{8760} u_t.$$

5 Annex: flowchart representation of the optimisation pipeline

The processing steps presented in Figure 2 correspond to the logic of the `run_pypsa_pl` function defined in `src\pypsa_pl\run_pypsa_pl.py`.

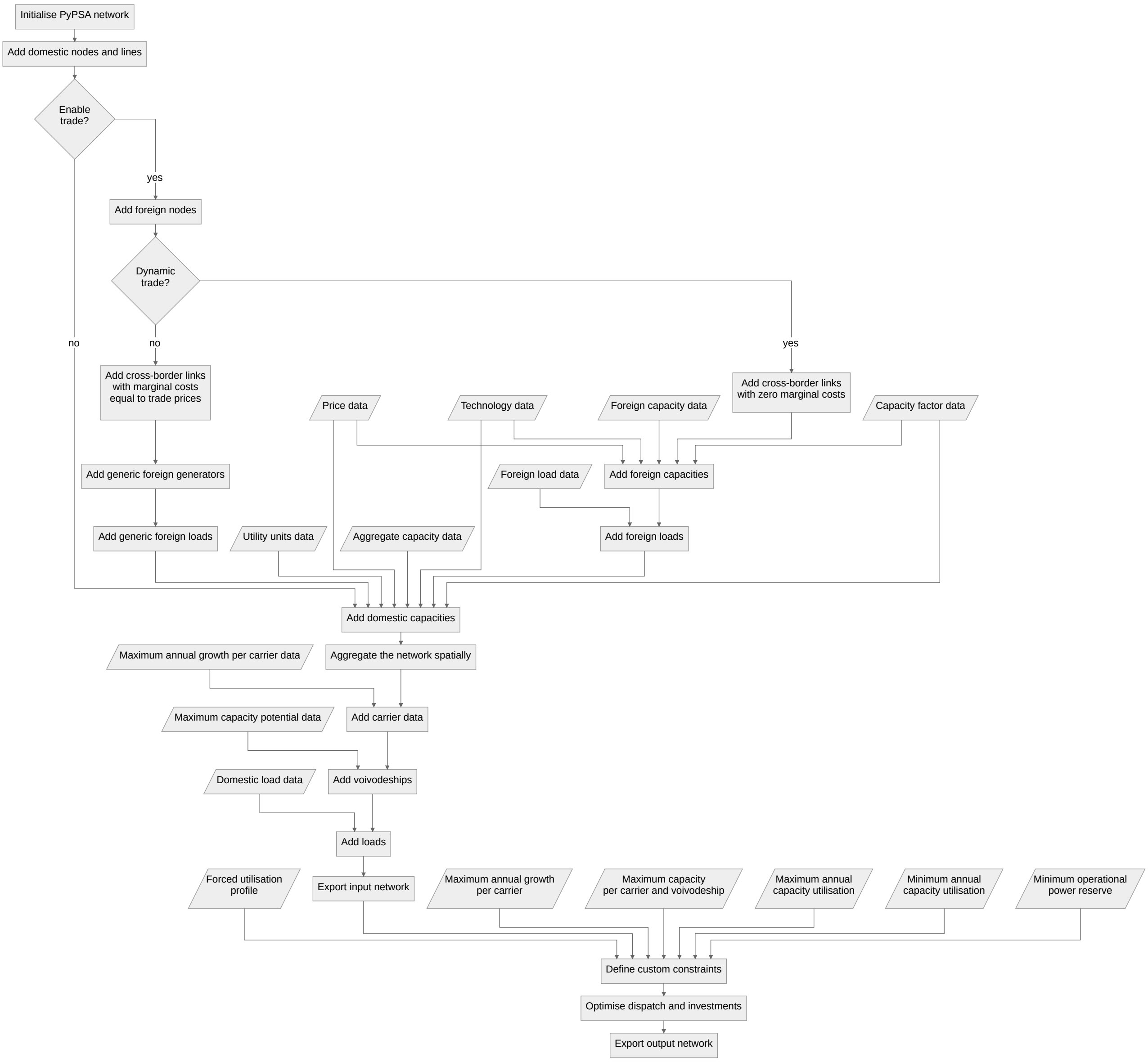

Figure 2: Flowchart representing the preparation and optimisation of the PyPSA network in the PyPSA-PL model

6 Changelog

- 2023-03-28: v2.0
  - new implementation of PyPSA-PL code, preserving PyPSA-PL v1 functionalities
  - most important new features:
    * investment optimisation
    * reserve requirement
    * explicit modelling of energy systems in neighbour countries and bidirectional trade flows
  - thorough update of default input data

7 References

Brown, T., 2020. Multi-Horizon Planning with Perfect Foresight. URL: https://nworbmot.org/energy/multihorizon.pdf.

Brown, T., Hörsch, J., Schlachtberger, D., 2018. PyPSA: Python for Power System Analysis. Journal of Open Research Software 6, 4. DOI: 10.5334/jors.188.