PyPSA-PL

Version 2.1. User guide

Patryk Kubiczek (patryk.kubiczek@instrat.pl)

2023-12-12

Table of Contents

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 power, heating, hydrogen, and light vehicle sectors – given the final use demand together with capital and operation costs for assets – or just to optimise the hourly dispatch of the utility units – given the final use demand and operation costs only.

The components of the PyPSA-PL model attachable to electricity, heat, light vehicles, and hydrogen nodes are presented in Figure 1. The energy flows and energy conversions between the components are modelled with high temporal resolution under the cost-optimality assumption.

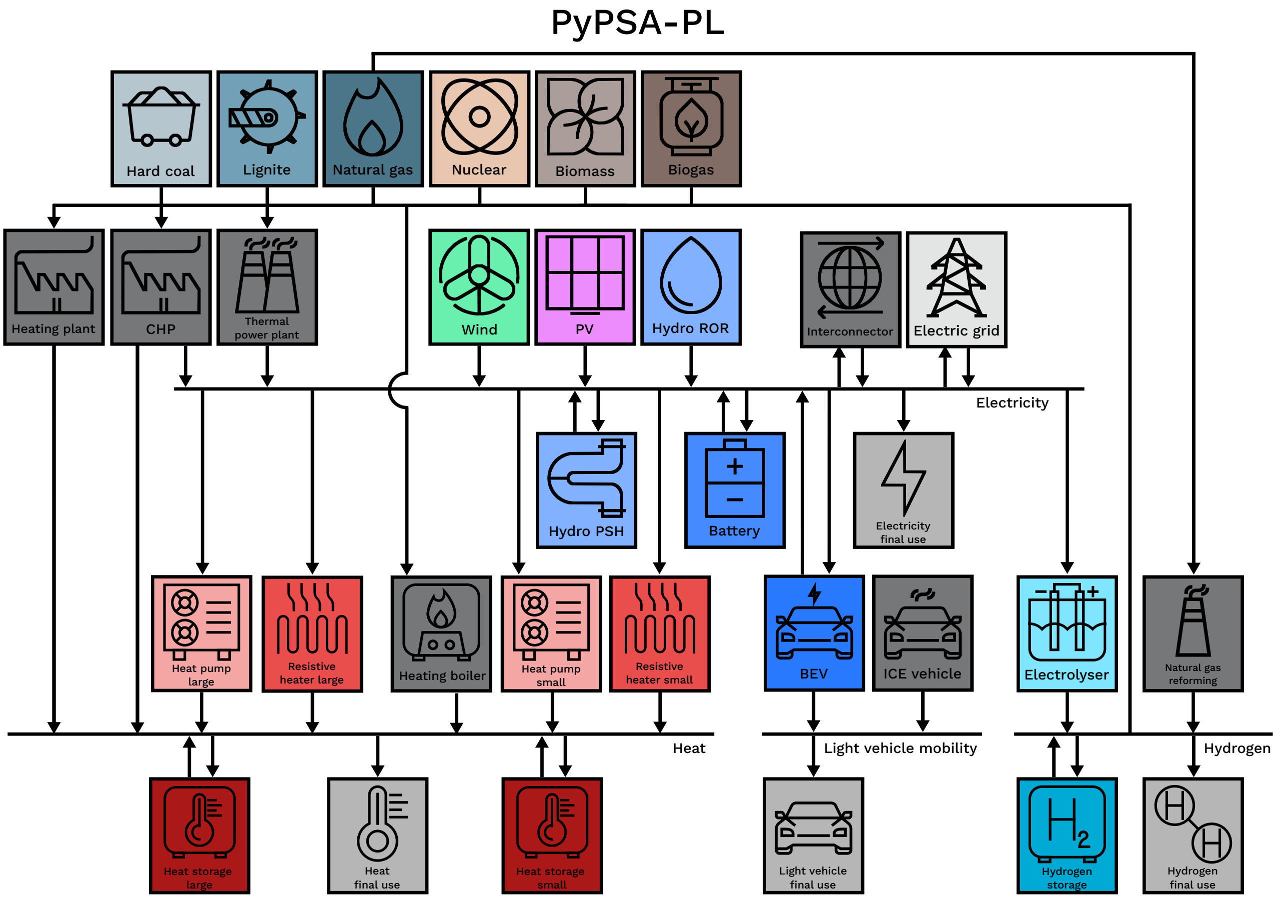

Figure 1: Energy flows and conversions between PyPSA-PL components

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`

Parameter	Meaning	Default value
`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	`instrat_ambitious`
`years`	years to be simulated	`[2025]`
`sectors`	sectors to be simulated	`["electricity", "heat", "light vehicles", "hydrogen"]`
`extendable_technologies`	list of technologies whose capacities should be optimally extended or `None` for pure dispatch optimisation	`None`
`decommission_only_technologies`	list of technologies whose capacities should be optimally retired prior to their end-of-life or `None` for pure dispatch optimisation	`None`
`extend_from_zero`	for extendable technologies, ignore the capacities defined by `aggregate_units` dataset?	`False`
`thermal_units`	source of thermal power plant data at the unit level	`None`
`renewable_units`	source of renewable power plant data at the unit level	`None`
`storage_units`	source of electricity storage unit data at the unit level	`pypsa_pl_v2.1`
`decommission_year_inclusive`	are the assets active in the decommissioning year?	`True`
`aggregate_units`	source of data on existing and planned capacity investments in the power sector per technology and area	`instrat_ambitious_copper_plate`

Parameter	Meaning	Default value
sectoral_units	source of data on existing and planned capacity investments in sectors other than power per technology and area	instrat_ambitious_copper_plate
exclude_technologies	list of technologies that are excluded from the model, even if they are listed in the input data	None
virtual_dsr	add virtual DSR capacity to avoid potential solution infeasibilities?	True
capacity_potentials	source of data on capacity potentials per technology and area	instrat_ambitious
capacity_max_growth	source of data on maximum capacity growth assumptions	instrat_ambitious
fuel_potentials	source of data on fuel availability potentials per fuel and area	instrat_2021
renewable_utilization_profiles	source of data on maximum hourly utilisation of wind and solar energy	PECD3+EMHIRES
hydro_utilization	source of data on annual utilisation of hydropower	entsoe_2020
chp_utilization_profiles	source of data on hourly utilisation of CHP plants (not used if heat sector is explicitly simulated)	regression
heat_pump_efficiency_profiles	source of data on hourly heat pump COP (coefficient of performance)	instrat
demand	source of annual demand data	instrat_2023
electricity_demand_correction	rescaling annual electricity demand	1
electricity_load_profile	source of hourly electricity demand profile data	entsoe
heat_load_profile	source of hourly heat demand profile data	degree_days+bdew
light_vehicles_load_profile	source of hourly light vehicle demand profile data	gddkia

Parameter	Meaning	Default value
load_profile_year	historical year used to create hourly demand and capacity utilisation profiles (2012, 2013, or 2015)	2012
temporal_resolution	length of a single timestep	1H
buses	source of power grid node data	voivodeships
grid_resolution	spatial resolution of the electricity grid	copper_plate
interconnectors	source of cross-border flow capacity data	pypsa_pl_v2.1
trade_factor	rescaling cross-border flow capacities	0.7
imports	allow electricity imports?	True
exports	allow electricity exports?	True
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 power 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
random_seed	random seed used for adding random noise from range [-1, 1] to marginal costs or None if no randomisation is desired	0

Parameter	Meaning	Default value
`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`
`enforce_bio`	minimum annual capacity utilisation factor for biomass and biogas power plants (expressed as a share of maximum technically available capacity factor)	`0`
`biogas_substrate_price_factor`	factor by which the price of agricultural biomass is multiplied to obtain the price of biogas substrate	`1`
`industrial_utilization`	fixed hourly utilisation of industrial power plants	`0.5`
`electricity_distribution_loss`	electricity transmission and distribution losses for sectoral electricity uses	`0.05`
`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`
`cold_reserve_need_per_demand`	required cold reserve capacity per hourly electricity load	`0`
`cold_reserve_need_per_demand`	required cold reserve capacity per hourly electricity imports	`0`
`cold_reserve_categories`	list of assets' categories that can be used to provide cold reserve capacity	`["JWCD"]`
`max_snsp`	maximum allowed hourly SNSP (System Non-Synchronous Penetration)	`1.0`

Parameter	Meaning	Default value
`ns_carriers`	list of non-synchronous carriers	`["Wind onshore", "Wind offshore", "PV roof", "PV ground", "DC", "Battery large", "Battery small", "BEV V2G"]`
`unit_commitment_categories`	assets' categories that are to be modelled as committable	`None`
`linearized_unit_commitment`	treat unit-commitment constraints in a linearised manner?	`True`
`district_heating_loss`	heat distribution losses in district heating systems	`0.12`
`district_heating_distribution_cost`	heat distribution cost in district heating systems (PLN/MWh_{th})	`72`
`share_district_heating_min`	minimum share of district heating in satisfying the total heat demand	`0.32`
`share_district_heating_max`	maximum share of district heating in satisfying the total heat demand	`0.32`
`share_biomass_boiler_min`	minimum share of individual biomass boilers in satisfying the total heat demand	`0.2`
`share_biomass_boiler_max`	maximum share of individual biomass boilers in satisfying the total heat demand	`0.2`
`max_resistive_to_heat_pump_ratio`	maximum ratio of annual heat production by individual resistive heaters to annual heat production by individual heat pumps	`0.02`
`bev_availability_max`	maximum availability of BEVs for charging or discharging to grid	`0.95`
`bev_availability_mean`	average availability of BEVs for charging or discharging to grid	`0.8`

Parameter	Meaning	Default value
`v2g_factor`	share of BEV chargers enabling V2G service	`0.25`
`minimum_bev_charge_hour`	hour of the day when the minimum BEV charging level is required	`6`
`minimum_bev_charge_level`	minimum charging level of BEV batteries at the hour specified by `minimum_bev_charge_hour`	`0.75`
`srmc_v2g`	marginal cost of V2G service (excluding electricity cost) (PLN/MWh$_e$)	`50`
`oil_to_petroleum_product_price_ratio`	ratio of crude oil price to the price of oil-based fuels	`2`

⚠ Only copper plate spatial resolution is supported now

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

⚠ Spinning reserve constraints are not supported in v2.1

Support for this feature, including different kind of reserves (primary, secondary, tertiary) might be added in the future.

Total installed capacity

The total installed capacity in a simulation is an aggregation of data defined by `thermal_units`, `renewable_units`, `storage_units`, `aggregate_units`, and `sectoral_units` together with optimal investments and retirements for technologies listed in `extendable_technologies` and `decommission_only_technologies`, excluding any technologies listed in `exclude_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 and `*.csv` files. 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.

Ignored sheets, columns, and rows

In `*.xlsx` spreadsheets, all sheets, columns, and rows that start with `#` are ignored by the model.

2.2.1 Existing and planned utility units

Files:

- `thermal_units;source={thermal_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 groups

2.2.2 Existing and planned aggregate capacity investments

Files:

- `aggregate_units;source={aggregate_units}.xlsx`
- `sectoral_units;source={sectoral_units}.xlsx`

The remaining utility units are aggregated at the area level (voivodeship or country) and their additions 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 those files can be subject to investment optimisations.

It is also possible to specify the total 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. run-of-river plants) or that are subject to early retirement optimisation (e.g. coal power plants).

For sectoral units, the capacities have to be attributed to a specific technology bundle. A technology bundle delivers a fixed hourly share of the total sectoral demand – this share is subject to optimisation.

> Technology bundles
>
> Technology bundles are a way to go beyond the copper-plate assumptions in sectors other than power. For example, we assume that some buildings are equipped with heat pumps supported by resistive heaters and heat storage, while others are equipped with individual gas boilers only. No heat flows between those two kinds of buildings should be possible.

Spreadsheet structure:

- rows: areas, categories, technologies, carriers, sectors (only sectoral units), technology bundles (only sectoral units), value types (addition or total)
- columns: capacities per year (in MW)
- sheets: unique unit groups

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: categories, technologies, carriers
- 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: areas
- columns: potentials per year (in MW or MWh)
- sheets: carriers

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 or MWh)

2.2.6 Fuel potentials

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

This spreadsheet contains maximum annual fuel availabilities for each fuel, taking into account environmental or technical constraints.

Spreadsheet structure:

- rows: areas
- columns: fuel availability per year (in TWh)
- sheets: fuels

2.2.7 Utilisation profiles of wind and solar power plants

Files: `timeseries/{technology}_utilization_profile;source={renewable_utilization_profiles};year={load_profile_year}.csv`

Those files 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.

CSV file structure:

- rows: hours
- columns: utilisation factor per area

2.2.8 Efficiency profiles of heat pumps

Files: `timeseries/{technology}_efficiency_profile;source={heat_pump_efficiency_profiles}; year={load_profile_year}.csv`

Those files contain the efficiency profiles for heat pumps. The profiles are hourly time series of COP (coefficient of performance) values, taking into account the effects of ambient temperature. Different COP profiles are used for individual and systemic heat pumps.

CSV file structure:

- rows: hours
- columns: COP per area

2.2.9 CHP utilisation profiles

File: `timeseries/chp_{fuel}_profile;source={chp_utilization_profiles}; year={load_profile_year}.csv`

Those files contain the utilisation profiles for CHP plants. The utilisation of CHP plants is enforced by the input utilisation profiles if heating sector is not explicitly modelled.

CSV file structure:

- rows: hours
- columns: utilisation factor per area

2.2.10 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

2.2.11 Annual energy demand

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

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

Spreadsheet structure:

- rows: areas
- columns: demand per year (in TWh)
- sheets: energy types

2.2.12 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)

2.2.13 Load profiles

Files:

- `electricity_load_profile;source={electricity_load_profile};year={load_profile_year}.csv`
- `heat_space_load_profile;source={heat_load_profile};year={load_profile_year}.csv`
- `light_vehicles_load_profile;source={light_vehicles_load_profile};year={load_profile_year}.csv`
- `neighbors_load_profile;source={neighbors_load_profile};year={load_profile_year}.csv`

Those files contain the load profiles in different sectors for each area (including electricity load profiles for neighbouring countries). The load profiles are hourly time series of the fraction of the annual sectoral demand and they are based on historical load data (electricity) or are generated synthetically (space heating, mobility of light vehicles). Water heating and hydrogen demand profiles are assumed constant, hence no input is needed. The load year has to match the weather year used to derive the utilisation profiles of vRES and CHP plants.

CSV file structure:

- rows: hours
- columns: load factor per area

2.2.14 Nodes of the electricity grid

Files:

- `buses;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

2.2.15 Cross-border flow capacities

File: `interconnectors;source={interconnectors}.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: individual interconnector properties

2.2.16 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

2.2.17 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: technology years
- sheets: technologies

2.2.18 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 energy production and consumption 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,
- capacity additions per technology,
- total investment costs per technology,
- annuitised investment costs per technology,
- annual fixed costs per technology,
- total annual operating costs per technology,
- annual CO_2 emission costs per technology,
- production and consumption structure in selected peak hours of the year (top exogenous demand, top vRES production, top residual load, top total load).

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 all energy demands 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 (lower heating value), η_s is the net 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 of useful energy produced.

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 annuitised 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`.

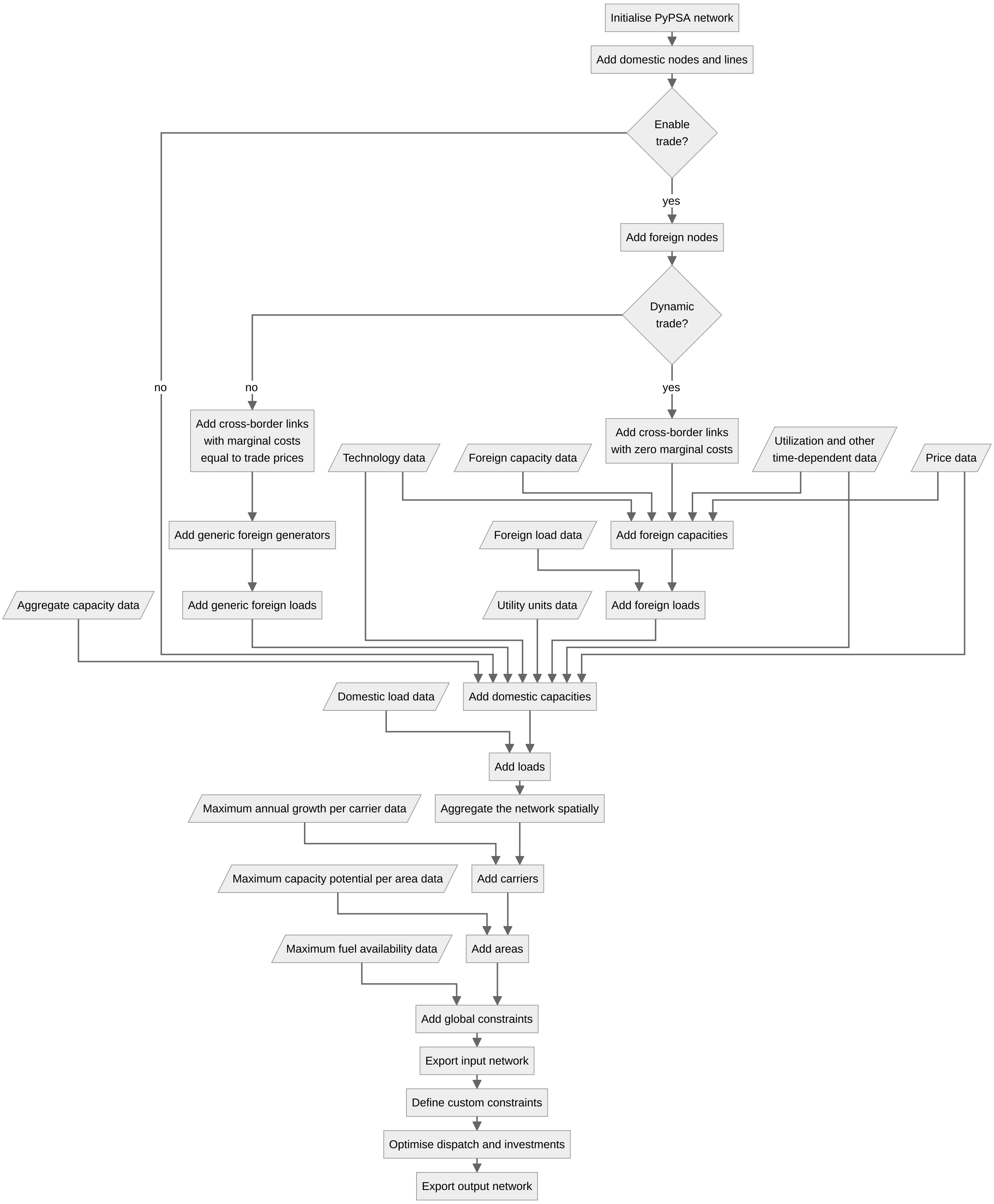

Figure 2: Flowchart representing the preparation and optimisation of the PyPSA network in the PyPSA-PL model

6 Changelog

- 2023-12-12: v2.1
  - major extension of the model by the heating, hydrogen, and light vehicle sectors
  - general updates:
    - change the format of the demand data
    - change the extension of the input files with timeseries from `*.xlsx` to `*.csv`
    - distinguish between category, technology, and carrier for assets in the input data
    - implement optimisation of technology bundles' shares
    - allow optimisation of early retirements of existing assets based on their fixed costs
    - add new statistics of output data
  - updates in the electricity sector:
    - base the interconnector capacities on historical cross-border flow data and change the format of interconnector input data
    - add constraint on the maximum allowed hourly SNSP (System Non-Synchronous Penetration)
    - allow simulating thermal power plants in an aggregate manner or as committable units
    - model biogas production, storage, and consumption (for electricity and heat generation) in a separate manner
    - constrain the biomass and biogas power plants by the fuel availability rather than by the maximum installed capacity
- 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. (Visited on 30/08/2022).