NTNU

Documentation and Guidelines:
Building and running the EMPIRE Model

The Department of Industrial Economics and Technology Management (IØT)

The Norwegian University of Science and Technology (NTNU)
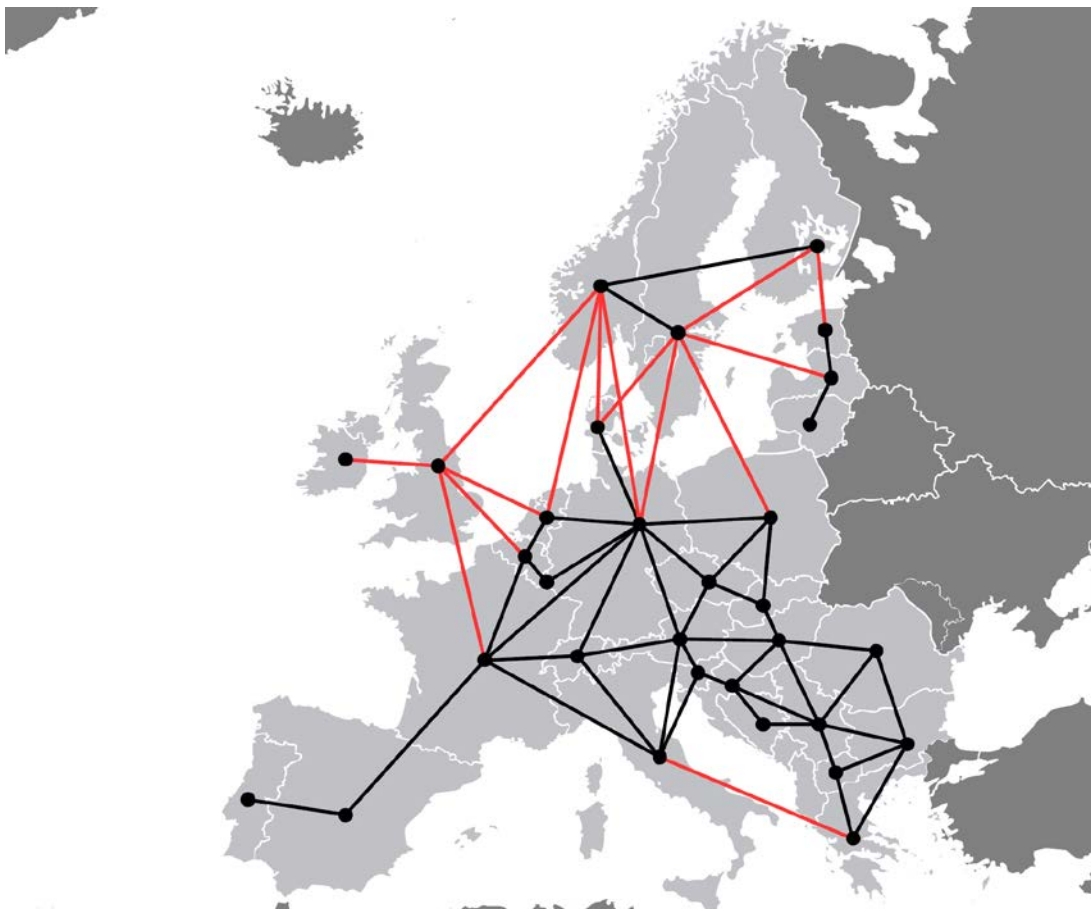
Compiled by:
Stian Backe
NTNU
Email: stian.backe@ntnu.no

Table of Contents

1 Introduction

The EMPIRE model is a stochastic linear programming model to analyze developments of the European power market. It was created as part of Dr. C. Skar's doctoral thesis supervised by Prof. A. Tomasgard, and it has been used to analyze decarbonization of the European power system focusing on the supply side of the power market. The purpose of the model is to support long-term capacity expansion of the power system under short-term operational uncertainty with a special focus of representing variable renewable energy sources (VRES), namely wind, solar and hydro power. Energy demand, as well as investment options for energy supply technologies, their related costs and operational characteristics, are exogenous input data in the model. The output supports decisions regarding technology choices, investment volume and investment timing with requirements of ensuring a reliable energy system at minimum cost. The model output also supports short-term operational decisions of energy system components under uncertainty. The model is subject to operational uncertainty, so the stochastic scenarios represent different realizations of the following: (1) Availability of VRES (wind and solar), (2) availability of hydro reservoirs and (3) hourly electricity loads.

This documentation contains three main parts. Firstly, where to download and run the EMPIRE model is explained. The licensing and how to contribute working on EMPIRE are also described. Furthermore, a short overview on the mathematical formulation and the preprocessed calculations is presented. Secondly, the documentation gives a detailed explanation about the input data. This contains information about the input structure, as well as different kinds of data needed to run the model. The input data can be provided as Excel files, or as .tab-files, which is also further explained in this chapter. Lastly, you will find a description of the output data and what kind results EMPIRE produces. It also shows possibilities to visualize and interpret results.

2 The EMPIRE Model

This section provides information about editing and running the EMPIRE model. It shows how to download it and what software is needed. You will also find a license overview and an overview on the mathematical description.

2.1 Required Software

The EMPIRE model is available in the Python-based, open-source optimization modelling language Pyomo. Running the model thus requires some coding skills in Python. To run the model, make sure Python, Pyomo and a third-party solver like Gurobi or CPLEX is installed and loaded to the respective computer or cluster. More information on how to install Python and Pyomo can be found here.

All scripts and data can be downloaded from the Git repository called 'OpenEMPIRE'. To download, you need to install Git and clone the repository. Note that the repository makes use of Git Large File Storage (LFS) which also needs to be installed for input data-files to be downloaded when cloning the repository. Once both Git and Git LFS has been successfully installed, the model is downloaded to the desired directory with the following commands in the command prompt or Git BASH:

```
$ cd <path to directory>
$ git clone https://github.com/ntnuiotenergy/OpenEMPIRE.git
```

2.2 Git-Repository

EMPIRE consists of five programming scripts:

(1) **run.py:** The main script used to run EMPIRE. It links to all other scripts. This is the only script a *user* of EMPIRE needs to use and potentially modify.

(2) **Empire.py:** Contains the abstract formulation of EMPIRE in Pyomo. This script also contains code related to printing the results.

(3) **scenario_random.py:** Generates random operational scenarios as .tab-files through sampling.

(4) **reader.py:** Generates .tab-files input based on data provided in Excel workbooks.

(5) **test_run.py:** Same as the main run-script (run.py), but it is linked to a small test instance of EMPIRE that usually finishes in 1-2 min.

In the repository, the 'Data handler'-folder contains the Excel workbooks that are used to store and modify input data. The workbooks are contained within folders representing instance-versions of EMPIRE, e.g. 'europe_v50'. The 'test'-folder contains input data for a small test-instance of EMPIRE. For more details regarding input data, see Section 3.

Within an instance-version in the 'Data handler'-folder, there is a folder called 'ScenarioData' containing large data sets used to generate stochastic scenarios in EMPIRE. If EMPIRE is run with

random scenario generation, representative time series are sampled once per scenario and season for each random input parameter. For more details on scenario generation, see Section 4.

2.3 Running

To run EMPIRE, make sure Python, Pyomo and a third-party solver like Gurobi or CPLEX is installed and loaded to the respective computer or cluster. Before staring your EMPIRE-run, you should perform a test run by running the 'test_run.py' script to check that Pyomo and a third-party solver is installed properly. Additionally, you should open the 'test_run.py' and/or the 'run.py'-script in your preferred text editor and confirm that all settings under '##USER##' are correct, including e.g. details on which instance-version you are running, the temporal dimensions of your instance, and whether or not to generate new stochastic scenarios. An overview and description of user settings description to be confirmed/edited before running EMPIRE is presented in Table 1.

EMPIRE is run from the directory in which EMPIRE was cloned by calling the 'run.py'-script:

```
C:\Users\name> cd <path to directory>              #Change directory

C:\Users\name\path_to_folder> python run.py        #Run the code
```

Note that generating scenarios and building the instance in Pyomo for a base case of EMPIRE can take around 40 min.

Table 1 Overview of settings in the 'run.py'-script

Input name	Type	Default	Description
USE_TEMP_DIR	True/ False	False	If true, all instance-files related to solving EMPIRE is stored in the directory defined by **temp_dir** (see below). This is useful when running a large instance of EMPIRE to avoid memory problems.
temp_dir	Text-string	'/panfs/nas-0-0.local/work/stianba c'	The path to which temporary files will be stored if **USE_TEMP_DIR = True**; .lp-file is stored if **WRITE_LP = True**; and .plk-file is stored if **PICKLE_INSTANCE = True.**
version	Text-string	'europe_v50'	The name of the version to be run. Note that this is the folder-name containing input data in 'Data handler'
NoOfPeriods	Integer	8	The number of investment periods. NB! Must correspond with data for version
NoOfScenarios	Integer	3	The number of scenarios in every investment period.

NoOfRegSeason	Integer	4	The number of regular seasons in every investment period. NB! Must correspond with scenario generation
lengthRegSeason	Integer	168	The number of chronological time steps in a regular season. NB! Must correspond with data for version
regular_seasons	Array	["winter", "spring", "summer", "fall"]	The identity-string for the regular seasons. NB! Must correspond with data for version
NoOfPeakSeason	Integer	2	The number of regular seasons in every investment period. NB! Must correspond with data for version
lengthPeakSeason	Integer	24	The number of chronological time steps in a peak season. NB! Must correspond with data for version
discountrate	Float	0.05	The discount rate
WACC	Float	0.05	The weighted average cost of capital (WACC)
LeapYearsInvestment	Integer	5	The number of years covered in one investment period
solver	Text-string	"Xpress"	Specifies the solver. Options: "Xpress", "Gurobi", "CPLEX"
scenariogeneration	True/ False	True	If true, new operational scenarios will be generated. NB! If false, .tab-files must be manually added to the 'ScenarioData'-folder in the **version**.
EMISSION_CAP	True/ False	True	If true, emissions in every scenario are capped according to the specified cap in 'General.xlsx'. If false, the CO2-price specified in 'General.xlsx' applies.
WRITE_LP	True/ False	False	If true, the solver-file will be saved. Useful for debugging.
PICKLE_INSTANCE	True/ False	False	If true, instance will be saved/pickled. Useful for printing alternative results

2.4 Licensing

The EMPIRE model and all additional files in the git repository are licensed under the MIT license. In short, that means you can use and change the code of EMPIRE. Furthermore, you can change the license in your redistribution but must mention the original author. We appreciate if you inform us about changes and send a merge request via git.

For further information please read the LICENSE file, which contains the license text, or go to https://opensource.org/licenses/MIT

2.5 Mathematical Description

EMPIRE is a multi-horizon stochastic linear program, and it has been designed to support capacity expansion of the power system. The model represents a network of nodes and arcs where decisions are made in two temporal scales: investment time steps and operational time steps. Operational decisions are subject to uncertainty that is discretized in several stochastic scenarios.

The abstract stochastic programming model can be formulated in the following way:

$$\min z = \sum_{i \in \mathcal{I}} \delta_i \big(c_i x_i + \vartheta \sum_{\omega \in \Omega} \pi_\omega \sum_{h \in \mathcal{H}} q_i y_{i,h,\omega} \big) \tag{1}$$

$$s.t. \quad a_i x_i \le b_i, \qquad\qquad\qquad i \in \mathcal{I}, \tag{2}$$

$$x_i \ge 0, \qquad\qquad\qquad i \in \mathcal{I}, \tag{3}$$

$$w_i y_{i,h,\omega} + t_{i,h,\omega} x_i \le h_{i,h,\omega}, \qquad i \in \mathcal{I}, \ h \in \mathcal{H}, \ \omega \in \Omega, \tag{4}$$

$$y_{i,h,\omega} \ge 0, \qquad\qquad\qquad i \in \mathcal{I}, \ h \in \mathcal{H}, \ \omega \in \Omega. \tag{5}$$

where the set $\mathcal{I}$ represents investment periods, the set $\mathcal{H}$ represents operational periods and the set Ω represents a set of operational scenarios. The variables x_i represent investment decisions in investment period $i \in I$, while the variables $y_{i,h,\omega}$ represent operational decisions in operational period $h \in \mathcal{H}$ and scenario $\omega \in \Omega$ within investment period $i \in \mathcal{I}$. These variables are also dependent on node (location) and asset type (generator, storage, transmission).

The expression (1) is the objective function identifying total costs for investing and operating assets in the system. The total costs are quantified by summing over all time periods, both investment and operational. The term $\delta_i = (1+r)^{-n(i-1)}$ discounts all future costs at an annual discount rate of r with n years in between each investment period such that all costs are given with respect to the value in the first investment period. The term $\vartheta = \alpha \sum_{j=0}^{n}(1+r)^{-j}$ scales operational costs to annual values through α and discounts this annual operational cost n years ahead (until the next investment period). The input c_i represent investment costs and q_i represent operational costs in investment period $i \in \mathcal{I}$. Since the operational decisions are subject to uncertainty, the operational costs in (1) represent the *expected* operational costs by scaling with π_ω representing the probability that scenario $\omega \in \Omega$ will occur.

Constraints (2)-(3) make sure investments in all assets are bounded, that the lifetime of the asset is considered across the investment periods and that there are no negative investments. Constraints (4)-(5) make sure operation of assets is bounded by investment decisions and asset availability, that supply balances demand (and possibly lost load), that energy levels in storages is balanced, that thermal generators have limited up-ramping capabilities, that policies (e.g. emission caps) are respected, and that operation of assets is not negative. The input w_i represent type dependent operational specifications for assets, while $t_{i,h,\omega}$ represent scenario dependent operational

specifications. The latter input data affect the value of investment decisions in different operational scenarios.

For a more detailed mathematical formulation, see:

Skar et.al., "A multi-horizon stochastic programming model for the European power system", CenSES working paper 2, ISBN: 978-82-93198-13-0, 2016.

The EMPIRE formulation supports investment decisions in power generation, storage, and transmission with an objective of minimizing total system cost, which means the model simulates perfect competition. The strength of the model is that these investment decisions are made linked with the chronological operation of the assets to satisfy demand, and that the chronological operations are subject to uncertainty.

EMPIRE is generally used to investigate pathways for decarbonizing the power sector in Europe. However, this is instance specific, meaning the abstract model could be used for other places than Europe or other sectors than power that can be modelled as a network flow, e.g. capacity expansion of a gas system.

3 Input Data

The EMPIRE Model reads .tab-files, which provide all needed sets and input data. For editing and storing the data, excel-files are used.

There are seven excel-files in total of which six contain indexed input data and one is to provide the indices/sets. The excel-files are sorted by the following categories: General data, generation data, country/node data, set/index data, transmission data, and storage data. These files contain multiple tables regarding for example investment costs and initial capacity. In the following, the content of the excel-files is described. The general structure of all files is given before each file and its content is presented.

3.1 Structure of the input files

We differentiate the structure of the data files and the set/index file. The difference between these two file types is that the data files have indices while the set file defines the indices. Every excel-file contains multiple worksheets. In every sheet, there is one data table and some additional data.

For the data files, the first row in every sheet is for the source, i.e. where the data is from. The second row provides a brief description of the data and its role in EMPIRE. The rows beneath contain the table with header and values. Therefore, the tables have the structure like the table beneath with N index columns and a value column:

Source			
Description			
Column Name 1	(...)	Column Name N	Value
Index 1.1	(...)	Index N.1	value 1
Index 1.2	(...)	Index N.2	value 2

For the set/index files, the sheets are separated by groups. There are two different kinds of sheets. In the first kind, columns are filled with the sets/indices, and there are no source or description rows. Every sheet contains a group and every column a type of this group (e.g. group 'Generator' with categories of generator types). The other kind of set/index sheets contain tuples defining subsets or sets with double index, e.g. transmission connections between countries.

3.2 Description of the Input Files

3.2.1 Sets

The set file contains indices for parameters and variables used in the model. Each sheet contains a group of sets/indices or double index sets. One can add values to these columns to expand the dimensions of an instance. Indices defined here must be consistent with the other data files or the user input in the 'run.py'-script.

In the following, the five different groups of sets/indices and a short description of them is given:

- **Nodes**

  This sheet contains one column. The name of the column is 'Node' and it contains all countries that are used in the model

- **Storage**

  This sheet contains two columns:
  - **Storage:** All types of storages.
  - **Dependent Storage:** Storage types where charging/discharging capacity is dependent on the energy storage capacity

- **Technology**

  This sheet contains all the technology groups for generators in one column. The technology groups are used to put resource restrictions on generator types using the same resource.

- **Generators**

  This sheet contains all types of generators that can be used in the model. They may rely on the same technology.
  - **Generator:** All types of used generators.
  - **Hydro Generator:** All generators using hydro power.
  - **Hydro Generator with Reservoir:** All generators using hydro power that can be regulated.
  - **Thermal Generators:** All generators incinerating fuel to produce electricity. These generators are subject to ramping constraints.

- **Line Type**

  This sheet contains different transmission line types in one column

Additionally, there are sheets for defining double index sets. Each sheet contains two or three columns:

- **Storage at Nodes:** Available storages per country
- **Directional Lines:** Existing/possible connections between countries
- **Line Type of Directional Lines:** Define the line type of the transmission connections
- **Generators of Node:** Available generators per country
- **Generators of Technologies:** Categorize generator type by technology (resource)

3.2.2 Generator

The file 'Generator.xlsx' contains different data regarding the generator technologies:

- **Capital Costs**
  - Capital costs per kW of all generator types in all investment periods in three columns:
    - Index: Generator type
    - Index: Period
    - Value: Total capital costs in EUR per kW (default: 0)

- **Fixed OM Costs**
  - Fixed annual operation and maintenance costs for generator technologies in EUR per kW-year in three columns:
    - Index: Generator type

- ▪ Index: Period
  - ▪ Value: Fixed OM Costs in EUR per kW (default: 0)
- **Variable OM Costs**
  - o Operation dependent operation and maintenance costs for generator types in EUR per MWh/h in three columns:
    - ▪ Index: Generator type
    - ▪ Index: Period
    - ▪ Value: Variable OM Costs in EUR per MWh (default: 0)
- **Fuel Costs**
  - o Period dependent fuel costs for generator types in EUR per GJ in three columns:
    - ▪ Index: Generator Technology
    - ▪ Index: Period
    - ▪ Value: Fuel Costs in EUR per GJ (default: 0)
- **CCS Costs TS Variable**
  - o Costs of transporting and storing captured CO2 in EUR per tonCO2eq. in two columns:
    - ▪ Index: Period
    - ▪ Value: CCS TS costs in euro per tCO2 (default: 0)
- **Efficiency**
  - o Efficiency of converting fuel to electricity for generator types in each period in three columns:
    - ▪ Index: Generator type
    - ▪ Index: Period
    - ▪ Value: Generator Efficiency in MWh-electricity per MWh-fuel (default: 1)
- **Ref Initial Cap**
  - o The capacity in the reference investment period in three columns:
    - ▪ Index: Node
    - ▪ Index: Generator type
    - ▪ Value: Generator Reference Initial Capacity in MW (default: 0)
- **Scale Factor Initial Cap**
  - o The share of capacity that retired compared to the reference period in three columns (value = 0 means no retirement of Ref Initial Cap):
    - ▪ Index: Generator Technology
    - ▪ Index: Period
    - ▪ Value: Generator Retirement Factor Initial Capacity (default: 0)
- **Initial Capacity**

(NB! Set to default (0) when using 'Ref Initial Cap' and 'Scale Factor Initial Cap')
  - o The initial capacity in all investment periods in four columns:
    - ▪ Index: Node
    - ▪ Index: Generator type
    - ▪ Index: Period
    - ▪ Value: Generator initial Capacity in MW (default: 0)
- **Maximum Built Capacity**

- o Use to restrict the capacity expansion of certain technologies. The generation capacity that can maximally be built in an investment period for any country in four columns:
    - Index: Node
    - Index: Technology **(NB! Technology, NOT generator type)**
    - Index: Period
    - Value: Maximum Built Capacity in MW (default: 500 000)
- **Maximum Installation Capacity**
  - o The maximum capacity that can exist of a generator technology (resource limit) in any country or investment period in three columns:
    - Index: Node
    - Index: Technology **(NB! Technology, NOT generator type)**
    - Value: Maximum Installed Capacity in MW (default: 0)
- **Ramp Rate**
  - o The maximum change of output from one hour to the next hour for thermal generators in two columns:
    - Index: Thermal Generator
    - Value: Ramp Rate (default: 0)
- **Generator Availability**
  (NB! Set to default (0) when the generator have stochastic availability)
  - o The availability factor as a share of installed capacity for all defined generators in two columns:
    - Index: Generator
    - Value: Availability (default: 0)
- **CO2 Content**
  - o The CO2 intensity of generator type depending on fuel in two columns:
    - Index: Generator type
    - Value: CO2 Content in tCO2/GJ (default: 0)
- **Lifetime**
  - o The lifetime of a generator type in years in two columns:
    - Index: Generator type
    - Value: Lifetime in years (default: 0)

3.2.3 Nodes

The file 'Node.xlsx' contains the data specific to all specified countries:

- **Electric Annual Demand**
  - o The annual demand in nodes used to adjust hourly load profiles for future investment periods in three columns:
    - Index: Node
    - Index: Period
    - Value: Annual electric demand in MWh (default: 0)
- **Node Lost Load Cost**
  - o The cost of not generating electricity in an hour in three columns:
    - Index: Node

- ▪ Index: Period
        - ▪ Value: Node Lost Load Cost in EUR (default: 22 000)
- **Hydro Generator Maximum Annual Production**
    - o The maximum production of all regulated hydro generators in a country per year in two columns:
        - ▪ Index: Node
        - ▪ Value: Max production in MWh

3.2.4 General

The file 'General.xlsx' contains the scale factor for all seasons and data related to emission policies:

- **Season Scale**
    - o The scaling of each representative season to add up to a full year in two columns:
        - ▪ Index: Season
        - ▪ Value: Season Scale (default: 1)
- **CO2 Cap**
  (NB! If EMISSION_CAP = True)
    - o The maximum allowed annual emissions for all countries combined in any scenario of an investment period in two columns:
        - ▪ Index: Period
        - ▪ Value: CO2 Cap in Mton CO2 per year (default: 5 000)
- **CO2 Price**
  (NB! If EMISSION_CAP = False)
    - o The assumed CO2 price adding to the operational costs of CO2 emitting generators in two columns:
        - ▪ Index: Period
        - ▪ Value: CO2 price in EUR per tCO2 (default: 0)

3.2.5 Storages

The file 'Storage.xlsx' contains data regarding the storage technologies:

- **Power Initial Capacity**
    - o The initial capacity of charging/discharging storage capacity for a period in four columns:
        - ▪ Index: Nodes
        - ▪ Index: Storage Types
        - ▪ Index: Period
        - ▪ Value: Initial Capacity in MW (default: 0)
- **Power Capital Costs**
    - o The capital cost for investing in charging/discharging storage capacity in three columns:
        - ▪ Index: Storage Type
        - ▪ Index: Period
        - ▪ Value: Capital Cost in Euro per kW (default: 0)
- **Power Fixed OM Costs**

- o The fixed operation and maintenance cost for investing in charging/discharging storage capacity in three columns:
    - Index: Storage Type
    - Index: Period
    - Value: Fixed OM Costs in Euro per kW (default: 0)
- **Power Max Built Capacity**
    - o The maximum capacity of charging/discharging storage capacity that can be built in a period in four columns:
        - Index: Nodes
        - Index: Storage Types
        - Index: Period
        - Value: Max Built Capacity in MW (default: 500 000)
- **Power Max Installed Capacity**
    - o The maximum installed charging/discharging storage capacity in any period in three columns:
        - Index: Nodes
        - Index: Storage Types
        - Value: Max installed Capacity in MW (default: 0)
- **Energy Capital Costs**
    - o The capital cost for investing in energy storage capacity in three columns:
        - Index: Storage Type
        - Index: Period
        - Value: Capital Cost in EUR per kWh (default: 0)
- **Energy Fixed OM Costs**
    - o The fixed operation and maintenance cost for investing in energy storage capacity in three columns:
        - Index: Storage Type
        - Index: Period
        - Value: Fixed OM Costs in EUR per kWh (default: 0)
- **Energy Initial Capacity**
    - o The initial energy storages capacity in a period in four columns:
        - Index: Nodes
        - Index: Storage Types
        - Index: Period
        - Value: Initial Capacity in MWh (default: 0)
- **Energy max Built Capacity**
    - o The maximum energy storage capacity that can be built in a period in four columns:
        - Index: Nodes
        - Index: Storage Types
        - Index: Period
        - Value: Max Built Capacity in MWh (default: 500 000)
- **Energy Max Installed Capacity**
    - o The maximum installed energy storage capacity in any period in three columns:

- ▪ Index: Nodes
- ▪ Index: Storage Types
- ▪ Value: Max installed Capacity in MWh (default: 0)
- **Storage Initial Energy Level**
  - o The initial energy level of a storage as a percentage of the installed energy capacity in two columns:
    - ▪ Index: Storage Type
    - ▪ Value: Initial Energy Level as percentage of Installed Energy Capacity (default: 0)
- **Storage Charge Efficiency**
  - o The efficiency of charging a storage (non-spillage during charging) in two columns:
    - ▪ Index: Storage Type
    - ▪ Value: Storage Charging Efficiency (default: 1)
- **Storage Discharge Efficiency**
  - o The efficiency of discharging a storage (non-spillage during discharging) in two columns:
    - ▪ Index: Storage Type
    - ▪ Value: Storage Discharging Efficiency (default: 1)
- **Storage Power to Energy**
  - o The required ratio between installed power and energy storage for dependent storage in two columns:
    - ▪ Index: Dependent Storage Type
    - ▪ Value: Storage Ratio (default: 1)
- **Storage Bleed Efficiency**
  - o The hourly percentage of spillage (self-discharge) in two columns (value = 1 means no self-discharge):
    - ▪ Index: Storage Type
    - ▪ Value: Storage Bleed Efficiency (default: 1)
- **Lifetime**
  - o Lifetime of storage types in years in two columns:
    - ▪ Index: Storage Type
    - ▪ Value: Lifetime in years (default: 0)

3.2.6 Transmission

The file 'Transmission.xlsx' contains the data specific to all transmission connections between countries:

- **Line Efficiency**
  - o The percentage of transmission that reaches destination in any time step in three columns:
    - ▪ Index: From Node
    - ▪ Index: To Node
    - ▪ Value: Line Efficiency (default: 0.97)
- **Max Install Capacity**

- o The maximum allowed capacity of transmission between nodes in the given investment period in four columns:
  - Index: From Node
  - Index: To Node
  - Index: Period
  - Value: Max Install Capacity in MW (default: 0)
- **Maximum Built Capacity**
  - o The maximum transmission capacity that can be built in a period in four columns:
    - Index: From Node
    - Index: To Node
    - Index: Period
    - Value: Transmission Maximum Built Capacity in MW (default: 0)
- **Length**
  - o The length of net transfer capacity between two nodes in three columns:
    - Index: From Node
    - Index: To Node
    - Value: Line length in km (default: 0)
- **Line Type Capital Costs**
  - o This sheet contains the cost per MW-km of investing in a transmission with a given line type in three columns:
    - Index: Line Type
    - Index: Period
    - Value: Type Capital Costs in EUR per MW-km (default: 0)
- **Line Type Fixed OM Cost**
  - o The cost for operation and maintenance of transmission line types in three columns:
    - Index: Line Type
    - Index Period
    - Value: Fixed OM Cost in EUR per MW (default: 0)
- **Initial Capacity**
  - o The initial transmission capacity in a period in four columns:
    - Index: From Node
    - Index: To Node
    - Index: Period
    - Value: Transmission Initial Capacity in MW (default: 0)
- **Lifetime**
  - o The lifetime of transmission Lines in years in three columns:
    - Index: From Node
    - Index: To Node
    - Value: Lifetime in years (default: 40)

3.3 Calculation

Before building the model, calculations are performed with parts of the input data. The calculation procedures are part of the 'Empire.py'-script. The most relevant functions are described here to explain how to the final input data is calculated.

- **prepSceProbab_rule**
  Calculates an equiprobable scenario probability depending on the number of scenarios, $|\Omega|$, per investment period:

$$\pi_\omega = \frac{1}{|\Omega|}$$

- **prepInvCost_rule**
  Calculates investment costs, $InvCost_{p,a}$, per MW for generation-, storage-, or transmission asset a in period p:

$$InvCost_{p,a} = \frac{1 - (1 + \delta)^{-min(\tau(|P|-p+1),\ L_a)}}{1 - \dfrac{1}{1 + \delta}} AnnualCost_{p,a} \cdot 1000$$

Where:

$$AnnualCost_{p,a} = \frac{WACC}{1 + WACC - (1 + WACC)^{1-L_a}} CapCost_{p,a} + FixOMCost_{p,a}$$

In the $InvCost_{p,a}$ calculation, δ is the discount rate, τ is the number of leap years in between each investment period, $|P|$ is the total number of investment periods, and L_a is the lifetime of the asset in years. Note that the discount term makes sure investment costs are not paid for asset lifetime that is not considered by the model horizon $|P|$. The $CapCost_{p,a}$ for any transmission connection is also scaled with its length.

For CCS generators, additional fixed transportation- and storage costs, $FixT\&SCost_{p,a}$, apply to the investment costs to ensure the handling of the captured CO2:

$$AnnualCostCCS_{p,a} = AnnualCost_{p,a} + FixT\&SCost_{p,a} \cdot CO2Rem_{p,a} \cdot \frac{3.6 \cdot CO2_{p,a}}{\varepsilon_a}$$

In the $AnnualCostCCS_{p,a}$ calculation, $CO2Rem_{p,a}$ is the fraction of CO2 removed by CCS generator a, $CO2_{p,a}$ is the CO2 factor in tCO2/GJ (there is 3.6 GJ/MWh), and ε_a is the fuel conversion efficiency of the generator.

- **prepOperationalCostGen_rule**
  Calculates the generator operational costs, $OpCost_{p,g}$, for generator g in period p:

$$OpCost_{p,g} = VarOMCost_g + \frac{3.6}{\varepsilon_g}(FuelCost_{p,g})$$

Where $VarOMCost_g$ is the variable operation and maintenance cost of the generator, ε_g is the fuel efficiency of the generator, $FuelCost_{p,g}$ is the fuel cost in EUR/GJ (there is 3.6 GJ/MWh).

If **EMISSION_CAP=False**, CO2 costs also apply:

$$OpCostCO2_{p,g} = OpCost_{p,g} + \frac{3.6}{\varepsilon_g}(CO2_{p,g} \cdot CO2Price_{p,g})$$

where $CO2_{p,a}$ is the CO2 factor in tCO2/GJ, and $CO2Price_{p,g}$ is the CO2 price in EUR/tCO2.

For CCS generators, additional costs for handling transportation- and storage of CO2, $VarT\&SCost_{p,g}$, apply:

$$OpCostCCS_{p,g} = OpCost_{p,g} + \frac{3.6}{\varepsilon_g}((1 - CO2Rem_{p,g}) \cdot CO2_{p,g} \cdot CO2Price_{p,g}$$
$$+ CO2Rem_{p,g} \cdot CO2_{p,g} \cdot VarT\&SCost_{p,g})$$

- **prepInitialCapacityNodeGen_rule**
  Calculates initial capacity, $InitCap_{n,g,p}$, of generator type g in node n and period p:
  $$InitCap_{n,g,p} = RefInitCap_{n,g}(1 - ScaleInitCap_{n,g,p})$$
  Where $RefInitCap_{n,g}$ is the reference initial capacity and $ScaleInitCap_{n,g,p}$ is the share of the reference initial capacity that has retired in that period.

- **prepSload_rule**
  Calculates the future electricity load, $\xi^{\text{load}}_{n,h,i,\omega}$, in node n, operational time step h, scenario ω, and period i:
  $$\xi^{\text{load}}_{n,h,i,\omega} = \xi^{\text{load}}_{n,h,1,\omega} - \xi^{\text{load,avg}}_{n,1,\omega} + \xi^{\text{dem,avg}}_{n,i}$$
  Where $\xi^{\text{load}}_{n,h,1,\omega}$ is the reference load, $\xi^{\text{load,avg}}_{n,1,\omega}$ is the average reference load for all operational time steps in scenario ω and period i, and $\xi^{\text{dem,avg}}_{n,i}$ is the average demand per operational time step based on the future estimate for annual electricity demand (see Section 3.2.3).

4 Scenario generation

This section explains the scenario generation routine implemented for the OpenEMPIRE. The routine is written in the 'scenario_random.py'-script, and it samples chronological time steps from the raw data in the 'ScenarioData' folder in the version-folder in the 'Data handling' folder for the version to be solved. The routine is activated by choosing **scenariogeneration = True** in the 'run.py'-script.

The scenario generation routine works as follows:

Algorithm 1: Basic scenario generation routine for EMPIRE

$\Xi \leftarrow$ Stochastic processes;
$D_\xi \leftarrow$ Historical realizations of the stochastic process $\xi \in \Xi$;
$S \leftarrow$ Seasons/partitions of a year;
$\mathcal{I} \leftarrow$ Investment periods;
$\Omega \leftarrow$ Scenarios per investment period;
$l^{\text{reg}} \leftarrow$ Duration of regular representative time series;
$l^{\text{peak}} \leftarrow$ Duration of peak representative time series;
forall $i \in \mathcal{I}$ **do**
    forall $\omega \in \Omega$ **do**
        $y \leftarrow$ random year in input data;
        Sample regular representative time series;
        forall $s \in S$ **do**
            $h \leftarrow$ random hour in season s and year y;
            forall $\xi \in \Xi$ **do**
                $X_{\xi,i,\omega,s} \leftarrow$ sample l^{reg} consecutive hours from D_ξ starting on hour h;
            end
        end
        Sample peak representative time series;
        $h^{\text{max}} \leftarrow$ hour with maximum combined load in year y;
        forall $\xi \in \Xi$ **do**
            $X^{\text{peak1}}_{\xi,i,\omega} \leftarrow$ sample l^{peak} consecutive hours from D_ξ with hour h^{max} in the middle;
        end
        $h^{\text{node-max}} \leftarrow$ hour with maximum load of a single node in year y;
        forall $\xi \in \Xi$ **do**
            $X^{\text{peak2}}_{\xi,i,\omega} \leftarrow$ sample l^{peak} consecutive hours from D_ξ with hour $h^{\text{node-max}}$ in the middle;
        end
    end
end

There are six stochastic processes that are realized in each scenario generated by the scenario generation routine:

- The hourly availability of solar, wind onshore, wind offshore, and hydro run-of-river plants (four processes)
- The seasonal availability of hydro regulated (one process)
- The electric load profile (one process)

The six processes have a new realization for every operational time step, every investment period, every node, and every scenario. The new realizations are ensured by randomly picking a year to sample data for the stochastic processes. Hours are also picked random within the random year to construct representative time periods. There are more data for solar and wind than for load and hydro, hence the random year selection is separate for solar and wind (three stochastic processes) and hydro and load (three stochastic processes).

Prior to the sampling, the scenario generation routine makes four equal partitions of each year (three months each), and each partition represents a regular season that is realized for every scenario. Each regular season has a duration of one complete week, i.e. $7 \cdot 24 = 168$ representative hours.

In addition, there are two peak seasons, peak1 and peak2, that are also part of every scenario. The first season is sampled based on the highest combined load of all nodes in the randomly selected year. The second season is sampled based on the highest load of a single node in the randomly selected year. Since there is a limited number of years to sample from, peak seasons are likely to be the same in different scenarios and investment periods.

5 Output Data

5.1 Output Files

EMPIRE's standard output encompasses nine .csv files Each file contains one or several tables with figures ordered by indices, e.g. year, technology, country, etc. This section describes the content of the result files.

5.1.1 results_objective.csv

Prints the objective function value. Also available in the logfile.

5.1.2 results_output_curtailed_prod.csv

Curtailed production shows the expected amount of curtailed energy from variable renewable energy sources per year in GWh in four columns:

- Index: Node
- Index: RESGeneratorType
- Index: Period
- Value: ExpectedAnnualCurtailment_GWh

5.1.3 results_output_EuropePlot.csv

Results for purposes of easy and fast visualization. There are five tables in this file:

- *genInstalledCap_MW*
  Installed capacity for the whole Europe, per investment period and generator type:
  - Rows: Period
  - Columns: Generator types
  - Value: Installed Capacity in MW for all nodes
- *genExpectedAnnualProduction_GWh*
  Expected annual energy production for the whole Europe, per investment period and generator type:
  - Rows: Period
  - Columns: Generator types
  - Value: Expected annual electricity output in GWh for all nodes
- *storPWInstalledCap_MW*
  Installed storage capacityfor the whole Europe, per investment period and storage type:
  - Rows: Period
  - Columns: Storage types
  - Value: Installed charging/discharging capacity in MW for all nodes
- *storENInstalledCap_MW*
  Installed energy storage capacity for the whole Europe, per investment period and storage type:
  - Rows: Period
  - Columns: Storage types
  - Value: Installed energy storage capacity in MW for all nodes

- *storExpectedAnnualDischarge_GWh*

  Expected annual storage discharge for the whole Europe, per investment period and storage type:
  - o Rows: Period
  - o Columns: Storage types
  - o Value: Expected annual discharge volume in GWh for all nodes

5.1.4 results_output_EuropeSummary.csv

Results for Europe as a whole. There are three tables:

- *Europe-wide general values per year scenario:*
  - o Index: Period
  - o Index: Scenario
  - o Value: Annual CO2emissions in Tons
  - o Value: CO2 shadow price in EUR/ton (if EMISSION_CAP=True)
  - o Value: CO2 cap in Tons
  - o Value: Total annual generation in GWh
  - o Value: Average CO2-factor in ton/MWh
  - o Value: Average undiscounted hourly electricity price in EUR/MWh
  - o Value: Annual curtailment of solar, wind, and hydro run-of-river in GWh
  - o Value: Annual losses from storage handling in GWh
  - o Value: Annual losses from transmission in GWh
- *Europe-wide generation values per year and type:*
  - o Index: Generator type
  - o Index: Period
  - o Value: Capacity built over the investment period in MW
  - o Value: Capacity installed in the investment period in MW
  - o Value: Discounted investment cost in EUR/MW
  - o Value: Expected annual production in GWh
- *Europe-wide storage values per year and type:*
  - o Index: Storage type
  - o Index: Period
  - o Value: Charging/discharging capacity built over the investment period in MW
  - o Value: Charging/discharging capacity installed in the investment period in MW
  - o Value: Energy storage capacity built over the investment period in MW
  - o Value: Energy storage capacity installed in the investment period in MW
  - o Value: Discounted investment cost in EUR/MW-MWh
  - o Value: Expected annual discharge in GWh

5.1.5 results_output_gen.csv

Results for generation by country, type, and year:

- Index: Node
- Index: Generator type
- Index: Period

- Value: Capacity built in the node over the investment period in MW
- Value: Capacity installed in the node in the investment period in MW
- Value: Discounted investment cost in EUR/MW
- Value: Expected annual production in the node in GWh

5.1.6   results_output_Operational.csv

Results on the hourly dispatch of electricity among all the nodes, periods, and scenarios:

- Index: Node
- Index: Period
- Index: Scenario
- Index: Season
- Index: Hour
- Value: All generation in MWh/h
- Value: Original load in MWh/h
- Value: Load after considering storage and transmission handling in MWh/h
- Value: Production from 'generator 1' in MWh/h
- Value: Production from 'generator 2' in MWh/h
- (…)
- Value: Production from 'generator N' in MWh/h
- Value: Charging of all storage in the node in MWh/h
- Value: Discharging of all storage in the node in MWh/h
- Value: Energy stored in all storage in the node in MWh
- Value: Losses from storage handling in MWh/h
- Value: Sum of transmission out of the node in MWh/h
- Value: Sum of transmission into the node in MWh/h
- Value: Losses from transmission in MWh/h
- Value: Load shed in MWh/h
- Value: Undiscounted hourly shadow price of electricity in EUR/MWh/h
- Value: Average CO_2 intensity on the production in the node in $kgCO_2$/MWh

5.1.7   results_output_stor.csv

Results for storage by country, type, and year:

- Index: Node
- Index: Storage type
- Index: Period
- Value: Charging/discharging capacity built in the node over the investment period in MW
- Value: Charging/discharging capacity installed in the node in the investment period in MW
- Value: Energy storage capacity built in the node over the investment period in MW
- Value: Energy storage capacity installed in the node in the investment period in MW
- Value: Discounted investment cost in EUR/MW-MWh
- Value: Expected annual discharge in the node in GWh
- Value: Expected annual storage handling losses in GWh

5.1.8   results_output_transmision.csv

Results for transmission by transmission connection and year:

- Index: Node 1
- Index: Node 2
- Index: Period
- Value: Capacity built in the transmission link over the investment period in MW
- Value: Capacity installed in the transmission link in the investment period in MW
- Value: Discounted investment cost in EUR/MW
- Value: Expected annual transmission volume in GWh
- Value: Expected annual transmission losses in GWh

5.1.9   results_output_transmision_operational.csv

Results on the hourly dispatch of transmission links among all the nodes, periods, and scenarios:

- Index: Node 1
- Index: Node 2
- Index: Period
- Index: Season
- Index: Scenario
- Index: Hour
- Value: Transmission received from Node 1 to Node 2
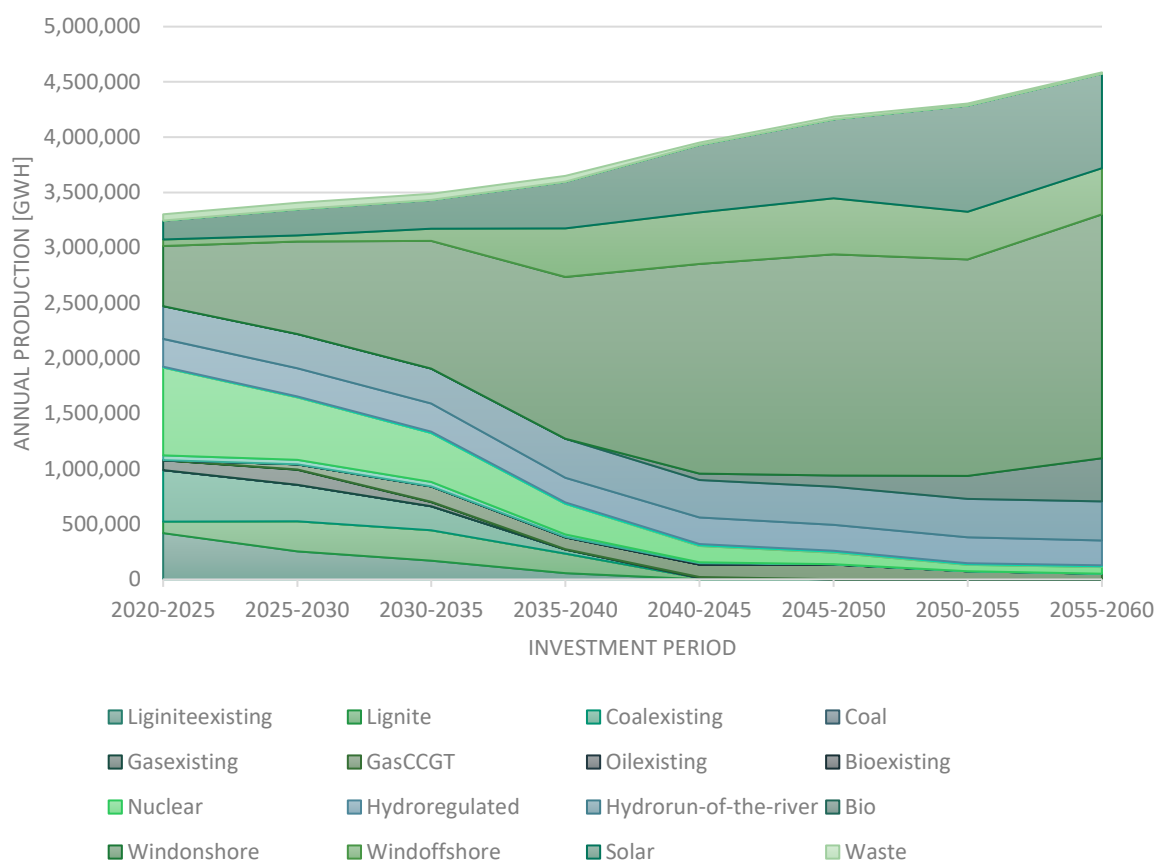- Value: Transmission losses from Node 1 to Node 2

5.2 Typical EMPIRE charts

From most of the output files, meaningful charts can be created. Typical charts for most files are below.
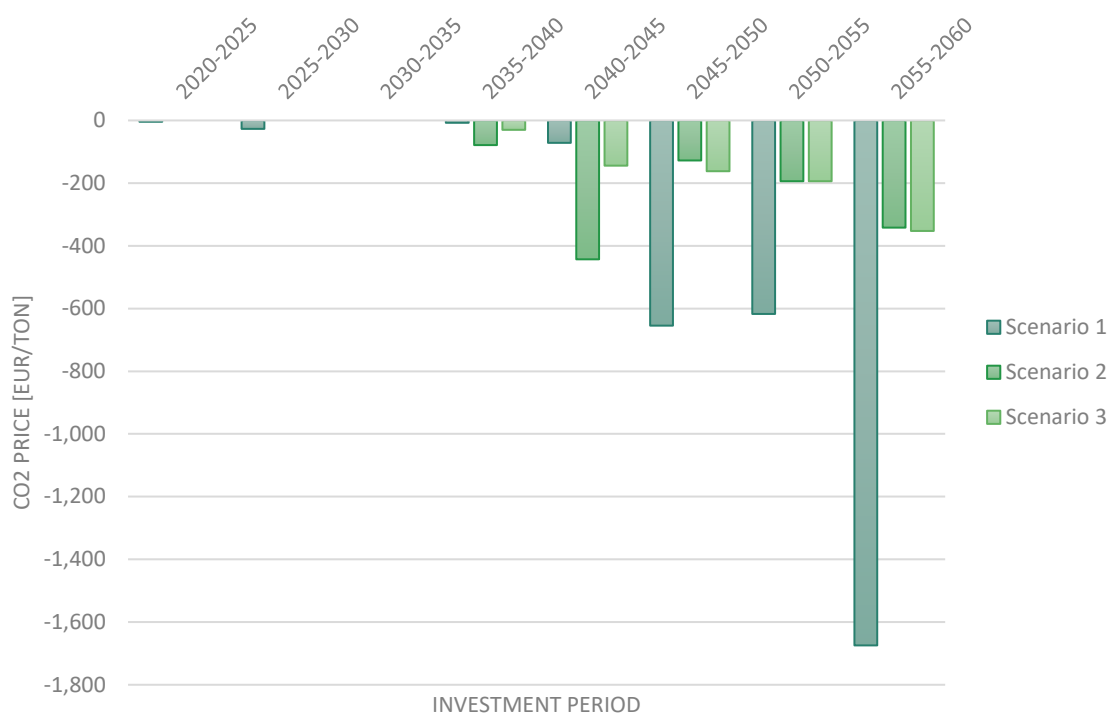
5.2.1 results_output_curtailed_prod.csv
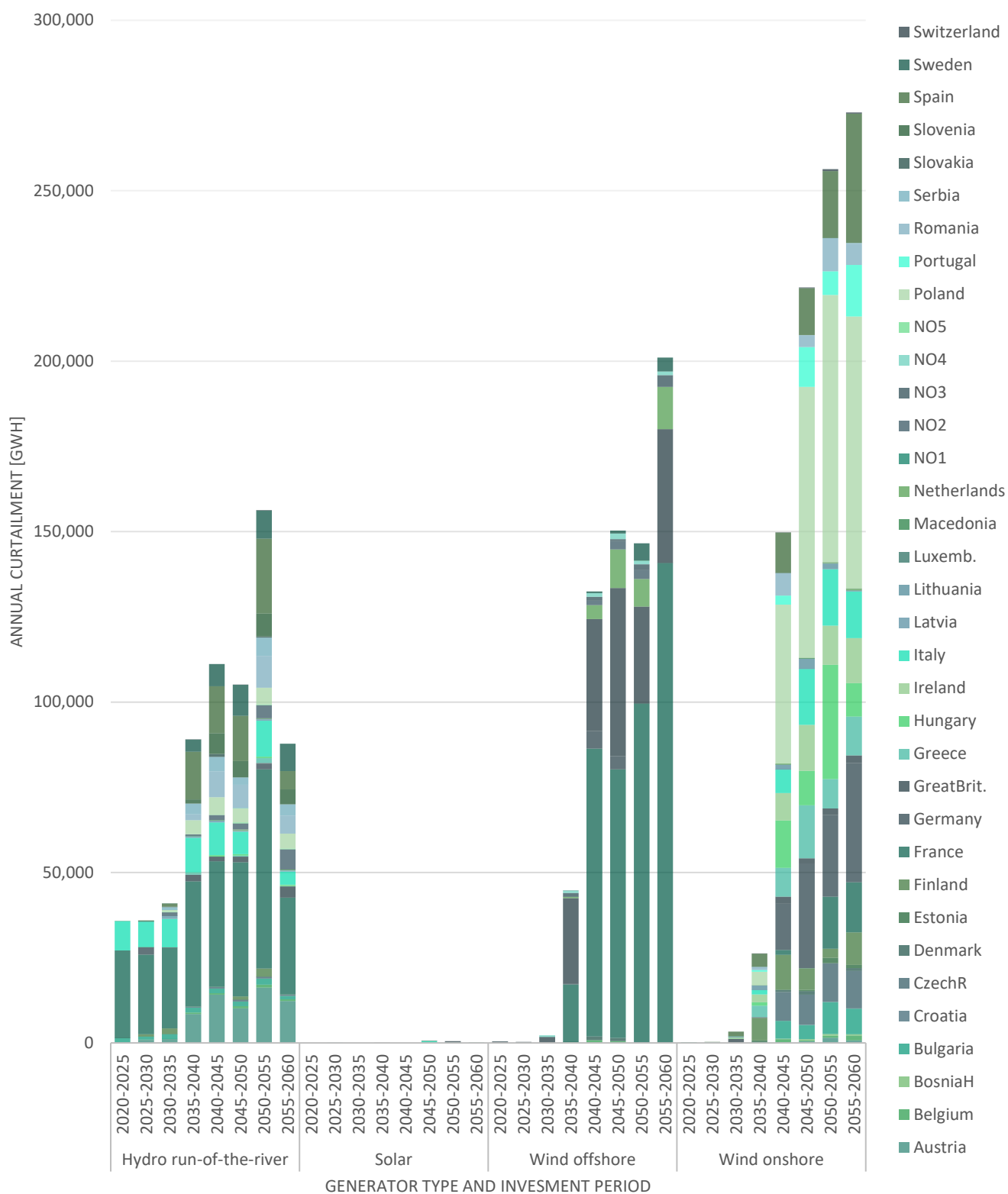
5.2.2   results_output_EuropePlot.csv

5.2.3   results_output_EuropeSummary.csv

5.2.4   results_output_transmision.csv

EMPIRE can provide the necessary data to geographically plot the intra-European energy transmission paths and capacities. In the figure below, each transmission link has its own thickness equivalent to the expected annual capacity expansion for all investment periods.