DispaSET Documentation

Release 2.4.post177+g264e619

Sylvain Quoilin

Jan 14, 2022

Contents

The Dispa-SET model is an open-source unit commitment and optimal dispatch model focused on the balancing and flexibility problems in European grids. Its pre and post-processing tools are written in Python and the main solver can be called via GAMS.

Dispa-SET is mainly developed within the Joint Research Centre of the EU Commission, in close collaboration with the University of Liège and the KU Leuven (Belgium).

CHAPTER 1

Downloading Dispa-SET

The public version of Dispa-SET can be downloaded in the *Releases* section or from its github repository (using the Clone or Download button on the right side of the screen): https://github.com/energy-modelling-toolkit/Dispa-SET

CHAPTER 2

How to cite

Depending on the version that was used, one of the following JRC technical reports can be selected to cite Dispa-SET:

- Kavvadias, K., Hidalgo Gonzalez, I., Zucker, A. and Quoilin, S., Integrated modelling of future EU power and heat systems: The Dispa-SET v2.2 open-source model, JRC Technical Report, EU Commission, 2018
- Quoilin, S., Hidalgo Gonzalez, I. and Zucker, A., Modelling Future EU Power Systems Under High Shares of Renewables: The Dispa-SET 2.1 open-source model, JRC Technical Report, EU Commission, 2017
- Hidalgo González, I., Quoilin, S. and Zucker, A., Dispa-SET 2.0: unit commitment and power dispatch model, Tech. rep., Publications Office of the European Union, 2014.

CHAPTER 3

Documentation

A pdf documentation of the model is available in the 2017 JRC technical report: Modelling Future EU Power Systems Under High Shares of Renewables.

In addition, the lastest model documentation can be obtained by running sphinx in the Docs folder of the project or by consulting the online documentation. This documentation corresponds to the latest available public version of Dispa-SET: http://www.dispaset.eu/latest/index.html

CHAPTER 4

Main contributors:

- Sylvain Quoilin (KU Leuven, Belgium))
- Konstantinos Kavvadias (Joint Research Centre, EU Commission)
- Matija Pavičević (KU Leuven, Belgium)

CHAPTER 5

Contents

5.1 Overview

Organization Joint Research Centre, European Commission

Version 2.4 (2.4.post177+g264e619)

Date Jan 14, 2022

The Dispa-SET model is mainly developed within the "Joint Research Centre" of the European Commission and focuses on the balancing and flexibility problems in European grids[1].

It is written in GAMS and uses csv files for input data handling. The optimisation is defined as a Linear Programming (LP) or Mixed-Integer Linear Programming (MILP) problem, depending on the desired level of accuracy and complexity. Continuous variables include the individual unit dispatched power, the shedded load and the curtailed power generation. The binary variables are the commitment status of each unit. The main model features can be summarized as follows:

5.1.1 Features

- Minimum and maximum power for each unit
- Power plant constraints: minimum power, ramping limits, Minimum up/down times, start-up, no-load costs
- Outages (forced and planned) for each units
- Reserves (spinning & non-spinning) up and down
- Load Shedding
- Curtailment
- Pumped-hydro storage
- Non-dispatchable units (e.g. wind turbines, run-of-river, etc.)

[1] Quoilin, S., Hidalgo Gonzalez, I., & Zucker, A. (2017). Modelling Future EU Power Systems Under High Shares of Renewables: The Dispa-SET 2.1 open-source models. Publications Office of the European Union.

- Multi-nodes with capacity constraints on the lines (congestion)
- Constraints on the targets for renewables and/or CO2 emissions
- CHP power plants and thermal storage
- Power-to-heat (heat pump, electrical heater) and thermal storage
- DSM-ready demand
- Integrated mid-term scheduling and short-term optimal dispatch
- Different model formulations and levels of clustering complexity generated from the same dataset.

The demand is assumed to be inelastic to the price signal. The MILP objective function is therefore the total generation cost over the optimization period.

5.1.2 Libraries used and requirements

- Python 3.7
- pandas for input and result data handling
- matplotlib for plotting
- GAMS_api for the communication with GAMS

the above are auto installed in a conda environment if you follow the instructions of the Quick start.

5.1.3 Dispa-SET in the scientific literature

In the past years, Dispa-SET has been used in various scientific works covering different geographical areas and with different focus points. The works for which scientific articles have been published are summarized hereunder:

- Hydropower for flexibility services in the European power system[2].
- Generating stylized flexibility constraints for the JRC-EU-TIMES model[3][4].
- Impact of Electric Vehicle deployment in The Netherlands[5].
- Open-source model of the Balkans area, with some simulations involving high shares of renewables[6][7].
- Specific country studies for RES integration (Belgium, Greece)[3][9]

[2] Sánchez Pérez, A. (2017), Modelling Hydropower in detail to assess its contribution to flexibility services in the European power system. Master Thesis, University of Utrecht, Netherlands.

[3] Quoilin, S., Nijs, W., Gonzalez, I. H., Zucker, A. and Thiel, C. (2015), Evaluation of simplified flexibility evaluation tools using a unit commitment model, In 12th International Conference on the European Energy Market (EEM), pp. 1 5.

[4] Quoilin, S., Nijs, W. and Zucker, A. (2017), Evaluating flexibility and adequacy in future EU power systems: model coupling and long-term forecasting, In Proceedings of the 2017 ECOS Conference, San Diego.

[5] Beltramo, A., Julea, A., Refa, N., Drossinos, Y., Thiel, C. and Quoilin, S. (2017),'Using electric vehicles as flexible resource in power systems: A case study in the Netherlands, In 14th International Conference on the European Energy Market (EEM).

[6] Pavičević, M., Tomić, I., Quoilin, S., Zucker, A. and Pukšec, T. and Krajačić, G. (2017), Applying the Dispa-SET model on the Western Balkans power systems, In Proceedings of the 2017 SDEWES Conference

[7] Tomić, I., Pavičević, M., Quoilin, S., Zucker, A., Krajačić, G., Pukšec, T. and Duić, N. (2017), Applying the Dispa-SET model on the seven countries from the South East Europe, In 8th Energy Planning and Modeling of Energy Systems-Meeting, Belgrade

[9] Ricardo Fernandez Blanco Carramolino, Konstantinos Kavvadias, I Hidalgo Gonzalez (2017). Quantifying the water-power linkage on hydrothermal power systems: A Greek case study. Applied Energy.

- Comparison between model formulations and levels of clustering[10][11][20]
- Benders decomposition for capacity expansion[14]
- The water-energy nexus in Greece and in Africa[8][9][15]
- Soft-linking between JRC-EU-TIMES and Dispa-SET at the EU level[17]
- Quantifying the flexibility provided by coupling the heating and power sectors[12][13][18][19][16]
- Power systems adequacy and flexibility assessments in developing countrie (Africa, Bolivia)[15][21]

5.1.4 Ongoing developments

The Dispa-SET project is relatively recent, and a number of improvements will be brought to the project in a close future:

- Grid constraints (DC power-flow)
- Stochastic scenarios
- Modelling of investment and capacity expansion
- Modeling of the ancillary markets

5.1.5 Licence

Dispa-SET is a free software licensed under the "European Union Public Licence" EUPL v1.2. It can be redistributed and/or modified under the terms of this license.

[10] Pavičević, M., Quoilin, S. and Pukšec, T., (2018). Comparison of Different Power Plant Clustering Approaches for Modeling Future Power Systems, Proceedings of the 3rd SEE SDEWES Conference, Novi Sad.

[11] Pavičević, M., Kavvadias, K. and Quoilin, S. (2018). Impact of model formulation on power system simulations - Example with the Dispa-SET Balkans model, EMP-E conference 2018: Modelling Clean Energy Pathways, Brussels.

[20] Pavičević, M., Kavvadias, K., Pukšec, T., & Quoilin, S. (2019, June). Comparison of different model formulations for modelling future power systems with high shares of renewables – The Dispa-SET Balkans model. Applied Energy.

[14] Matthias Zech, Acceleration strategies of the Generation Expansion Planning problem using Benders Decomposition, Master Thesis, Dresden University of Technology, 2018

[8] Ricardo Fernandez Blanco Carramolino, Konstantinos Kavvadias, Ignacio Hidalgo Gonzalez (2017). Water-related modelling in electric power systems: WATERFLEX Exploratory Research Project.

[15] Matteo De Felice, Iratxe Gonzalez-Aparicio, Thomas Huld, Sebastian Busch, Ignacio Hidalgo-Gonzalez . Analysis of the water-power nexus in the West African power pool. JRC Technical Report, 2019.

[17] Matija Pavičević, Wouter Nijs, Konstantinos Kavvadias, Sylvain Quoilin (2019). Modelling flexible power demand and supply in the EU power system: soft-linking between JRC-EU-TIMES and the open-source Dispa-SET model. Proceedings of the 32nd International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems.

[12] Juan Pablo Jiménez Navarro, Konstantinos Kavvadias, Sylvain Quoilin, Zucker Andreas (2018). The joint effect of centralised cogeneration plants and thermal storage on the efficiency and cost of the power system. Energy.

[13] Kavvadias, K., Jimenez Navarro, J.-P., Zucker, A., & Quoilin, S. (2018). Case study on the impact of cogeneration and thermal storage on the flexibility of the power system (KJ-NA-29082-EN-N). Netherlands: Publication Office of the European Commission.

[18] Konstantinos Kavvadias, Georg Thomassen, Matija Pavičević, Sylvain Quoilin (2019). Electrifying the heating sector in Europe: The impact on the power sector. Proceedings of the 32nd International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems.

[19] Konstantinos Kavvadias, Juan Pablo Jimenez Navarro, Georg Thomassen (2019). Decarbonising the EU heating sector: Integration of the power and heating sector.

[16] Matija Pavičević, Juan-Pablo Jimenez, Konstantinos Kavvadias, Sylvain Quoilin (2019). Modeling the flexibility offered by coupling the heating sector and the power sector: an assessment at the EU level. 5th International Conference On Smart Energy Systems.

[21] Rojas Candia, R., Balderrama Subieta, S. L., Adhemar Araoz Ramos, J., Vicente Senosiain, M., Peña Balderrama, G., Jaldín Florero, H., & Quoilin, S. (2019). Techno-economic assessment of high variable renewable energy penetration in the Bolivian interconnected electric system. International Journal of Sustainable Energy Planning and Management, 22.

5.1.6 Main Developers

- Sylvain Quoilin (University of Liège, KU Leuven)
- Konstantinos Kavvadias (European Commission, Joint Research Centre)
- Matija Pavičević (KU Leuven, Belgium)

5.1.7 References

5.2 Releases

Major stable releases:

- Dispa-SET v2.4
- Dispa-SET v2.3
- Dispa-SET v2.2
- Dispa-SET v2.1
- Dispa-SET v2.0

5.2.1 Changelog

Version 2.x

- **Variable time step**
  - The pre-processing and the GAMS file have been update to handle different time steps (not only one hour)
  - This is currently restricted to three time steps: 15min, 1h, 24h
  - The input data whose time step is lower than the desired one is averaged

Version 2.4

- **Mid-term scheduling**
  - The yearly storage level profiles can now be calculated internally (i.e. without providing exogenous profiles).
  - A first, simplified version of dispa-set is run over a whole year to generate these profiles during the pre-processing phase
  - This option is activated in the config file and is transparent for the user.
- **Flexible Demand:**
  - To model demand-side management, it is now possible to define a share of the demand curve as "flexible"
  - In this flexible demand, the load can be shifted from one hour to the other
  - The maximum flexibility is characterized by the equivalent number of storage hours for the shifted load, which is defined as parameter in the configuration file.

- **Power-to-heat units**
  - P2HT units (heat pumps, electrical heater) have now been added
  - They are coupled to a heat demand and possibly to a thermal storage capacity
  - COP can be defined as temperature-dependent. An additional input with temperature times for each zone has been defined.
- Transmission prices have been added to the pre-processing and can now be fully parametrized
- Fuel Prices can now be country-specific
- Input data in the csv files can now be defined with time stamps from any year or with a numerical index
- **Post-processing:**
  - Improved dispatch plot with shifted, shed loads and electricity consumption from P2HT units
  - Storage levels are now differentiated by technology
- **Miscellaneous:**
  - Multiple bug fixes, code improvement and usability improvement.
  - All config files and the example scripts have been checked and cleaned
  - New formulation of the clustering function with significant simulation time improvements
  - The Pyomo version of Dispa-SET has now been removed since it was no longer up-to-date
  - The end-of horizon reservoir level is no longer a firm constraint. A water value can be defined to impose a price on the unmet level requirements.
  - Excel configuration files are now subject to versioning, which ensures backward compatibility with older configuration files.
  - Countries are now renamed into “zones” in all API functions.
  - The option to cache csv file data when loading has been removed
  - Implemented a more robust versioning system

Version 2.3

- **Input Data:**
  - A complete EU dataset has been included to the repository for the year 2016.
  - More information: *Dispa-SET for the EU28*.
- **Reformulation of the reserve constraints:**
  - Secondary reserves are now covered by spinning units only.
  - Tertiary reserves can also be covered by quick start units.
  - In total, three different reserve markets are now considered: Secondary up; Secondary down; and Tertiary up
- **Implementation of a new formulation (integer clustering) for power plant related constraints. This formulation divides th**
  - Standard formulation: low capacity or highly flexible units are merged
  - No clustring: all units are considered individually
  - LP clustering: all units are aggregated by technology and binary constraints are removed

  - Integer clustering: a representative unit is considered for each technology and multiplied N times.
- **Improved pre-processing:**
  - Improved log message during input data checks
  - New config files to test the different clustering methods
  - Added functions to perform parametric studies
  - Example scripts for Monte Carlo analyses using lating hypercube samplings
- **Improved post-processing:**
  - Netting interconnections in dispatch plots
  - New colour palette and polished dispatch plot
  - New fuels included
  - Improved representation of curtailment
- **External dependencies:**
  - Removed pre-compiled libraries for unix systems
  - Use of the low-level GAMS API (https://github.com/kavvkon/gams-api)
- **Python 3.7:**
  - Dispa-SET now runs exclusively on Python 3.7.
  - The compatibility with previous Python versions (2.7, 3.6) is not guaranteed anymore.
- **Miscellaneous:**
  - Unit tests on travis (https://travis-ci.org/energy-modelling-toolkit/Dispa-SET)
  - Bug fixes

Version 2.2

- Inclusion of CHP, power2heat and thermal storage (these new features can be tested by running the config file for Cyprus: 'ConfigFiles/ConfigCY.xlsx')
- Bug fixes
- Improved user interface

Version 2.1

- Major refactoring of the folder structure
- New data included in the database
- Inclusion of the LP formulation (in addition to the MILP)

Version 2.0

First public version of the Dispa-SET model.

5.3 Getting Started

This short tutorial describes the main steps to get a practical example of Dispa-SET running.

5.3.1 Prerequisites

Install Python 3.7, with full scientific stack. The Anaconda distribution is recommended since it comprises all the required packages. If Anaconda is not used, the following libraries and their dependencies should be installed manually:

- future >= 0.15
- click >= 3.3
- numpy>=1.10
- scipy>=0.15
- matplotlib>=1.5.1
- pandas>= 0.19
- xlrd >= 0.9
- pickle
- pyyaml >= 5.1
- pytest

Using Dispa-SET:

Dispa-SET is primarily designed to run with GAMS and therefore requires GAMS to be installed with a valid user licence. Currently, only the 64-bit version of GAMS is supported in Dispa-SET!

The GAMS api for python has been pre-compiled in the "Externals" folder and is usable for Windows 64 bit systems. If the pre-compiled binaries are not available or could not be loaded, they must be installed manually using following command in the Anaconda prompt:

```
pip install gdxcc gamsxcc optcc
```

Alternatively, the gams python api can also be compiled from the source provided in the GAMS installation folder (e.g. "C:\GAMS\win64\24.3\apifiles\Python\api"):

```
python setup.py install
```

NB: For Windows users, the manual api compilation might require the installation of a C++ compiler for Python. This corresponds to the typical error message: "Unable to find vcvarsall.bat". This can be solved by installing the freely available "Microsoft Visual C++ Compiler for Python". In some cases the path to the compiler must be added to the PATH windows environment variable (e.g. C:Program FilesCommon FilesMicrosoftVisual C++ for Python9.0)

The api requires the path to the gams installation folder. The "get_gams_path()" function of dispa-set performs a system search to automatically detect this path. It case it is not successful, the user is prompted for the proper installation path.

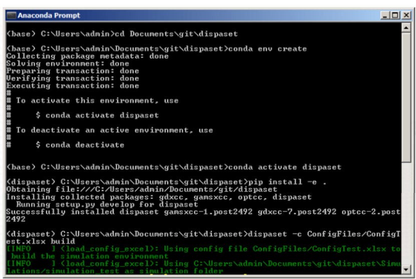
5.3.2 Step-by-step example of a Dispa-SET run

This section describes the pre-processing and the solving phases of a Dispa-SET run. Three equivalent methods are described in the next sections:

- Using the command line interface
- Using the Dispa-SET API
- Using GAMS

1. Using the command line interface

Dispa-SET can be run from the command line. To that aim, open a terminal window and change de directory to the Dispa-SET root folder.



1.0. Install Dispa-SET and the required dependencies

Use the following commands in a terminal (Anaconda prompt in Windows):

```
conda env create  # Automatically creates environment based on environment.yml
conda activate dispaset
pip install -e . # Install editable local version
```

The above commands create a dedicated environment so that your anconda configuration remains clean from the required dependencies installed. If preferred, the Gams libraries can also be installed without creating a dedicated environment. In that case, replace the above commands with these ones:

```
pip install gamsxcc gdxcc optcc
python setup.py install
```

To check that everything runs fine, you can build and run a test case by typing:

```
dispaset -c ConfigFiles/ConfigTest.xlsx build simulate
```

1.1. Check the configuration file

Dispa-SET runs are defined in dedicated excel configuration files stored in the "ConfigFiles" folder. The configuration file "ConfigTest.xlsx" is provided for testing purposes. It generates a 10-days optimisation using data relative a fictitious power system composed of two zones Z1 and Z2.

1.2. Pre-processing

From the command line, specify the configuration file to be used as an argument and the actions to be performed. Within the "Dispa-SET" folder, run:

```
dispaset -c ./ConfigFiles/ConfigTest.xlsx build
```

1.3. Check the simulation environment

The simulation environment folder is defined in the configuration file. In the test example it is set to "Simulations/simulation_test". The simulation inputs are written in three different formats: excel (34 excel files), Python (Inputs.p) and GAMS (Inputs.gdx).

1.4. Run the optimisation

The simulation can be started directly from the main DispaSet python file after the pre-processing phase. From the "Dispa-SET" folder, run:

```
dispaset -c ./ConfigFiles/ConfigTest.xlsx simulate
```

This runs the optimisation, and stores the results in the same folder. Note that this can only work is the simulation has been pre-processed before (step 1.2). It is possible to combine the pre-processing and simulation step in one command:

```
dispaset -c ./ConfigFiles/ConfigTest.xlsx build simulate
```

2. Using the Dispa-SET API.

The steps to run a model can be also performed directly in python, by importing the Dispa-SET library. An example file ("build_and_run.py") is available in the "scripts/" folder.

To run the commands below, the Gams libraries are required. Install them using the following command in an Anaconda prompt:

```
pip install gamsxcc gdxcc optcc
```

After checking the configuration file "ConfigTest.xlsx" (in the "ConfigFiles" folder). Run the following python commands:

2.1 Import Dispa-SET:

```
import dispaset as ds
```

2.2 Load the configuration file:

```
config = ds.load_config_excel('ConfigFiles/ConfigTest.xlsx')
```

2.3 Build the simulation environment (Folder that contains the input data and the simulation files required for the solver):

```
SimData = ds.build_simulation(config)
```

2.4 Solve using GAMS:

```
r = ds.solve_GAMS(config['SimulationDirectory'], config['GAMS_folder'])
```

A more detailed description of the Dispa-SET functions in available in the API section.

3. Using GAMS

It is sometimes useful to run the dispa-SET directly in GAMS (e.g. for debugging purposes). In that case, the preprocessing must be run first (steps 1.2 or 2.1, 2.2 and 2.3) and the gams file generated in the simulation folder can be used to run the optimization.

Using the GAMS graphical user interface:

From the simulation folder (defined in the config file), the Dispa-SET model can be run following the instruction below:

1. Open the UCM.gpr project file in GAMS
2. From GAMS, open the UCM_h.gmx model file
3. Run the model in GAMS.

The result file is written in the gdx format and stored in the Simulation folder, together with all input files.

Using the GAMS command line:

GAMS can also be run from the command line (this is the only option for the Linux version).

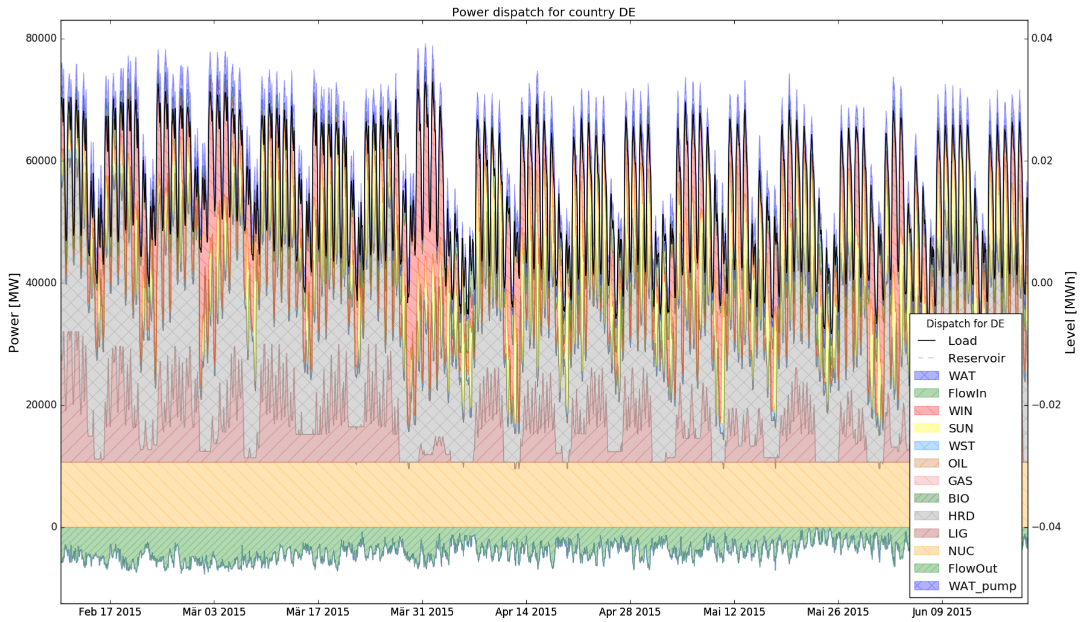
1. Make sure that the gams binary is in the system PATH
2. From the simulation environment folder, run:

   ```
   gams UCM_h.gms
   ```

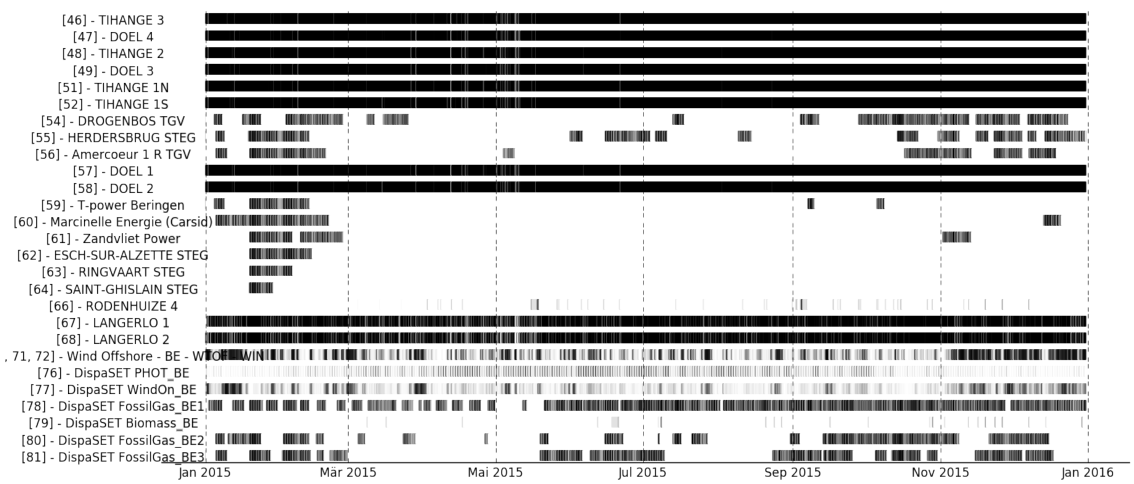
5.3.3 Postprocessing and result display

Various functions and tools are provided within the PostProcessing.py file to load, analyse and plot the siimulation results. The use of these functions is illustrated into the the “Read_results_notebook.ipynb” Notebook or in the “read_results.py” script, which can be run by changing the path to the simulation folder. The type of results provided by the post-processing is illustrated hereunder.

The power dispatch can be plotted for each simulated zone. In this plot, the units are aggregated by fuel type. The power consumed by storage units and the exportations are indicated as negative values.



It is also interesting to display the results at the unit level to gain deeper insights regarding the dispatch. In that case, a plot is generated, showing the commitment status of all units in a zone at each timestep. Both the dispatch plot and the commitment plot can be called using the CountryPlots function.



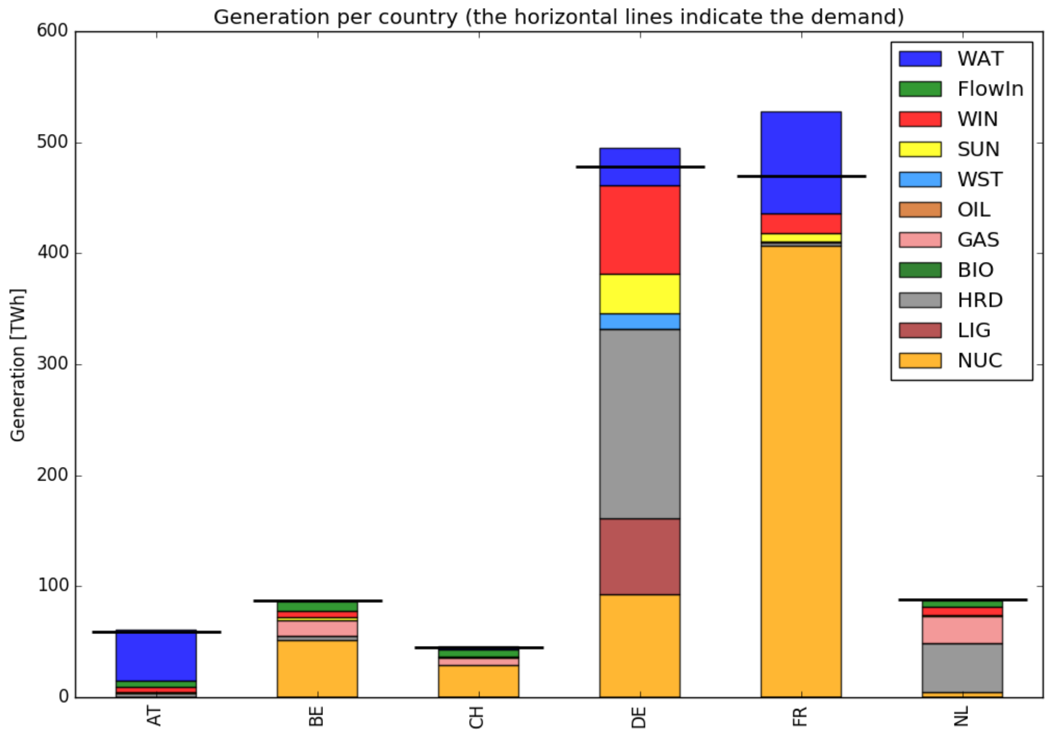
Some aggregated statistics on the simulations results can also be obtained, including the number of hours of congestion in each interconnection line, the yearly energy balances for each zone, the amount of lost load, etc.

```
Aggregated statistics for the considered area:
Total consumption:1227.07310992 TWh
Peak load:203182.461067 MW
Net importations:-42.20072928 TWh

Country-Specific values (in TWh or in MW):
        Demand      PeakLoad  NetImports  LoadShedding  Curtailment
AT   59.375448  10144.000000    5.144132           NaN          NaN
BE   86.971154  13632.250000    8.911190           NaN          NaN
CH   44.694098   7794.262468    7.199527           NaN          NaN
DE  478.030824  76212.250000  -17.260122           NaN          NaN
FR  470.075612  90588.000000  -51.878128           NaN          NaN
NL   87.925973  16285.500000    5.682672           NaN          NaN

Number of hours of congestion on each line:
{'AT -> CH': 5917,
 'AT -> DE': 430,
 'BE -> FR': 62,
 'BE -> NL': 344,
 'CH -> AT': 720,
 'CH -> DE': 15,
 'CH -> FR': 56,
 'DE -> AT': 1522,
 'DE -> CH': 4378,
 'DE -> NL': 2803,
 'FR -> BE': 2689,
 'FR -> CH': 7665,
 'NL -> BE': 1403,
 'NL -> DE': 60}
```

The yearly energy balance per fuel or per technology is also useful to compare the energy mix in each zone. This can be plotted using the EnergyBarPlot function, with the following results:



5.4 Input Data

In this section, "Input Data" refers to the data stored in the Dispa-SET database. The format of this data is pre-defined and imposed, in such a way that it can be read by the pre-processing tool.

Two important preliminary comments should be formulated:

- All the time series should be registered with their timestamps (e.g. '2013-02-20 02:00:00') or with a numerical index. Dispa-SET will issue an error if the day is located before the month. It is also advised to remove all time zone information from the time stamps. If the index is an integer, Dispa-SET will only recognize it if contains 8760 elements (one full year) or if it has exactly the same length as the simulation horizon.
- Although the optimisation model is designed to run with any technology or fuel name, the pre-processing and the post-processing tools of Dispa-SET use some hard-coded values. The Dispa-SET database should also comply with this convention (described in the next sections). Any non-recognized technology or fuel will be discarded in the pre-processing.

5.4.1 General simulation parameters

A number of simulation options and parameters need to be defined in the configuration file. In order to obtain default values and a complete description of the options, it is commended to open the ConfigTest.xlsx configuration file, which is always kept up-to-date.

The options to be filled are summarized hereunder.

Table 5.1: Dispa-SET Simulation Options

General Options	Description
SimulationDirectory	Folder with all simulation files and input data
WriteGDX	Write the inputs in a GDX file (required for gams)
WritePickle	Write the inputs to a pickle file
GAMS_folder	Path the GAMS installation folder
cplex_path	Path to the cplex folder
Horizon Settings	
StartDate	Start date of the simulation
StopDate	End data of the simulation
HorizonLength	Simulation horizon length in days
Look ahead	Overlap period in days
DataTimeStep	Time step of the date in the csv files
SimulationTimeStep	Time step for the simulation
Simulation Options	
SimulationType	Stanard/LP/LP clustered/Integer clustering
ReserveCalculation	Generic (only available option for now)
AllowCurtailment	True/False
Mid-term scheduling	
HydroScheduling	Off/Zonal/Regional
HydroSchedulingHorizon	"Annual"/"Stop-date driven"
InitialFinalReservoirLevel	True/False (if False, use StorageProfile)
ReservoirLevelInitial	Initial res. level if the above option is true
ReservoirLevelFinal	Fainl reservoir level if the above option is true

5.4.2 Technologies

The Dispa-SET input distinguishes between the technologies defined in the table below. The VRES column indicates the variable renewable technologies (set "tr" in the optimisation) and the Storage column indicates the technologies which can accumulate energy.

Table 5.2: Dispa-SET technologies

Technology	Description	VRES	Storage
Power only			
HDAM	Conventional hydro dam	N	Y
HROR	Hydro run-of-river	Y	N
HPHS	Pumped hydro storage	N	Y
PHOT	Solar photovoltaic	Y	N
WAVE	Wave energy	Y	N
WHEN	Waste heat engine	N	N
WTOF	Offshore wind turbine	Y	N
WTON	Onshore wind turbine	Y	N
Combined heat and power			
COMC	Combined cycle	N	N
GTUR	Gas turbine	N	N
ICEN	Internal combustion engine	N	N
SCSP	Concentrated Solar Power	Y	Y
STUR	Steam turbine	N	N
Storage			

Continued on next page

Table 5.2 – continued from previous page

Technology	Description	VRES	Storage
BATS	Stationary batteries	N	Y
BEVS	Battery-powered electric vehicles	N	Y
CAES	Compressed air energy storage	N	Y
P2GS	Power-to-gas storage	N	Y
THMS	Thermal storage	N	Y
Heat only			
GETH	Geothermal district heating	Y	N
HOBO	Heat only boiler	N	N
SOTH	Solar thermal district heating	Y	N
Power to heat			
ABHP	Absorption heat pump (solar/geothermal/gas)	Y/N	N
ASHP	Air source heat pump	Y/N	N
GSHP	Ground source heat pump	Y/N	N
HYHP	Hybrid heat pump (Ground/air & HP/GAS-OIL	Y/N	N
P2HT	Power-to-heat	Y/N	N
REHE	Resistive heater	Y/N	N
WSHP	Water source heat pump	Y/N	N

5.4.3 Fuels

Dispa-SET only considers a limited number of different fuel types. They are summarised in the following table, together with some examples.

Table 5.3: Dispa-SET fuels

Fuel	Examples
AIR	Air energy from the surrounding environment (used by heat pumps and other heat generation technologies)
AMO	Ammonia
BIO	Bagasse, Biodiesel, Gas From Biomass, Gasification, Biomass, Briquettes, Cattle Residues, Rice Hulls Or Padi Husk, Straw, Wood Gas (From Wood Gasification), Wood Waste Liquids Excl Blk Liq (Incl Red Liquor, Sludge, Wood,Spent Sulfite Liquor And Oth Liquids, Wood And Wood Waste
GAS	Blast Furnace Gas, Boiler Natural Gas, Butane, Coal Bed Methane, Coke Oven Gas, Flare Gas, Gas (Generic), Methane, Mine Gas, Natural Gas, Propane, Refinery Gas, Sour Gas, Synthetic Natural Gas, Top Gas, Voc Gas & Vapor, Waste Gas, Wellhead Gas
GEO	Geothermal steam
HRD	Anthracite, Other Anthracite, Bituminous Coal, Coker By-Product, Coal Gas (From Coal Gasification), Coke, Coal (Generic), Coal-Oil Mixture, Other Coal, Coal And Pet Coke Mi, Coal Tar Oil, Anthracite Coal Waste, Coal-Water Mixture, Gob, Hard Coal / Anthracite, Imported Coal, Other Solids, Soft Coal, Anthracite Silt, Steam Coal, Subbituminous, Pelletized Synthetic Fuel From Coal, Bituminous Coal Waste)
HYD	Hydrogen
LIG	Lignite black, Lignite brown, lignite
NUC	U (Uranium), Pu (Plutonium)
OIL	Crude Oil, Distillate Oil, Diesel Fuel, No. 1 Fuel Oil, No. 2 Fuel Oil, No. 3 Fuel Oil, No. 4 Fuel Oil, No. 5 Fuel Oil, No. 6 Fuel Oil, Furnace Fuel, Gas Oil, Gasoline, Heavy Oil Mixture, Jet Fuel, Kerosene, Light Fuel Oil, Liquefied Propane Gas, Methanol, Naphtha, ,Gas From Fuel Oil Gasification, Fuel Oil, Other Liquid, Orimulsion, Petroleum Coke, Petroleum Coke Synthetic Gas, Black Liquor, Residual Oils, Re-Refined Motor Oil, Oil Shale, Tar, Topped Crude Oil, Waste Oil
OTH	All other energy carriers
PEA	Peat Moss
SUN	Solar energy
WAT	Hydro energy
WIN	Wind energy
WST	Digester Gas (Sewage Sludge Gas), Gas From Refuse Gasification, Hazardous Waste, Industrial Waste, Landfill Gas, Poultry Litter, Manure, Medical Waste, Refused Derived Fuel, Refuse, Waste Paper And Waste Plastic, Refinery Waste, Tires, Agricultural Waste, Waste Coal, Waste Water Sludge, Waste
WHT	Waste heat, Excess heat

Different fuels may be used to power a given technology, e.g. steam turbines may be fired with almost any fuel type. In Dispa-SET, each unit must be defined with the pair of values (technology,fuel). The next tables is derived from a commercial power plant database and indicates the number of occurences of each combination. It appears clearly that, even through some combinations are irrelevant, both characteristics are needed to define a power plant type.

f/t	COMC	GTUR	HDAM	HPHS	HROR	ICEN	PHOT	STUR	WTOF	WTON	Total
AMO	1	1									
BIO		2				10		79			91
GAS	485	188				28		97			798
GEO								10			10
HRD	4							389			393
HYD		1						1			2
LIG								249			249
NUC								138			138
OIL	7	94				27		146			274
PEA								17			17
SUN							20	7			27
UNK		2				1		1			4
WAT			33	23	21			1			78
WIN									9	27	36
WST		3				7		46			56
Total	496	290	33	23	21	73	20	1181	9	27	2173

5.4.4 Unit-specific or technology-specific inputs

Some parameters, such as the availability factor, the outage factor or the inflows may be defined at the unit level or at the technology level. For that reason, the pre-processing tool first lookups the unit name in the database to assign it a value, and then lookups the technology or the fuel if no unit-specific information has been found.

5.4.5 Demand

Electricity demand is given per zone and the first row of each column with the time series should be the zone name.

Heat demand timeseries is needed where CHP or P2HT plants are used. In the current formulation, each CHP/P2HT unit is covering a heat load. In other words, one power plant is connected to a single district heating network. Therefore, in the heat demand input file, the first column has to be a time index and the following columns the heat demand in MW. The first row should contain the exact name of the power plant that will cover this demand.

It si possible to assume that a share of the demand is flexible (see model formulation for more information). In that case, this flexible share is provided as times series for each zone (see for example the tests/dummy_data/ShareFlexible.csv file), referencend in the “FlexibleDemand” field of the config file. It is also necessary to specify the number of hours of equivalent demand shifting capacity. This is achieved through the “DemandFlexibility” field of the config file and is expressed in hours (i.e. the number of hours during which the maximum flexible demand can be stored for shifting). An example of such configuration is proivded in the ConfigTest

5.4.6 Countries

Although the nodes names can be freely user-defined in the database, for the Dispa-SET EU model, the ISO 3166-1 standard has been adopted to describe each country at the NUTS1 level (except for Greece and the United Kingdom, for which the abbreviations EL and UK are used according to EU Interinstitutional style guide). The list of countries is defined as:

Code	Country
AT	Austria
BE	Belgium
BG	Bulgaria
CH	Switzerland
CY	Cyprus
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
EL	Greece
ES	Spain
FI	Finland
FR	France
HR	Croatia
HU	Hungary
IE	Ireland
IT	Italy
LT	Lituania
LU	Luxembourg
LV	Latvia
MT	Malta
NL	Netherlands
NO	Norway
PL	Poland
PT	Portugal
RO	Romania
SE	Sweden
SI	Slovenia
SK	Slovakia
UK	United Kingdom

5.4.7 Power plant data

The power plant database may contain as many fields as desired, e.g. to ensure that the input data can be traced back, or to provide the id of this plant in another database. However, some fields are required by Dispa-SET and must therefore be defined in the database.

Common fields

The following fields must be defined for all units:

Table 5.5: Common fields for all units

Description	Field name	Units
Unit name	Unit	n.a.
Installed Power or Heat Capacity (for one unit)	PowerCapacity	MW
Number of thermal blocks belonging to one unit	Nunits	n.a.
Technology	Technology	n.a.
Primary fuel	Fuel	n.a.
Zone (Power)	Zone	n.a.
Zone (Heat)	Zone_th	n.a.
Efficiency	Efficiency	%
Efficiency at minimum load	MinEfficiency	%
CO2 intensity	CO2Intensity	TCO2/MWh
Minimum load	PartLoadMin	%
Ramp up rate	RampUpRate	%/min
Ramp down rate	RampDownRate	%/min)
Start-up time	StartUPTime	h
Minimum up time	MinUpTime	h
Minimum down time	MinDownTime	h
No load cost	NoLoadCost	EUR/h
Start-up cost	StartUpCost	EUR
Ramping cost	RampingCost	EUR/MW

NB: the fields indicated with % as unit must be entered in a non-dimensional way (i.e. 90% should be written 0.9).

Storage units

Some parameters must only be defined for the units equipped with storage. They can be left blank for all other units.

Table 5.6: Specific fields for storage units

Description	Field name	Units
Storage capacity	STOCapacity	MWh
Self-discharge rate	STOSelfDischarge	%/d
Maximum charging power	STOMaxChargingPower	MW
Charging efficiency	STOChargingEfficiency	%

In the case of a storage unit, the discharge efficiency should be assigned to the common field "Efficiency". Similarly, the common field "PowerCapacity" is the nominal power in discharge mode.

CHP units

Some parameters must only be defined for the units equipped with CHP. They can be left blank for all other units.

Table 5.7: Specific fields for CHP units

Description	Field name	Units
CHP Type	CHPType	extraction/back-pressure/p2h
Power-to-heat ratio	CHPPowerToHeat	•
Power Loss factor	CHPPowerLossFactor	•
Maximum heat production	CHPMaxHeat	MW(th)
Capacity of heat Storage	STOCapacity	MWh(th)
% of storage heat losses per day	STOSelfDischarge	%/d

In the current version of DispaSet three type of combined heat and power units are supported:

- Extraction/condensing units
- Backpressure units
- Power to heat

For each of the above configurations the following fields must be filled:

Table 5.8: Mandatory fields per type of CHP unit (X: mandatory, o:optional)

Description	Extraction	Backpressure	Power to heat
CHPType	X	X	X
CHPPowerToHeat	X	X	
CHPPowerLossFactor	X		X
CHPMaxHeat	o	o	X
STOCapacity	o	o	o
STOSelfDischarge	o	o	o

There are numerous data checking routines to ensure that all data provided is consistent.

Warning: For extraction/condensing CHP plants, the power plant capacity (*PowerCapacity*) must correspont to the nameplate capacity in the maximum heat and power mode. Internal Dispaset calculations will use the equivalent stand-alone plants capacity based on the parameters provided.

P2HT units

Some parameters must only be defined for the power-to-heat units (heat pumps, electrical heaters). They can be left blank for all other units.

Table 5.9: Specific fields for P2HT units

Description	Field name	Units
Nominal coefficient of performance	COP	•
Nominal temperature	Tnominal	°C
First coefficient	coef_COP_a	•
Second coefficient	coef_COP_b	•
Capacity of heat Storage	STOCapacity	MWh(th)
% of storage heat losses per day	STOSelfDischarge	%/d

NB:

- Electrical heaters can be simulated by setting the nominal COP to 1 and the temperature coefficients to 0
- The two coefficients a and b aim at correcting the COP for the ambient temperatures. They are calculated as follows:

$$COP = COP_{nom} + coef_a \cdot (T - T_{nom}) + coef_b \cdot (T - T_{nom})^2$$

where T is the atmospheric temperature which needs to be provided as a times sereis for each zone in a csv file. The first row of the csv file is the zone name and a proper time index is required. The csv file path must be provided in the "Temperatures" field of the configuration file (see ConfigTest.xlsx for an example)

> **Warning:** For power-to-heat units, the power plant capacity (*PowerCapacity*) must correspont to the nameplate nominal ELECTRICAL consumption, thus given by the thermal capacity divided by the nominal COP.

5.4.8 Renewable generation

Variable renewable generation is defined as power generation from renewable source that cannot be stored: its is either fed to the grid or curtailed. The technologies falling under this definition are the ones described in the subset "tr" in the model definition.

The time-dependent genration of for these technologies must be provided as an exogenous time series in the form of an "availability factor". The latter is defined as the proportion of the nominal power capacity that can be generated at each hour.

In the database, the time series are provided as column vectors with the technology name as header. After the pre-processing, an availability factor is attributed to each unit according to their technology. Non-renewable technologies are assigned an availability factor of 1.

5.4.9 Storage and hydro data

Storage units are an extension of the regular units, including additional constraints and parameters. In the power plant table, four additional parameters are required: storage capacity (in MWh), self-discharge (in %/d), discharge power (in MW) and discharge efficiency (in %).

Some other parameters must be introduced in the form of time series in the "HydroData" section of the Dispa-SET database. There are described hereunder.

It should be noted that the nomenclature adopted for the modeling of storage units refers to the characteristics of hydro units with water reservoirs. However, these parameters (e.g. inflows, level) can easily be transposed to the case of alternative storage units such as batteries or CAES.

Inflows

The Inflows are defined as the contribution of exogenous sources to the level (or state of charge) or the reservoir. They are expressed in MWh of potential energy. If the inflows are provided as m^3/h, they must be converted.

The input to dispaset is defined as "StorageInflows". It is the normalized values of the inflow with respect to the nominal power of the storage unit (in discharge mode). As an example, if the inflow value at a certain time is 100MWh/h and if the turbining capacity of the hydro plant is 200 MW, the scaled inflow value must be defined as 0.5.

Scaled inflows should be provided in the form of time series with the unit name or the technology as columns header.

Storage level

Because emptying the storage has a zero marginal cost, a non-constrained optimization tends to leave the storage completely empty at the end of the optimisation horizon. For that reason, a minimum storage level is imposed at the last hour of each horizon. In Dispa-SET, a typical optimisation horizon is a few days. The model is therefore not capable of optimising the storage level e.g. for seasonal variations. The minimum storage level at the last hour is therefore an exogenous input. It can be selected from a historical level or obtained from a long-term hydro scheduling optimization.

The level input in the Dispa-SET database is normalized with respect to the storage capacity: its minimum value is zero and its maximum is one.

Variable capacity storage

In special cases, it might be necessary to simulate a storage unit whose capacity varies in time. A typical example is the simulation of the storage capacity provided by electric vehicles: depending on the time of the day, the connected battery capacity varies.

This special case can be simulated using the "AvailabilityFactor" input. In the case of a storage unit, reduces the available capacity by a factor varying from 0 to 1.

5.4.10 Other storage units

Other storage units include H2 storage, batteries (BATS) and electric vehicles (BEVS). In case of H2 storage, the parameter StorageInflow are defined null at all times whereas StorageOutflow corresponds to the hydrogen demand at each timsestep. For batteries and BEVS, both parameters are set to 0 all the time.

5.4.11 Power plant outages

In the current version, Dispa-SET does not distinguish planned outages from unplanned outages. They are characterized for each unit by the "OutageFactor" parameter. This parameter varies from 0 (no outage) to 1 (full outage). The available unit power is thus given by its nominal capacity multiplied by (1-OutageFactor).

The outages are provided in the dedicated section of the Database for each unit. They consist of a time series with the unit name as columns header.

5.4.12 Interconnections

Two cases should be distinguished when considering interconnections:

- Interconnections occuring between the simulated zones
- Interconnections occuring between the simulated zones and the Rest of the World (RoW)

These two cases are addresses by two different datasets described here under.

Net transfer capacities

Dispa-SET indogenously models the internal exchanges between countries (or zones) using a commercial net transfer caapcity (NTC). It does not consider (yet) DC power flows or more complex grid simulations.

Since the NTC values might vary in time, they must be supplied as time series, whose header include the origin country, the string ' -> ' and the destination country. As an example, the NTC from belgium to france must be provided with the header 'BE -> FR'.

Because NTCs are not necessarily symetrical, they must be provided in both directions (i.e. 'BE -> FR' and 'FR -> BE'. Non-provided NTCs are considered to be zero (i.e. no interconnection).

Historical physical flows

In Dispa-SET, the flows between internal zones and the rest of the world cannot be modeled endogenously. They must be provided as exogenous inputs. These inputs are referred to as "Historical physical flows", although they can also be user-defined.

In the input table of historical flows, the headers are similar to those of the NTCs (ie. 'XX -> YY'). All flows occuring an internal zone of the simulation and outside zones are considered as external flows and summed up. As an example, the historical flows 'FR -> XX', 'FR -> YY' and 'FR -> ZZ' will be aggregated in to a single interconnection flow 'FR -> RoW' if XX, YY and ZZ are not simulated zones.

These aggregated historical flows are then imposed to the solver as exogenous inputs.

In Dispa-SET, the flows are defined as positive variables. For each zone, there will thus be a maximum of two vectors defining its exchanges with the rest of the world (e.g. 'FR -> RoW' and 'RoW -> FR').

As for the NTCs, undefined historical flows are considered to be zero, i.e. not provided any historical flows is equivalent to consider the system as islanded.

5.4.13 Fuel Prices

Fuel prices vary both geographically and in time. They must therefore be provided as a time series for each simulated zone. One table is provided per fuel type, with as column header the zone to which it applies. If no header is provided, the fuel price is applied to all the simulated zones.

5.5 Model Description

The model is expressed as a MILP or LP problem. Continuous variables include the individual unit dispatched power, the shedded load and the curtailed power generation. The binary variables are the commitment status of each unit. The main model features can be summarized as follows:

5.5.1 Variables

Sets

Name	Description
au	All units
f	Fuel types
h	Hours
i(h)	Time step in the current optimization horizon
l	Transmission lines between nodes
mk	{DA: Day-Ahead, 2U: Reserve up, 2D: Reserve Down, flex}
n	Zones within each country (currently one zone, or node, per country)
nth	District heating zones (Multiple units can supply heat)
p	Pollutants
p2h(au)	Power to heat units
p2h2(s)	Power to H2 storage units
t	Power generation technologies
th(au)	Units with thermal storage
hu(au)	Heat only units
tr(t)	Renewable power generation technologies
u(au)	Generation units (all units minus P2HT units)
s(u)	Storage units (including hydro reservoirs)
chp(u)	CHP units
wat(s)	Hydro storage technologies
z(h)	Subset of every simulated hour

Parameters

Name	Units	Description
AvailabilityFactor(au,h)	%	Percentage of nominal capacity available
CHPPowerLossFactor(u)	%	Power loss when generating heat
CHPPowerToHeat(u)	%	Nominal power-to-heat factor
CHPMaxHeat(chp)	MW	Maximum heat capacity of chp plant
CHPType	n.a.	CHP Type
CommittedInitial(u)	n.a.	Initial commitment status
CostFixed(u)	EUR/h	Fixed costs
CostLoadShedding(n,h)	EUR/MWh	Shedding costs
CostRampDown(u)	EUR/MW	Ramp-down costs
CostRampUp(u)	EUR/MW	Ramp-up costs
CostShutDown(u)	EUR/u	Shut-down costs for one unit
CostStartUp(u)	EUR/u	Start-up costs for one unit
CostVariable(au,h)	EUR/MWh	Variable costs
CostHeatSlack(nth,h)	EUR/MWh	Cost of supplying heat via other means
CostH2Slack(p2h2,h)	EUR/MWh	Cost of supplying H2 by other means
Curtailment(n)	n.a.	Curtailment {binary: 1 allowed}
Demand(mk,n,h)	MW	Hourly demand in each zone
Efficiency(p2h,h)	%	Power plant efficiency
EmissionMaximum(n,p)	tP	Emission limit per zone for pollutant p
EmissionRate(u,p)	tP/MWh	Emission rate of pollutant p from unit u

Continued on n

Table 5.10 – continued from previous page

Name	Units	Description
FlowMaximum(l,h)	MW	Maximum flow in line
FlowMinimum(l,h)	MW	Minimum flow in line
Fuel(u,f)	n.a.	Fuel type used by unit u {binary: 1 u uses f}
HeatDemand(nth,h)	MWh/u	Heat demand profile for chp units
K_QuickStart(n)	n.a.	Part of the reserve that can be provided by offline quickstart units
LineNode(l,n)	n.a.	Line-zone incidence matrix {-1,+1}
LoadMaximum(au,h)	%	Maximum load given AF and OF
LoadShedding(n,h)	MW	Load that may be shed per zone in 1 hour
Location(au,n)	n.a.	Location {binary: 1 u located in n}
LocationTH(au,nth)	n.a.	Location {binary: 1 u located in nth)
LPFormulation	n.a.	Defines the equation that will be present: 1 for LP and 0 for MIP
Markup	EUR/MW	Markup
MTS	n.a.	Defines the equation that will be present: 1 for MidTermScheduling, 0 for normal op
Nunits(u)	n.a.	Number of units inside the cluster
OutageFactor(au,h)	%	Outage factor (100 % = full outage) per hour
PartLoadMin(u)	%	Percentage of minimum nominal capacity
PowerCapacity(au)	MW/u	Installed capacity
PowerInitial(u)	MW/u	Power output before initial period
PowerMinStable(au)	MW/u	Minimum power for stable generation
PowerMustRun(u)	MW	Minimum power output
PriceTransmission(l,h)	EUR/MWh	Price of transmission between zones
QuickStartPower(u,h)	MW/h/u	Available max capacity for tertiary reserve
RampDownMaximum(u)	MW/h/u	Ramp down limit
RampShutDownMaximum(u,h)	MW/h/u	Shut-down ramp limit
RampStartUpMaximum(u,h)	MW/h/u	Start-up ramp limit
RampUpMaximum(u)	MW/h/u	Ramp up limit
Reserve(t)	n.a.	Reserve provider {binary}
StorageCapacity(au)	MWh/u	Storage capacity (reservoirs)
StorageChargingCapacity(au)	MW/u	Maximum charging capacity
StorageChargingEfficiency(au)	%	Charging efficiency
StorageDischargeEfficiency(au)	%	Discharge efficiency
StorageInflow(u,h)	MWh/u	Storage inflows
StorageInitial(au)	MWh	Storage level before initial period
StorageMinimum(au)	MWh/u	Minimum storage level
StorageOutflow(u,h)	MWh/u	Storage outflows (spills)
StorageProfile(u,h)	%	Storage long-term level profile
StorageSelfDischarge(au)	%/day	Self discharge of the storage units
Technology(au,t)	n.a.	Technology type {binary: 1: u belongs to t}
TimeDownMinimum(u)	h	Minimum down time
TimeStep	h	Duration of a timestep of optimization
TimeUpMinimum(u)	h	Minimum up time
VOLL()	EUR/MWh	Value of lost load

NB: When the parameter is expressed per unit ("/u"), its value must be provided for one single unit (even in the case of a clustered formulation).

Optimization Variables

Name	Units	Description
AccumulatedOverSupply(n,h)	MWh	Accumulated oversupply due to the flexible demand
Committed(u,h)	n.a.	Unit committed at hour h {1,0}
CostStartUpH(u,h)	EUR	Cost of starting up
CostShutDownH(u,h)	EUR	Cost of shutting down
CostRampUpH(u,h)	EUR	Ramping cost
CostRampDownH(u,h)	EUR	Ramping cost
CurtailedPower(n,h)	MW	Curtailed power at node n
CurtailedHeat(n_th,h)	MW	Curtailed heat at node nth
Flow(l,h)	MW	Flow through lines
H2Output(au,h)	MWh	H2 output from H2 storage to fulfill the demand
Heat(au,h)	MW	Heat output by chp plant
HeatSlack(nth,h)	MW	Heat satisfied by other sources
Power(u,h)	MW	Power output
PowerConsumption(p2h,h)	MW	Power consumption by P2H
PowerMaximum(u,h)	MW	Power output
PowerMinimum(u,h)	MW	Power output
PtLDemand(au,h)	MW	Demand of H2 for PtL at each time step for P2HT units
Reserve_2U(u,h)	MW	Spinning reserve up
Reserve_2D(u,h)	MW	Spinning reserve down
Reserve_3U(u,h)	MW	Non spinning quick start reserve up
ShedLoad(n,h)	MW	Shed load
StorageInput(au,h)	MWh	Charging input for storage units
StorageLevel(au,h)	MWh	Storage level of charge
StorageSlack(au,h)	MWh	Unsatisfied storage level
Spillage(s,h)	MWh	Spillage from water reservoirs
SystemCost(h)	EUR	Total system cost
LL_MaxPower(n,h)	MW	Deficit in terms of maximum power
LL_RampUp(u,h)	MW	Deficit in terms of ramping up for each plant
LL_RampDown(u,h)	MW	Deficit in terms of ramping down
LL_MinPower(n,h)	MW	Power exceeding the demand
LL_2U(n,h)	MW	Deficit in reserve up
LL_3U(n,h)	MW	Deficit in reserve up - non spinning
LL_2D(n,h)	MW	Deficit in reserve down
WaterSlack(s)	MWh	Unsatisfied water level at end of optimization period

Free Variables

Name	Units	Description
SystemCostD	EUR	Total system cost for one optimization period
DemandModulation	MW	Difference between the flexible demand and the baseline

Integer Variables

Name	Units	Description
Committed(u,h)	n.a.	Number of unit committed at hour h {1 0} or integer
StartUp(u,h)	n.a.	Number of unit startups at hour h {1 0} or integer
ShutDown(u,h)	n.a.	Number of unit shutdowns at hour h {1 0} or integer

5.5.2 Optimisation model

The aim of this model is to represent with a high level of detail the short-term operation of large-scale power systems solving the so-called unit commitment problem. To that aim we consider that the system is managed by a central operator with full information on the technical and economic data of the generation units, the demands in each node, and the transmission network.

The unit commitment problem considered in this report is a simplified instance of the problem faced by the operator in charge of clearing the competitive bids of the participants into a wholesale day-ahead power market. In the present formulation the demand side is an aggregated input for each node, while the transmission network is modelled as a transport problem between the nodes (that is, the problem is network-constrained but the model does not include the calculation of the optimal power flows).

The unit commitment problem consists of two parts: i) scheduling the start-up, operation, and shut down of the available generation units, and ii) allocating (for each period of the simulation horizon of the model) the total power demand among the available generation units in such a way that the overall power system costs is minimized. The first part of the problem, the unit scheduling during several periods of time, requires the use of binary variables in order to represent the start-up and shut down decisions, as well as the consideration of constraints linking the commitment status of the units in different periods. The second part of the problem is the so-called economic dispatch problem, which determines the continuous output of each and every generation unit in the system. Therefore, given all the features of the problem mentioned above, it can be naturally formulated as a mixed-integer linear program (MILP). However, the problem can also be relaxed to a linear program (LP).

There is a possibility of Mid Term scheduling. It allows to optimize the level of energy in the storage reservoirs over a year and use it as endogeneous input in the optimization of interest. In that case, the equations linked to unit commitment are ignored.

Since our goal is to model a large European interconnected power system, we have implemented a so-called tight and compact formulation, in order to simultaneously reduce the region where the solver searches for the solution and increase the speed at which the solver carries out that search. Tightness refers to the distance between the relaxed and integer solutions of the MILP and therefore defines the search space to be explored by the solver, while compactness is related to the amount of data to be processed by the solver and thus determines the speed at which the solver searches for the optimum. Usually tightness is increased by adding new constraints, but that also increases the size of the problem (decreases compactness), so both goals contradict each other and a trade-off must be found.

Objective function

The goal of the unit commitment problem is to minimize the total power system costs (expressed in EUR in equation), which are defined as the sum of different cost items, namely: start-up and shut-down, fixed, variable, ramping, transmission-related and load shedding (voluntary and involuntary) costs.

$$
\begin{aligned}
min\Big[& \sum_{u,i} CostFixed_u \cdot Committed_{u,i} \cdot TimeStep \\
& + \sum_{u,i} (CostStartUpH_{u,i} + CostShutDownH_{u,i}) \\
& + \sum_{u,i} (CostRampUpH_{u,i} + CostRampDownH_{u,i}) \\
& + \sum_{u,i} CostVariable_{u,i} \cdot Power_{u,i} \cdot TimeStep \\
& + \sum_{hu,i} CostVariable_{hu,i} \cdot Heat_{hu,i} \cdot TimeStep \\
& + \sum_{l,i} PriceTransimission_{l,i} \cdot Flow_{l,i} \cdot TimeStep \\
& + \sum_{n,i} CostLoadShedding_{i,n} \cdot ShedLoad_{i,n} \cdot TimeStep \\
& + \sum_{th,i} CostHeatSlack_{nth,i} \cdot HeatSlack_{nth,i} \cdot TimeStep) \\
& + \sum_{p2h2,i} CostH2Slack_{p2h2,i} \cdot StorageSlack_{p2h2,i} \cdot TimeStep \\
& + \sum_{chp,i} CostVariable_{chp,i} \cdot CHPPowerLossFactor_{chp} \cdot Heat_{chp,i} \cdot TimeStep) \\
& + \sum_{i,n} VOLL_{Power} \cdot (LL_{MaxPower,i,n} + LL_{MinPower,i,n}) \cdot TimeStep \\
& + \sum_{i,n} 0.8 \cdot VOLL_{Reserve} \cdot (LL_{2U,i,n} + LL_{2D,i,n} + LL_{3U,i,n}) \cdot TimeStep \\
& + \sum_{u,i} 0.7 \cdot VOLL_{Ramp} \cdot (LL_{RampUp,u,i} + LL_{RampDown,u,i}) \cdot TimeStep \\
& + \sum_{s,i} CostOfSpillage \cdot spillage_{s,i} \\
& + \sum_{s,i} WaterValue \cdot WaterSlack_s \Big]
\end{aligned}
$$

The costs can be broken down as:

- Fixed costs: depending on whether the unit is on or off.
- Variable costs: stemming from the power output of the units.
- Start-up costs: due to the start-up of a unit.
- Shut-down costs: due to the shut-down of a unit.
- Ramp-up: emerging from the ramping up of a unit.
- Ramp-down: emerging from the ramping down of a unit.
- Load shed: due to necessary load shedding.
- Transmission: depending of the flow transmitted through the lines.
- Loss of load: power exceeding the demand or not matching it, ramping and reserve.
- spillage: due to spillage in storage.

- H2: cost of unsatisfied hydrogen by production from electrolyzers
- Water : cost of water coming from unsatisfied water level at the end of the optimization period.

The variable production costs (in EUR/MWh), are determined by fuel and emission prices corrected by the efficiency (which is considered to be constant for all levels of output in this version of the model) and the emission rate of the unit (equation):

$$CostVariable_{au,h} = Markup_{au,h} + \sum_{n,f} \left(\frac{Fuel_{au,f} \cdot FuelPrice_{n,f,h} \cdot Location_{au,n}}{Efficiency_u} \right) \tag{5.1}$$

$$+ \sum_{p} (EmissionRate_{au,p} \cdot PermitPrice_p) \tag{5.2}$$

The variable cost includes an additional mark-up parameter that can be used for calibration and validation purposes.

From version 2.3, Dispa-SET uses a 3 integers formulations of the up/down status of all units. According to this formulation, the number of start-ups and shut-downs is at each time step is computed by:

$$Committed_{u,i} - Committed_{u,i-1} = StartUp_{u,i} - ShutDown_{u,i}$$

The start-up and shut-down costs are positive variables, calculated from the number of startups/shutdowns at each time step:

$$CostStartUp_{u,i} = CostStartUp_u \cdot StartUp_{u,i} \tag{5.3}$$

$$CostShutDown_{u,i} = CostShutDown_u \cdot ShutDown_{u,i} \tag{5.4}$$

Renewable units are enforced commited when the availability factor is non null and the outage factor is not 1 and decommited in the other case.

Ramping costs are defined as positive variables (i.e. negative costs are not allowed) and are computed with the following equations:

$$CostRampUp_{u,i} \geq CostRampUp_u \cdot (Power_{u,i} - Power_{u,i-1}) \tag{5.5}$$

$$CostRampDown_{u,i} \geq CostRampDown_u \cdot (Power_{u,i-1} - Power_{u,i}) \tag{5.6}$$

It should be noted that in case of start-up and shut-down, the ramping costs are added to the objective function. Using start-up, shut-down and ramping costs at the same time should therefore be performed with care.

In the current formulation, all other costs (fixed and variable costs, transmission costs, load shedding costs) are considered as exogenous parameters.

As regards load shedding, the model considers the possibility of voluntary load shedding resulting from contractual arrangements between generators and consumers. Additionally, in order to facilitate tracking and debugging of errors, the model also considers some variables representing the capacity the system is not able to provide when the minimum/maximum power, reserve, or ramping constraints are reached. These lost loads are a very expensive last resort of the system used when there is no other choice available. The different lost loads are assigned very high values (with respect to any other costs). This allows running the simulation without infeasibilities, thus helping to detect the origin of the loss of load. In a normal run of the model, without errors, all these variables are expected to be equal to zero.

Day-ahead energy balance

The main constraint to be met is the supply-demand balance, for each period and each zone, in the day-ahead market (equation). According to this restriction, the sum of all the power produced by all the units present in the node (including the power generated by the storage units), the power injected from neighbouring nodes, and the curtailed power from intermittent sources is equal to the load in that node, plus the power consumed for energy storage, minus the load interrupted and the load shed.

$$\sum_u (Power_{u,i} \cdot Location_{u,n}) + \sum_l (Flow_{l,i} \cdot LineNode_{l,n}) \tag{5.7}$$

$$= Demand_{DA,n,h} + \sum_s \left(StorageInput_{s,h} \cdot Location_{s,n}\right) - ShedLoad_{n,i} \tag{5.8}$$

$$+ \sum_{p2h} PowerConsumption_{p2h,i} \cdot Location_{p2h,n} - LL_{MaxPower\,n,i} + LL_{MinPower\,n,i} \tag{5.9}$$

Reserve constraints

Besides the production/demand balance, the reserve requirements (upwards and downwards) in each node must be met as well. In Dispa-SET, three types of reserve requirements are taken into account:

- Upward secondary reserve (2U): reserve that can only be covered by spinning units
- Downward secondary reserve (2D): reserve that can only be covered by spinning units
- Upward tertiary reserve (3U): reserve that can be covered either by spinning units or by quick-start offline units

The secondary reserve capability of committed units is limited by the capacity margin between current and maximum power output:

$$Reserve_{2U\,u,i} \leq PowerCapacity_u \cdot AvailabilityFactor_{u,i} \cdot (1 - OutageFactor_{u,i}) \cdot Committed_{u,i} \tag{5.10}$$

$$- Power_{u,i} \tag{5.11}$$

The same applies to the downwards secondary reserve capability, with an additional term to take into account the downard reserve capability of storage units:

$$Reserve_{2D\,u,i} \leq Power_{u,i} - PowerMustRun_{u,i} \cdot Committed_{u,i} \tag{5.12}$$

$$+ (StorageChargingCapacity_u \cdot Nunits_u - StorageInput_{u,i}) \tag{5.13}$$

The quick start (non-spining) reserve capability is given by:

$$Reserve_{3U\,u,i} \leq (Nunits_u - Committed_{u,i}) \cdot QuickStartPower_{u,i} \cdot TimeStep$$

The secondary reserve demand should be fulfilled at all times by all the plants allowed to participate in the reserve market:

$$Demand_{2U,n,h} \leq \sum_{u,t} \left(Reserve_{2U\,u,i} \cdot Technology_{u,t} \cdot Reserve_t \cdot Location_{u,n}\right) \tag{5.14}$$

$$+ LL_{2U\,n,i} \tag{5.15}$$

The same equation applies to downward reserve requirements (2D).

The tertiary reserve can also be provided by non-spinning units. The inequality is thus transformed into:

$$Demand_{3U,n,h} \leq \sum_{u,t} [(Reserve_{2U\,u,i} + Reserve_{3U\,u,i}) \cdot Technology_{u,t} \cdot Reserve_t \cdot Location_{u,n}] \tag{5.16}$$

$$+ LL_{3U\,n,i} \tag{5.17}$$

Reserve Requirements

The reserve requirements are defined by the users. In case no input is provided, one among the three methods modeled in Dispa-SET and briefly described here can be selected.

The first method proposed to evaluate the needs for reserves is static and based on an empirical formula which is function of the maximum expected load for each day. The empirical formula is described by:

$$Demand_{2U,n,i} = \sqrt{10 \cdot \max_h \left(Demand_{DA,n,h}\right) + 150^2} - 150$$

Downward reserves are defined as 50% of the upward margin:

$$Demand_{2D,n,h} = 0.5 \cdot Demand_{2U,n,h}$$

The second formulation proposed by Dispa-SET is dynamic and based on the (3+5)% rule. Reserve requirements are computed as a fraction of the forecasted demand and available wind and solar power at a certain hour of the day. Here the formula:

$$Demand_{2U,n,h} = 0.03 \cdot Demand_{DA,n,h} \tag{5.18}$$

$$+0.05 \cdot AvailableWindPower_{u,i} + 0.05 \cdot AvailablePhotPower_{u,i} \tag{5.19}$$

In this case downward reserves are equal to upward reserves.

The third and last method proposed in Dispa-SET is dynamic and probabilistic. It accounts for reserve requirements as the sum of two components as follows:

$$Demand_{2U,n,h} = \sqrt{10 \cdot Demand_{DA,n,h} + 150^2} - 150 \tag{5.20}$$

$$+2.74 \cdot \sqrt{\sigma^2_{L,n,h} + \sigma^2_{W,n,h} + \sigma^2_{S,n,h}} \tag{5.21}$$

where the second part of the function exploits the standard deviations of demand, solar and wind power forecast error functions assuming a confidence level equal to 99.7%. Also in this last case downward reserves are equal to upward reserves.

Power output bounds

The minimum power output is determined by the must-run or stable generation level of the unit if it is committed:

$$PowerMustRun_{u,i} \cdot Committed_{u,i} \leq Power_{u,i}$$

In the particular case of CHP unit (extration type or power-to-heat type), the minimum power is defined for for a heat demand equal to zero. If the unit produces heat, the minimum power must be reduced according to the power loss factor and the previous equation is replaced by:

$$\begin{aligned} PowerMustRun_{chp,i} \cdot Committed_{chp,i} \\ -StorageInput_{chp,i} \cdot CHPPowerLossFactor_u \\ \leq Power_{chp,i} \end{aligned}$$

The power output is limited by the available capacity, if the unit is committed:

$$\begin{aligned} Power_{u,i} \\ \leq PowerCapacity_u \cdot AvailabilityFactor_{u,i} \cdot (1 - OutageFactor_{u,i}) \cdot Committed_{u,i} \end{aligned}$$

The availability factor is used for renewable technologies to set the maximum time-dependent generation level. It is set to one for the traditional power plants. The outage factor accounts for the share of unavailable power due to planned or unplanned outages.

Ramping Constraints

Each unit is characterized by a maximum ramp up and ramp down capability. This is translated into the following inequality for the case of ramping up:

$$\begin{aligned}
Power_{u,i} - Power_{u,i-1} \leq \\
(Committed_{u,i} - StartUp_{u,i}) \cdot RampUpMaximum_u \cdot TimeStep \\
+StartUp_{u,i} \cdot RampStartUpMaximum_u \cdot TimeStep \\
-ShutDown_{u,i} \cdot PowerMustRun_{u,i} \\
+LL_{RampUp_{u,i}}
\end{aligned}$$

and for the case of ramping down:

$$\begin{aligned}
Power_{u,i-1} - Power_{u,i} \leq \\
(Committed_{u,i} - ShutDown_{u,i}) \cdot RampDownMaximum_u \cdot TimeStep \\
+ShutDown_{u,i} \cdot RampShutDownMaximum_u \cdot TimeStep \\
-StartUp_{u,i} \cdot PowerMustRun_{u,i} \\
+LL_{RampDown_{u,i}}
\end{aligned}$$

Note that this formulation is valid for both the clustered formulation and the binary formulation. In the latter case (there is only one unit u), if the unit remains committed, the inequality simplifies into:

$$\begin{aligned}
Power_{u,i} - Power_{u,i-1} \leq \\
RampUpMaximum_u \cdot TimeStep + LL_{RampUp_{u,i}}
\end{aligned}$$

If the unit has just been committed, the inequality becomes:

$$\begin{aligned}
Power_{u,i} - Power_{u,i-1} \leq \\
RampStartUpMaximum_u \cdot TimeStep + LL_{RampUp_{u,i}}
\end{aligned}$$

And if the unit has just been stopped:

$$\begin{aligned}
Power_{u,i} - Power_{u,i-1} \leq \\
-PowerMustRun_{u,i} + LL_{RampUp_{u,i}}
\end{aligned}$$

Minimum up and down times

The operation of the generation units is also limited as well by the amount of time the unit has been running or stopped. In order to avoid excessive ageing of the generators, or because of their physical characteristics, once a unit is started up, it cannot be shut down immediately. Reciprocally, if the unit is shut down it may not be started immediately.

To model this in MILP, the number of startups/shutdowns in the last N hours must be limited, N being the minimum up or down time. For the minimum up time, the number of startups during this period cannot be higher than the number of currently committed units:

$$\sum_{ii=i-\frac{TimeUpMinimum_u}{TimeStep}}^{i} StartUp_{u,ii} \leq Committed_{u,i}$$

i.e. the currently committed units are not allowed to have performed multiple on/off cycles between the optimization time minus TimeUpMinimum and the optimization time. The implied number of periods is computed by the ratio of TimeUpMinimum and TimeStep. If TimeUpMinimum is not a multiple of TimeStep, their fraction is rounded upwards. In case of a binary formulation (Nunits=1), if the unit is ON at time i, only one startup is allowed in the last TimeUpMinimum periods. If the unit is OFF at time i, no startup is allowed.

A similar inequality can be written for the ninimum down time:

$$\sum_{ii=i-\frac{TimeDownMinimum_u}{TimeStep}}^{i} ShutDown_{u,ii} \leq Nunits_u - Committed_{u,i}$$

Storage-related constraints

Generation units with energy storage capabilities (large hydro reservoirs, pumped hydro storage units, hydrogen storage units or batteries) must meet additional restrictions related to the amount of energy stored. Storage units are considered to be subject to the same constraints as non-storage power plants. In addition to those constraints, storage-specific restrictions are added for the set of storage units (i.e. a subset of all units). These restrictions include the storage capacity, inflow, outflow, charging, charging capacity, charge/discharge efficiencies, etc. Discharging is considered as the standard operation mode and is therefore linked to the Power variable, common to all units.

The first constraint imposes that the energy stored by a given unit is bounded by a minimum value:

$$StorageMinimum_s \leq StorageLevel_{s,i} \cdot Nunits_s$$

In the case of a storage unit, the availability factor applies to the charging/discharging power, but also to the storage capacity. The storage level is thus limited by:

$$StorageLevel_{s,i} \leq StorageCapacity_s \cdot AvailabilityFactor_{s,i} \cdot Nunits_s$$

The energy added to the storage unit is limited by the charging capacity. Charging is allowed only if the unit is not producing (discharging) at the same time (i.e. if Committed, corresponding to the normal mode, is equal to 0).

$$\begin{aligned} StorageInput_{s,i} \leq \\ StorageChargingCapacity_s \cdot (Nunits_s - Committed_{s,i}) \end{aligned}$$

Discharge is limited by the level of charge of the storage unit:

$$\begin{aligned} \frac{Power_{i,s} \cdot TimeStep}{StorageDischargeEfficiency_s} + StorageOutflow_{s,i} \cdot Nunits_s \cdot TimeStep \\ + Spillage_{wat,i} - StorageInflow_{s,i} \cdot Nunits_s \cdot TimeStep - StorageSlack_{p2h2,i} \\ \leq StorageLevel_{s,i} \end{aligned}$$

It is worthwhile to note that StorageInflow and StorageOuflow must be multiplied by the number of units because they are defined for a single storage plant. On the contrary StorageLevel, Spillage and Power are defined for all units s. StorageInflow and Storage Outflow are predefined time series, whose meaning depends on the type of storage units: for hydro units, it is the natural water flows. For hydrogen units, StorageInflow is 0 at all times, but StorageOutflow represents the hydrogen demand (for fuel cell vehicles, industries,…). For batteries, both parameters are null at all times.

Charge is limited by the level of charge of the storage unit:

$$\begin{aligned} StorageInput_{s,i} \cdot StorageChargingEfficiency_s \cdot TimeStep \\ - StorageOutflow_{s,i} \cdot Nunits_s \cdot TimeStep - Spillage_{wat,i} \\ + StorageInflow_{s,i} \cdot Nunits_s \cdot TimeStep + StorageSlack_{p2h2,i} \\ \leq StorageCapacity_s \cdot AvailabilityFactor_{s,i} \\ - StorageLevel_{s,i} \end{aligned}$$

Besides, the energy stored in a given period is given by the energy stored in the previous period, net of charges and discharges:

$$\begin{aligned} & StorageLevel_{s,i-1} + StorageInflow_{s,i} \cdot Nunits_s \cdot TimeStep \\ & + StorageInput_{s,i} \cdot StorageChargingEfficiency_s \cdot TimeStep + StorageSlack_{p2h2,i} \\ & = StorageLevel_{s,i} + StorageOutflow_{s,i} \cdot Nunits_s \cdot TimeStep \\ & + Spillage_{wat,i} + \frac{Power_{s,i} \cdot TimeStep}{StorageDischargeEfficiency_s} \end{aligned}$$

Some storage units are equiped with large reservoirs, whose capacity at full load might be longer than the optimisation horizon. Therefore, a minimum level constraint is required for the last hour of the optimisation, which otherwise would systematically tend to empty the reservoir as much a possible. An exogenous minimum profile is thus provided and the following constraint is applied:

$$StorageLevel_{s,N} \geq StorageFinalMin_s + WaterSlack_{wat}$$

where N is the last period of the optimization horizon, StorageProfile is a non-dimensional minimum storage level provided as an exogenous input and WaterSlack is a variable defining the unsatified water level. The price associated to that water is very high.

Heat balance

In Dispa-SET heat demand is specified for individual heating zones (nth). It can be covered either by a CHP plant, P2HT unit or by alternative heat supply options (Heat Slack) or a combination of all three types of units.

$$\begin{aligned} & \sum_{chp} Heat_{chp,i} \cdot LocationTH_{chp,nth} \\ & + \sum_{p2h} (Heat_{p2h,i} \cdot LocationTH_{p2h,nth}) \\ & + \sum_{hu} (Heat_{hu,i} \cdot LocationTH_{hu,nth}) \\ & = HeatDemand_{nth,i} - HeatSlack_{nth,i} \end{aligned}$$

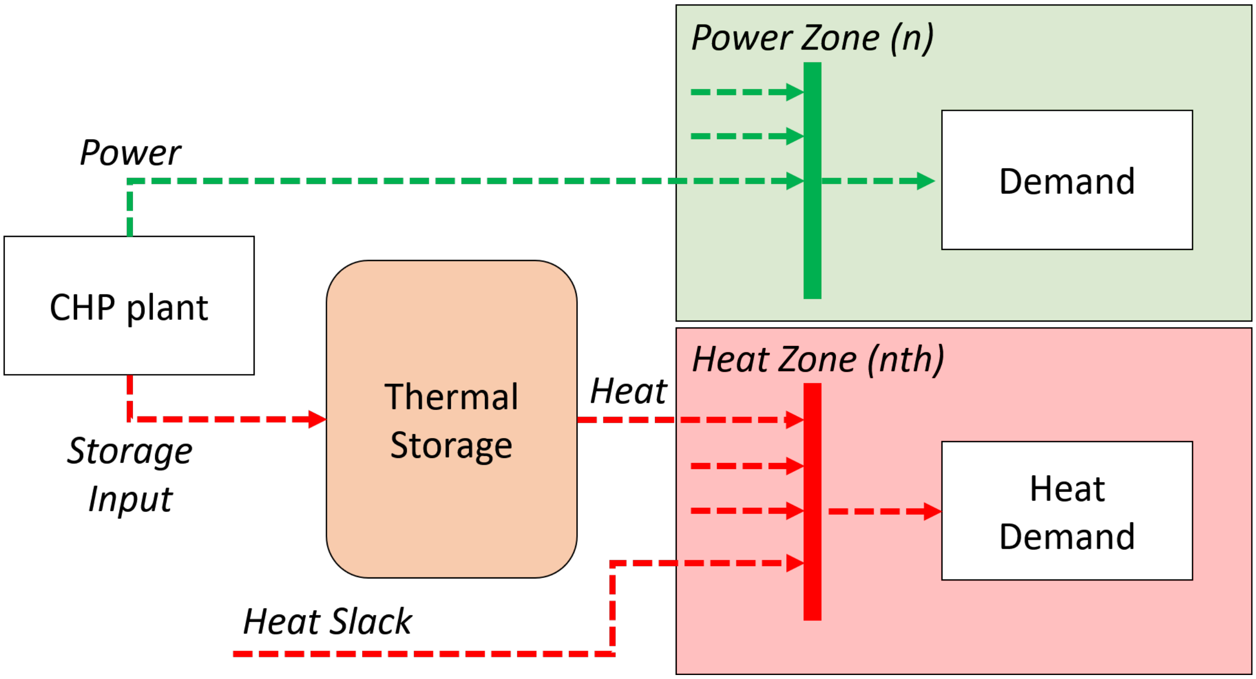
Heat output cosntraints

Simmilarly to Power output constraints, Heat output must be below maximum generation capacity.

$$Heat_{hu,i} \leq PowerCapacity_{hu} \cdot AvailabilityFactor_{hu,i} \cdot (1 - OutageFactor_{hu,i})$$

Heat production constraints (CHP plants only)

In DispaSET Power plants can be indicated as CHP which gives them the possibility to satisfy heat demand.



The following heat balance constraints are used for any CHP and P2H plant types.

$$StorageInput_{chp,i} \leq CHPMaxHeat_{chp} \cdot Nunits_{chp}$$

The constraints between heat and power production differ for each plant design and are explained within the following subsections.

Steam plants with Backpressure turbine

This options includes steam-turbine based power plants with a backpressure turbine. The feasible operating region is between AB. The slope of the line is the heat to power ratio.

$$Power_{chp,i} = StorageInput_{chp,i} \cdot CHPPowerToHeat_{chp}$$

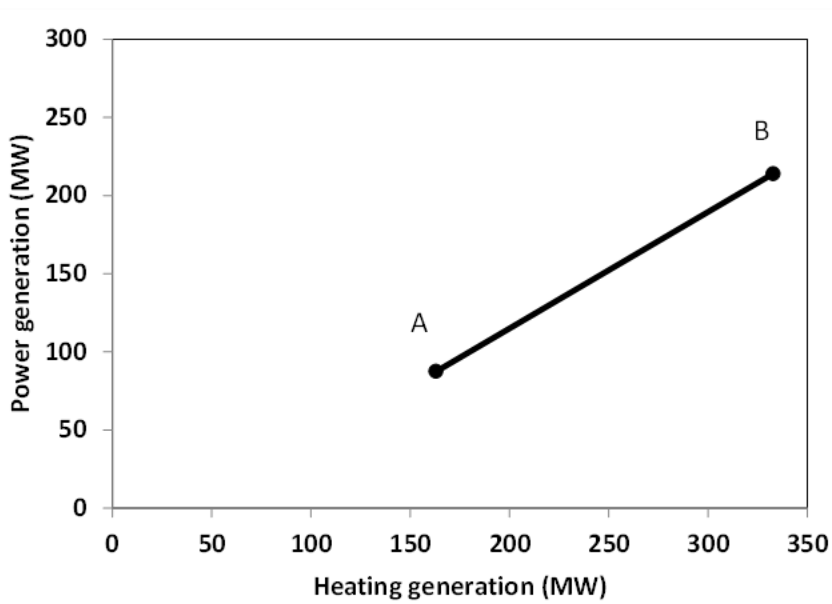
Steam plants with Extraction/condensing turbine

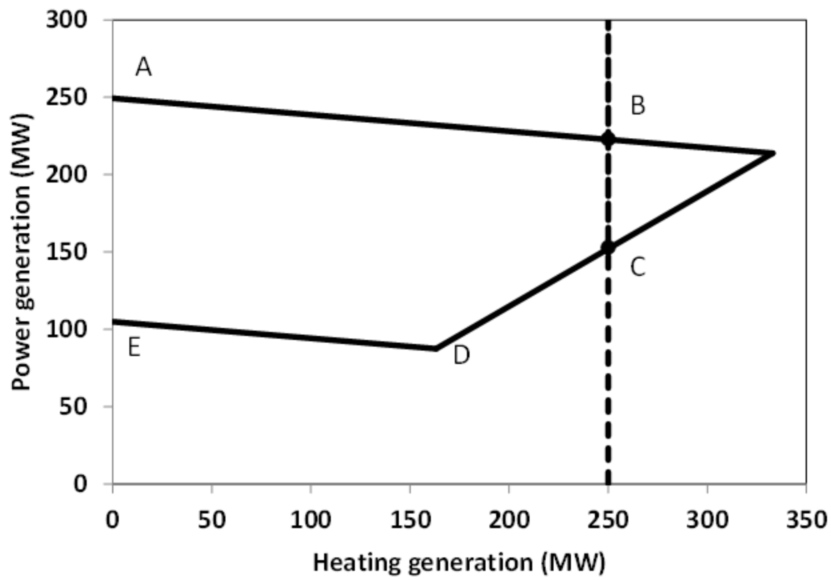
This options includes steam-turbine based power plants with an extraction/condensing turbine. The feasible operating region is within ABCDE. The vertical dotted line BC corresponds to the minimum condensation line (as defined by *CHPMaxHeat*). The slope of the DC line is the heat to power ratio and the slope of the AB line is the inverse of the power penalty ratio.

$$Power_{chp,i} \geq StorageInput_{chp,i} \cdot CHPPowerToHeat_{chp}$$

$$Power_{chp,i} \leq PowerCapacity_{chp} \cdot Nunits- StorageInput_{chp,i} \cdot CHPPowerLossFactor_{chp}$$

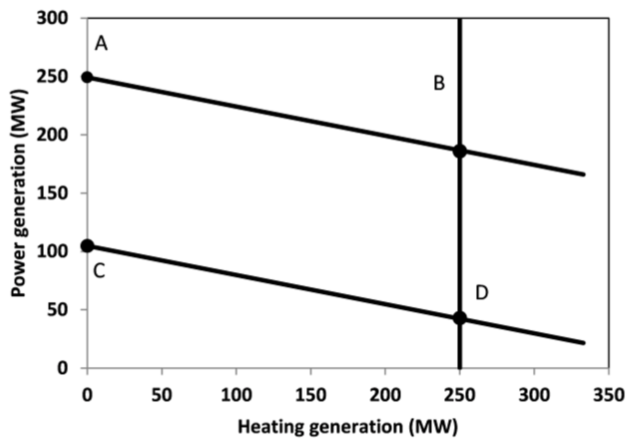
$$Power_{chp,i} \geq PowerMustRun_{chp,i} - StorageInput_{chp,i} \cdot CHPPowerLossFactor_{chp}$$


Power generation (MW)
Heating generation (MW)
A
B
0
50
100
150
200
250
300
350


Power generation (MW)
Heating generation (MW)
A
B
C
D
E
0
50
100
150
200
250
300
350

Power plant coupled with any power to heat option

This option includes power plants coupled with resistance heater or heat pumps. The feasible operating region is between ABCD. The slope of the AB and CD line is the inverse of the COP or efficiency. The vertical dotted line corresponds to the heat pump (or resistance heater) thermal capacity (as defined by *CHPMaxHeat*)



$$Power_{chp,i} \leq PowerCapacity_{chp} - StorageInput_{chp,i} \cdot CHPPowerLossFactor_{chp}$$

$$Power_{chp,i} \geq PowerMustRun_{chp,i} - StorageInput_{chp,i} \cdot CHPPowerLossFactor_{chp}$$

Power to heat units (labeled as P2HT technology)

Oposite to power plants coupled with any power to heat option, individual power to heat units (technology = P2HT) have only one mode of operation. They consume power to generate heat. In Dispa-SET these units are either small scale residential heat pumps or electric heaters or large industrial or district heating devices power by electricity. A shematic overview of these units is shown below:

They are subjet to the following set of constraints:

$$StorageInput_{p2h,i} = PowerConsumption_{p2h,i} \cdot Efficiency_{p2h,i}$$

$$PowerConsumption_{p2h,i} \leq PowerCapacity_{p2h} \cdot Nunits_{p2h}$$

Heat Storage

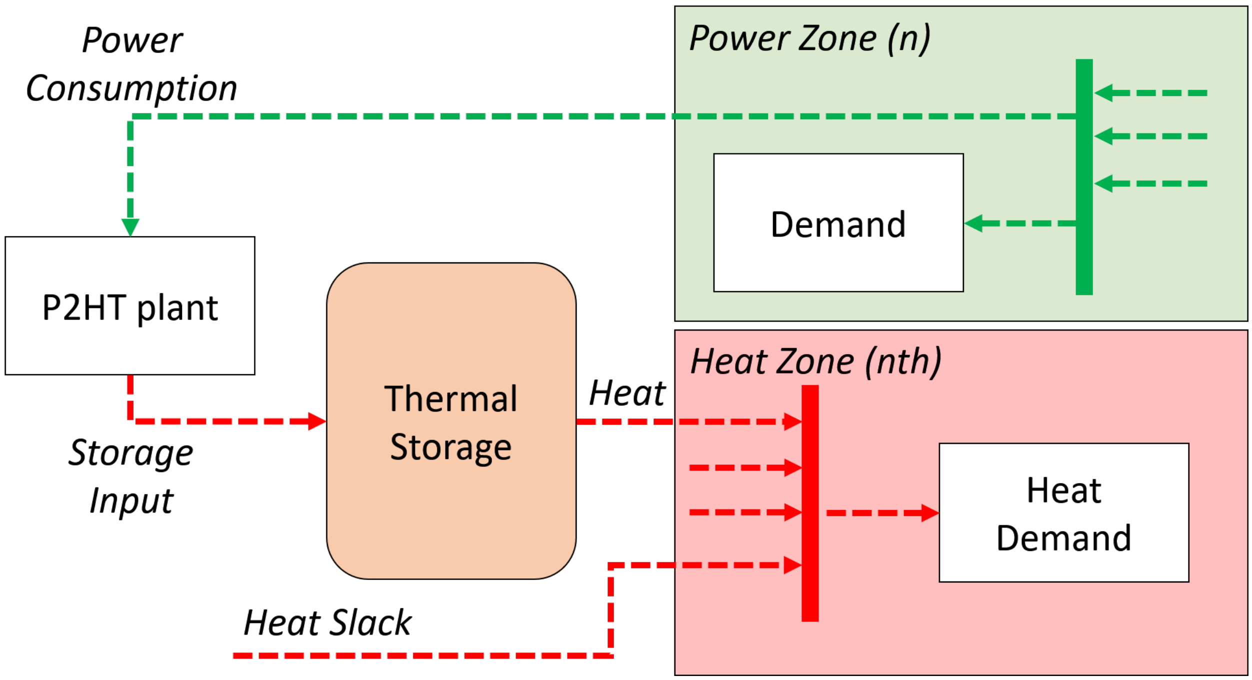
Heat storage is modeled in a similar way as electric storage as follows:

Heat Storage balance:

$$\begin{aligned} StorageLevel_{th,i-1} + StorageInput_{th,i} \cdot TimeStep = \\ StorageLevel_{th,i} + Heat_{th,i} \cdot TimeStep \\ + StorageSelfDischarge_{th} \cdot StorageLevel_{th,i} \cdot TimeStep/24 \end{aligned}$$

Storage level must be above a minimum and below storage capacity:

$$StorageMinimum_{th} \cdot Nunits_{th} \leq StorageLevel_{chp,i} \leq StorageCapacity_{th} \cdot Nunits_{th}$$



Emission limits

The operating schedule also needs to take into account any cap on the emissions (not only CO2) from the generation units existing in each node:

$$\sum_{u} (Power_{u,i} \cdot EmisionRate_{u,p} \cdot TimeStep \cdot Location_{u,n}) \leq EmisionMaximum_{n,p}$$

It is important to note that the emission cap is applied to each optimisation horizon: if a rolling horizon of one day is adopted for the simulation, the cap will be applied to all days instead of the whole year.

Network-related constraints

The flow of power between nodes is limited by the capacities of the transmission lines:

$$FlowMinimum_{l,i} \leq Flow_{l,i}$$
$$Flow_{l,i} \leq FlowMaximum_{l,i}$$

In this model a simple Net Transfer Capacity (NTC) between countries approach is followed. No DC power flow or Locational Marginal Pricing (LMP) model is implemented.

Load shedding

If load shedding is allowed in a node, the amount of shed load is limited by the shedding capacity contracted on that particular node (e.g. through interruptible industrial contracts)

$$ShedLoad_{n,i} \leq LoadShedding_{n,i}$$

5.5.3 Linear Program (LP) optimization

A possible simplification of the model is to run it as a LP instead of MILP. In that case, the LPFormulation parameter needs to be set to 1 (and to 0 otherwise).

In that case, the commitment status variables Commited, StartUp and ShutDown are not defined as binary and Commited is set smaller than 1. The equations describing the cost of starting up and shutting down are ignored, as well as the ones enforcing minimum up and down times.

5.5.4 Mid Term Scheduling (MTS)

As will be explained in more details hereunder, MTS allows to pre-define storage levels during the whole year based on a simplified equations.

Model in MTS mode

When MTS is activated, some equations are dropped/modified. MTS mode is activated by setting parameter MTS to 1. In this configuration, all equations concerning unit commitment are not considered and the binary variables Committed, StartUp and ShutDown are not defined. The following constraints are therefore ignored:

- The commitment equations
- The minimum Up and Down times equations
- The Ramp up and Ramp down limitation equations

Also, due to the absence of the variable Committed, some equations are modified. Firstly, the cost equation is modified as follow:

$$
\begin{aligned}
min\Big[&\sum_{u,i} CostFixed_u \cdot TimeStep \\
&+\sum_{u,i}(CostStartUpH_{u,i} + CostShutDownH_{u,i}) \\
&+\sum_{u,i}(CostRampUpH_{u,i} + CostRampDownH_{u,i}) \\
&+\sum_{u,i} CostVariable_{u,i} \cdot Power_{u,i} \cdot TimeStep \\
&+\sum_{hu,i} CostVariable_{hu,i} \cdot Heat_{hu,i} \cdot TimeStep \\
&+\sum_{l,i} PriceTransimission_{l,i} \cdot Flow_{l,i} \cdot TimeStep \\
&+\sum_{n,i} CostLoadShedding_{i,n} \cdot ShedLoad_{i,n} \cdot TimeStep \\
&+\sum_{th,i} CostHeatSlack_{th,i} \cdot HeatSlack_{th,i} \cdot TimeStep) \\
&+\sum_{p2h2,i} CostH2Slack_{p2h2,i} \cdot StorageSlack_{p2h2,i} \cdot TimeStep) \\
&+\sum_{chp,i} CostVariable_{chp,i} \cdot CHPPowerLossFactor_{chp} \cdot Heat_{chp,i} \cdot TimeStep) \\
&+\sum_{i,n} VOLL_{Power} \cdot (LL_{MaxPower,i,n} + LL_{MinPower,i,n}) \cdot TimeStep \\
&+\sum_{i,n} 0.8 \cdot VOLL_{Reserve} \cdot (LL_{2U,i,n} + LL_{2D,i,n} + LL_{3U,i,n}) \cdot TimeStep \\
&+\sum_{s,i} CostOfSpillage \cdot spillage_{s,i} \\
&+\sum_{s,i} WaterValue \cdot WaterSlack_s\Big]
\end{aligned}
$$

The upwards and downwards secondary reserve capabilities of units becomes:

$$
\begin{aligned}
Reserve_{2U\,u,i} \leq & PowerCapacity_u \cdot AvailabilityFactor_{u,i} \cdot (1 - OutageFactor_{u,i}) \quad (5.22) \\
& - Power_{u,i} \quad (5.23) \\
Reserve_{2D\,u,i} \leq & \; Power_{u,i} + (StorageChargingCapacity_u \cdot Nunits_u - StorageInput_{u,i}) \quad (5.24)
\end{aligned}
$$

Also the non spinning reserve is modified:

$$Reserve_{3U\,u,i} \leq Nunits_u \cdot QuickStartPower_{u,i} \cdot TimeStep$$

The output power available for each unit is now expressed as:

$$Power_{u,i} \leq PowerCapacity \cdot AvailibilityFactor \cdot (1 - OutageFactor)$$

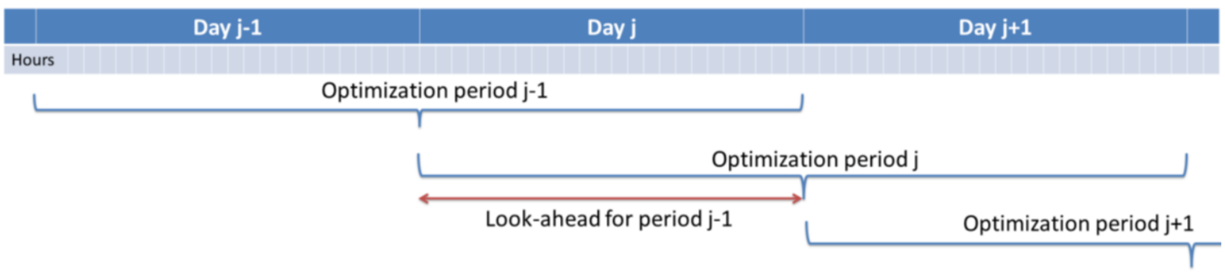
Finally, the maximum capacity of storage charging is:

$$StorageInput_{s,i} \leq StorageChargingCapacity_s \cdot Nunits_s$$

5.5.5 Rolling Horizon

The mathematical problem described in the previous sections could in principle be solved for a whole year split into time steps, but with all likelihood the problem would become extremely demanding in computational terms when attempting to solve the model with a realistically sized dataset. Therefore, the problem is split into smaller optimization problems that are run recursively throughout the year.

The following figure shows an example of such approach, in which the optimization horizon is two days, including a look-ahead (or overlap) period of one day. The initial values of the optimization for day j are the final values of the optimization of the previous day. The look-ahead period is modelled to avoid issues related to the end of the optimization period such as emptying the hydro reservoirs, or starting low-cost but non-flexible power plants. In this case, the optimization is performed over 48 hours, but only the first 24 hours are conserved.



The optimization horizon and overlap period can be adjusted by the user in the Dispa-SET configuration file. As a rule of thumb, the optimization horizon plus the overlap period should at least be twice the maximum duration of the time-dependent constraints (e.g. the minimum up and down times). In terms of computational efficiency, small power systems can be simulated with longer optimization horizons, while larger systems should reduce this horizon, the minimum being one day.

5.5.6 References

5.6 Model Formulations

Because of the constraints linked to computational efficiency and to data availability, it is not necessarily desirable to accurately model each individual unit in the power system. For that reason, Dispa-SET can operate under different modelling hypotheses and levels of complexity. In terms of formulation of the optimization problem, these include for example:

- A linear programming formulation, in which all units are clustered by technology
- An integer formulation in which a typical unit is considered for each technology and multiplied N times. The formulation allows taking into account constraints such as minimum up/down times, minimum load, etc.
- A binary formulation in which each power plant in the system is considered individually

The section describes the various clustering options and modeling formulations available in Dispa-SET. It is worthwhile to note that each clustering method and/or modelling formulation can be applied to the same reference dataset. This allows comparing the various methods in terms of computational efficiency, but also in terms of accuracy.

Four main formulations are currently available:

5.6.1 No clustering

In this case, the pre-processing tool does not modify the power plant input data. The user is allowed to cluster some power plants himself by defining the Nunits input variable in the power plants input csv file. Let us consider the following (incomplete) inputs as an example:

Unit	PowerCapacity	Nunits	Zone	Technology	Fuel
Maasvlakte	500	1	Z2	STUR	HRD
Diemen	430	1	Z2	COMC	GAS
CCGTs	400	6	Z2	COMC	GAS
Borssele	408	1	Z2	STUR	HRD
OCGT1	25	1	Z2	GTUR	GAS
TIHANGE 3	1000	1	Z1	STUR	NUC
DROGENBOS TGV	465	1	Z1	COMC	GAS
SISTERON	214	2	Z1	HDAM	WAT
SIERREUX	20	1	Z1	GTUR	GAS
ANGLEUR	30	1	Z1	GTUR	GAS
WindOn_Z1	200	1	Z1	WTON	WIN

In this example, there are a number of different unit types and two zones. Some power plants have Nunits=1, which implies that they will be considered individually in the optimization. Other power plants (CCGTS and SISTERON) are multiplied 6 and 2 times, respectively. This implies that the total capacity of CCGTs and SISTERON units is 2400 MW and 228 MW, respectively. Note that all unit characteristics in the input data (not appearing in the above table) should be defined for a single unit!

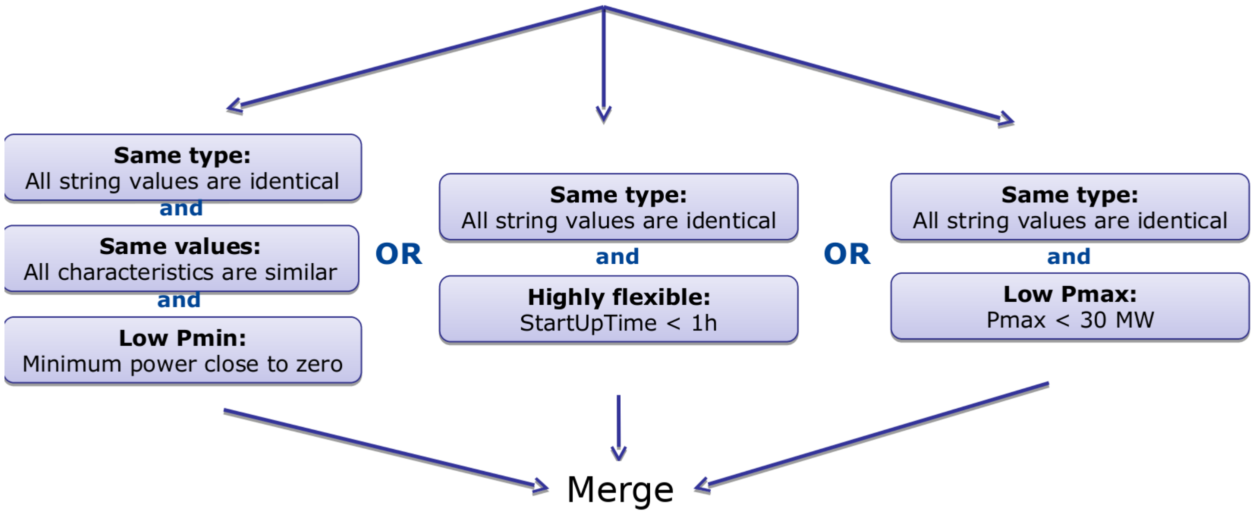
These two units are assigned an integer variable instead of a binary variable in the optimization. The solver successively starts 1, 2, 3, 4, etc. units with the exact same characteristics. In this approach, start-up costs, minimum up and down times, minimum part-load are considered, but with a significantly improved computational efficiency. The loss in accuracy resides in the hypothesis of identical characteritics for all units in a clustered group. This is however acceptable if not data is available at the individual power plant level, or if the complexity of the modeled system does not justify such a high disaggregation level.

5.6.2 Standard formulation

For computational efficiency reasons, it is useful to merge some of the original units into larger units. This reduces the number of continuous and binary variables and can, in some conditions, be performed without significant loss of simulation accuracy.

In the standard formulation (formerly call MILP formulation), the units that are either very small or very flexible are merged into larger units. Some of these units (e.g. the turbojets) indeed present a low capacity or a high flexibility: their output power does not exceed a few MW and/or they can reach full power in less than 15 minutes (i.e. less than the simulation time step). For these units, a unit commitment model with a time step of 1 hour is unnecessary and computationally inefficient. They are therefore merged into one single, highly flexible unit with averaged characteristics.

The condition for the merging of two units is a combination of subconditions regarding their type, maximum power, flexiblity and technical similarities. They are summarized in the figure below (NB: the thresholds are for indicative purpose only, they can be user-defined).



When two units are merged, the minimum and maximum capacities of new aggregated units (indicated by the star) are given by:

$$P^*_{min} = min(P_{j,min})$$

$$P^*_{max} = \sum_j (P_{j,min})$$

The last equation is also applied for the storage capacity or for the storage charging power.

The unit marginal (or variable cost) is given by:

$$Cost^*_{Variable} = \frac{\sum_j (P_{j,max} \cdot Cost_{Variable,j})}{P^*_{max}}$$

The start-up/shut-down costs are transformed into ramping costs (example with ramp-up):

$$Cost^*_{RampUp} = \frac{\sum_j (P_{j,max} \cdot Cost_{RampUp,j})}{P^*_{max}} + \frac{\sum_j (Cost_{StartUp,j})}{P^*_{max}}$$

Other characteristics, such as the plant efficiency, the minimum up/down times or the CO2 emissions are computed as a weighted averaged:

$$Efficiency^* = \frac{\sum_j (P_{j,max} \cdot Efficiency_j)}{P^*_{max}}$$

It should be noted that only very similar units are merged (i.e. their quantitative characteristics should be similar), which avoids errors due to excessive aggregation.

In the example provided in the above table the following would occur:

- SIERREUR and ANGLEUR would be merged because they are small and highly flexible.
- OCGT1 cannot be merged with SIERREUX and ANGLEUR since they don't belong to the same zone.
- Maasvlakte and Borssele are not merged, although they have the same technology, fuel and zone. This is because their size is significant and they flexibility is low.
- Diemen and CCGTS are merged only if their flexibility is high (i.e. they can start/stop or ramp to full load in less then one hour).

5.6.3 Integer clustering

In this formulation, all units of a similar technology, fuel and zone are clustered: a typical unit is defined by averaging the characteristics of all units belonging to the cluster. The total number of units is conserved, allowing a proper representation of constraints such as start-up costs, minimum up/down times or minimum stable load values. In the example provided above, the integer clustering would results into the following unit list:

Unit	PowerCapacity	Nunits	Zone	Technology	Fuel
Z2_STUR_HRD	454	2	Z2	STUR	HRD
Z2_COMC_GAS	404	7	Z2	COMC	GAS
OCGT1	25	1	Z2	GTUR	GAS
TIHANGE 3	1000	1	Z1	STUR	NUC
DROGENBOS TGV	465	1	Z1	COMC	GAS
SISTERON	214	2	Z1	HDAM	WAT
Z1_GTUR_GAZ	25	2	Z1	GTUR	GAS
WindOn_Z1	200	1	Z1	WTON	WIN

where the total capacity and number of units for each technology/fuel is conserved. More details regarding the formulation and the implementation of the integer clustering are available in[1]

5.6.4 LP clustering

Dispa-SET provides the possibility to generate the optimisation model as an LP problem (i.e. withtout the binary variables). In that case, the following constraints are removed since they can only be expressed in an MILP formulation:

- Minimum up and down times
- Start-up costs
- Minimum stable load

Since the start-up of individual units is not considered anymore, it is not useful to disaggrate them in the optimisation. All units of a similar technology, fuel and zone are merged into a single unit using the equations proposed in the previous sections. This formulation is used in the *Mid-term hydrothermal coordination* .

5.7 Implementation and interface

The typical step-by-step procedure to parametrize and run a DispaSET simulation is the following:

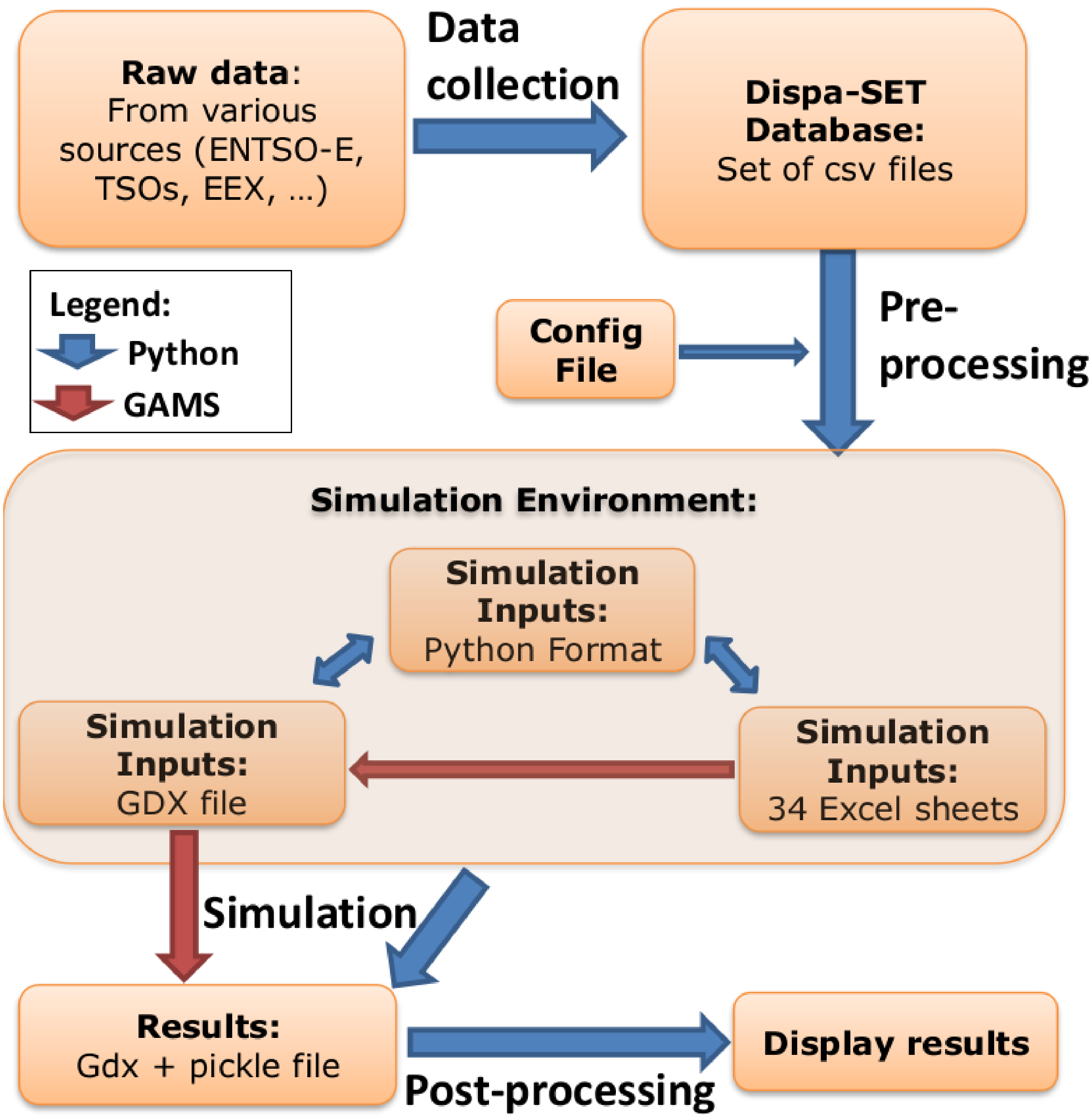
1. Fill the Dispa-SET database with properly formatted data (time series, power plant data, etc.)
2. Configure the simulation parameters (rolling horizon, data slicing) in the configuration file.
3. Generate the simulation environment which comprises the inputs of the optimisation
4. Open the GAMS simulation files (project: UCM.gpr and model: UCM_h.gms) and run the model.
5. Read and display the simulation results.

This section provides a detailed description of these steps and the corresponding data entities.

[1] Incorporating Operational Flexibility Into Electric Generation Planning: Impacts and Methods for System Design and Policy Analysis, Palmintier, B.S. (2012). Ph.D. Thesis, Engineering Systems Division, MIT

5.7.1 Resolution Flow Chart

The whole resolution process for a dispa-SET run is defined from the processing and formatting of the raw data to the generation of aggregated result plots and statistics. A flow chart of the consecutive data entities and processing steps is provided hereunder.



Each box in the flow chart corresponds to one data entity. The links between these data entities correspond to script written in Python or in GAMS. The different steps perform various tasks, which can be summarized by:

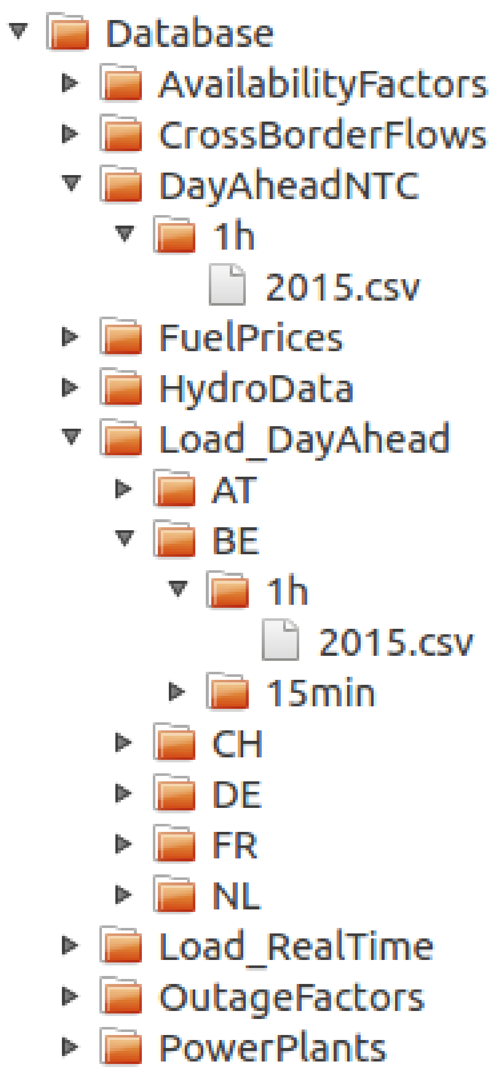
1. **Data collection:**
    - Read csv sheets, assemble data
    - Convert to the right format (timestep, units, etc).

- Define proper time index (duplicates not allowed)
- Write formatted input data to the Dispa-SET database

2. **Pre-processing:**
   - Read the config file
   - Slice the data to the required time range
   - Deal with missing data
   - Check data for consistency (min up/down times, startup times, etc.)
   - Calculate variable cost for each unit
   - Cluster units
   - Define scenario according to user inputs (curtailment, participation to reserve, amount of VRE, amount of storage, . . .)
   - Define initial state (basic merit-order dispatch)
   - Write the simulation environment to a user-defined folder
3. **Simulation environment and interoperability:**
   - **Self-consistent folder with all required files to run the simulation:**
     - GDX file
     - Input files in pickle format
     - Gams model files
4. **Simulation:**
   - The GAMS simulation file is run from the simulation environment folder
   - All results and inputs are saved within the simulation environment
5. **Post-processing:**
   - Reads the simulation results saved in the simulation environment
   - Aggregates the power generation and storage curves
   - Computates of yearly statistics
   - Generates plots

5.7.2 Dispa-SET database

The public version of Dispa-SET is released with a Database relative to the EU power system. The Dispa-SET input data is stored as csv file in directory structure. A link to the required data is then provided by the user in the configuration file.



The above figure shows a partially unfolded view of the database structure. In that example, data is provided for the day-ahead net transfer capacities for all lines in the EU, for the year 2015 and with a 1h time resolution. Time series are also provided for the day-ahead load forecast for Belgium in 2015 with 1h time resolution.

5.7.3 Configuration File

The excel config file is read at the beginning of the pre-processing phase. It provides general inputs for the simulation as well as links to the relevant data files in the database.

5.7.4 Simulation environment

This section describes the different simulation files, templates and scripts required to run the DispaSET model. For each simulation, these files are included into a single directory corresponding to a self-sufficient simulation environment.

A more comprehensive description of the files included in the simulation environment directory is provided hereunder.

UCM_h.gms and UCM.gpr

UCM_h.gms is the main GAMS model described in Chapter 1. A copy of this file is included in each simulation environment, allowing keeping track of the exact version of the model used for the simulation. The model must be run in GAMS and requires a proper input file (Inputs.gdx).

Requires:	Inputs.gdx	Input file for the simulation.
Generates:	Results.gdx	Simulation results in gdx format
.	Results.xlsx	Simulation results in xlsx format.

UCM.gpr is the GAMS project file which should be opened before UCM_h.gms.

Inputs.gdx

All the inputs of the model must be stored in the Inputs.gdx file since it is the only file read by the main GAMS model. This file is generated from the DispaSET template.

Requires:	InputDispa-SET – xxx.xlsx	DispaSET template files
Generates:		

5.7.5 Post-processing

Post-processing is implemented in the form of a series of functions to read the simulation inputs and results, to plot them, and to derive statistics.

The following values are computed:

- The total energy generated by each fuel, in each country.
- The total energy curtailed
- the total load shedding
- The overall country balance of the interconnection flows
- The total hours of congestion in each interconnection line
- The total amount of lost load, indicating (if not null) that the unit commitment problem was unfeasible at some hours
- The number of start-ups of power plants for each fuel

The following plots can be genrated:

- A dispatch plot (by fuel type) for each country
- A commitment status (ON/OFF) plot for all the unit in a given country
- The level (or state of charge) of all the storage units in a given country
- The overall power generation by fuel type for all countries (bar plot)

An example usage of these funciones is provided in the “Read_Results.ipynb” notebook.

5.8 Mid-term hydrothermal coordination

As discussed in previous sections the simulations depends on exogenous storage level profiles. These profiles have to be cogerent with the rest of the input parameters in order to ensure both the feasibility of the problem and accurate results.

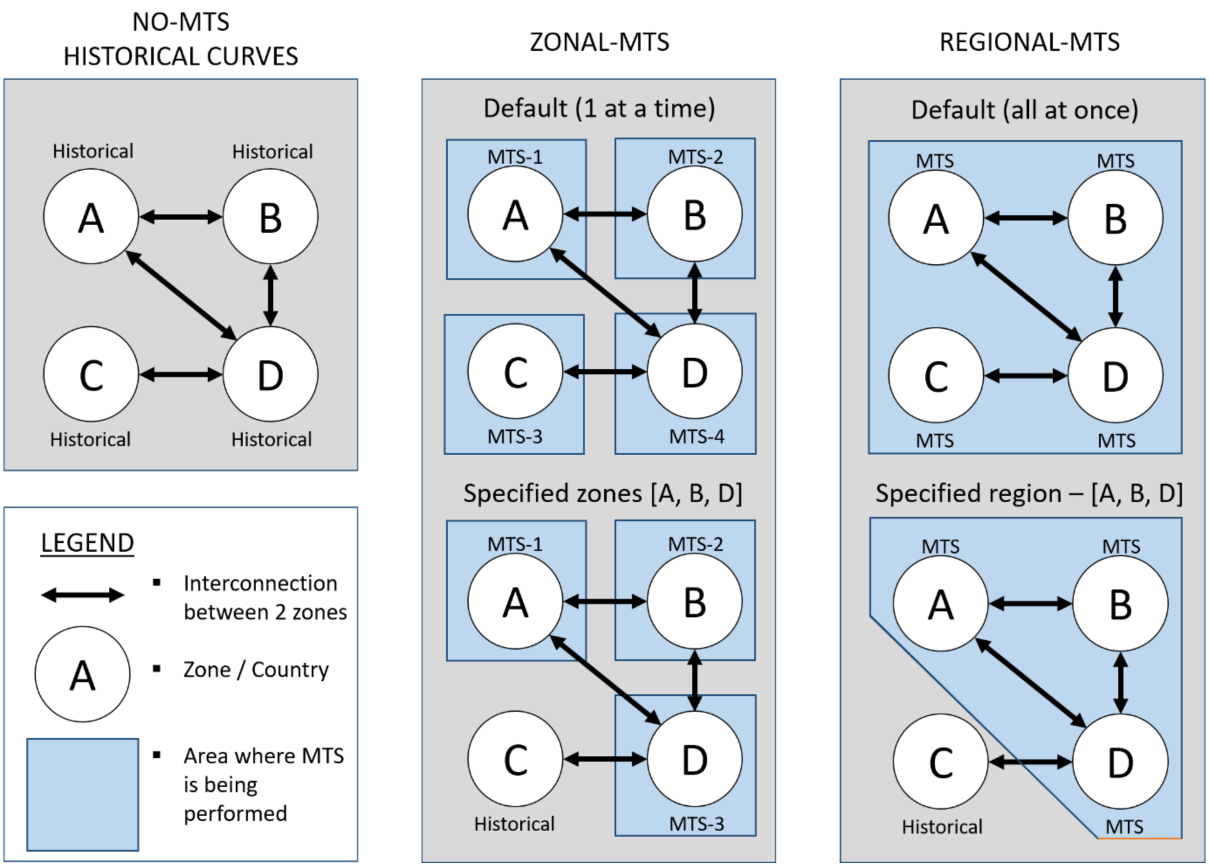
In many cases, collecting accurate and reliable historical storage levels and profiles in form of hourly timeseries might be a difficult or close to impossible task. In future scenarios storage levels are usually forecasted based on the historical data. The lack of such data also impacts the accurate modelling of such scenarios. In systems with high shares of hydro dams (HDAM) and pumped hydro storage (HPHS) units, such as Norway and Albania, this might have a huge impact on the overall results of the simulation.

In order to avoid this, Dispa-SET's Midterm Hydro-Thermal Scheduling (MTS) module represents a simplified version of the original MILP unit commitment and power dispatch model. This version is a simplified version of the linear programming formulation which allows perfect foresight and allocation of water resources for the whole optimization period and not only for the tactical horizon of each optimization step. This module enables quick calculation (later also referring as allocation) of reservoir levels which are then used as guidance curves (minimum level constraints) in one of the four main Dispa-SET formulations. The main options are:

- No-MTS, in which historical curves are used,
- Zonal-MTS, in which MTS is run for each Zone individually,
- Regional-MTS, in which MTS is run for two or more Zones from the selected region simultaneously.

5.8.1 MTS options

This section describes the above options available in the Dispa-SET. It is worthwhile to note that each MTS method and/or modelling formulation can be applied to the same input dataset. This allows comparing the various methods in terms of computational efficiency, but also in terms of accuracy. Graphical summary of MTS options is available in Figure:



Note: The MTS optimization (process) is being executed in the preprocessing phase. Here the simplified LP optimization estimates the reservoir levels for the entire year. These newly computed reservoir levels are then imposed as minimum level constraint used in the last time interval of the rolling horizon. As preprocessing includes LP optimization, it might take a while to complete and will be highly dependent on the number of selected zones (the more zones are selected the longer it will take to build the model). Depending on the operating system, command prompt may pop-up and interrupt other processes several times.

NO-MTS (Historical Curves)

HISTORICAL is the standard formulation of the Dispa-SET model. All reservoir levels are imposed as external time-series in form of scaled reservoir levels (in range from 0 - 1). These reservoir levels are used as minimum level constraint in the last time interval of the rolling horizon.

ZONAL-MTS

ZONAL formulation can be used in two ways: 1) If not specified otherwise (default option), MTS will be run for each of the selected zones individually. For a system of four zones configured as shown in figure 1, each of the zones A, B, C & D MTS will be run once. In this formulation NTC capacities are not considered. Instead, default cross border flows from the database are used. . 2) Zones where MTS should be run can also be specified in the list of zones. In this case, MTS module will only be run in the specified zones. In our example with four zones, zones A, B & D are selected for the MTS, while zone C keeps the default values from the database.

REGIONAL-MTS

REGIONAL formulation can be used in the same way as zonal formulation.

1) If not specified otherwise, MTS will be run once, for the whole region at once. For the system of four zones configured as shown in figure 1, zones A, B, C & D will be regarded as a region. In this formulation MTS will compute reservoir levels based on additional criteria such as available net transfer capacities (NTC) instead of historical cross border flows (CBF).

2) Region where MTS should be run can also be specified in the list of zones contained within one region. In this case MTS module will be run only in the selected region. In our example zones A, B & D. Zone C keeps the default values from the database. This formulation generates even more accurate results compared to the zonal formulation but might increase the memory usage and computation time significantly.

5.8.2 How to call MTS

MTS module is called automatically if the relevant parameters indicate the use of it. During the build phase Dispa-SET will read the MTS parameters from the Config file (either .xlsx or .yml), execute the MTS module and build SimData with newly computed reservoir levels. Additional options also allow selection of specific zones and ploting the difference between the historical and newly computed reservoir levels (this is only useful for debugging purposes). The latter is triggered if the *build_simulation* functions is called with the flag *mts_plot=True*.

Selection of MTS options inside the Config file:

Config.xlsx	Config.yml
Hydro scheduling	HydroScheduling:
• Off • Regional • Zonal	• ‘Off’ • ‘Regional’ • ‘Zonal’
Hydro scheduling horizon	HydroSchedulingHorizon:
• Annual • Stop-date driven	• ‘Annual’ • ‘Stop-date driven’
Initial/Final reservoir level	InitialFinalReservoirLevel:
• TRUE • FALSE	• 1.0 • 0.0
• Initial level & <- Final level	ReservoirLevelInitial: & ReservoirLevelFinal:
• ‘’ • 0 - 1	• ‘’ • 0 - 1

5.8.3 Examples

Computing reservoir levels by using the MTS module increases the accuracy of HDAM and HPHS units and avoids infeasible ramping up/down rates proposed by the historical levels. A good example of one such case is when the difference between initial and final reservoir levels is higher than the sum of all the inflows during that time horizon.

Lets assume that ReservoirLevelInital = 1 MWh , ReservoirLevelFinal = 30 MWh, Pmax = 1 MW, InFlows = 1 MW per hour which totals 24 MWh for horizon length of one day. In this case the reservoir cannot reach the storage target as it is constrained by the generation capacity.

This would produce infeasible solution which would then have a price of water slackof 100€/MWh imposed on the difference between historical and calculated reservoir levels in the last time interval of the rolling horizon (an imaginary well that fills the reservoir). In our example this would then amount to 30MWh – 1MWh – 24MWh = 5 MW *100 €/MWh = 500€.

This might still be the case even if MTS is turned on. Thus, **Hydro scheduling horizon** option inside the Config files provides additional flexibility for running the MTS:

- **Stop-date driven** runs the MTS only between the selected start and stop dates, while
- **Annual** runs the MTS between 01.01. and 31.12. of the selected year.

Initial/Final reservoir levels can be imposed as TRUE/FALSE statement. When TRUE Initial and Final reservoir levels can be imposed externally and should always be in range between 0 and 1 (0 completely empty reservoir and 1 for 100% full reservoir). This option will override initial and final reservoir levels for all hydro units in all selected zones. This might be useful when analysing countries/zones with data scarcity. When FALSE historical values for Initial and Final reservoir levels will be considered.

5.9 Case Studies

In the past years, Dispa-SET has been used in various scientific works covering different geographical areas and with different focus points. It is originally designed to simulate EU countries (one node per country) but has also been applied to other regions such as Western Africa, Bolivia or the Balkans. The model includes the constraints linked to each generation unit (min/max power, ramping rates, efficiencies, storage capacities, etc.), to the interconnections and to the power and thermal demands. It uses high-resolution time series for the demands, renewable generation and outages in each simulated country.

The case studies for which scientific analysis has been carried out are summarized hereunder:

5.9.1 Dispa-SET for the EU28

Description

Dispa-SET is provided with a ready-to-use dataset of the EU28 (+Norway +Switzerland) power system. A detailed description of the model and of the selected input data is available in[1] and[2].

The power plants in the model are represented as a cluster of units powered by the same fuel type and technology. They can be modelled together with a large number of RES units with separate hourly distribution curves.

Features

The model is expressed as an optimization problem. Continuous variables include the individual unit dispatched power, the shedded load and the curtailed power generation. The binary variables are the commitment status of each unit. The main model features can be summarized as follows:

- Minimum and maximum power for each unit

[1] Kavvadias, K., Hidalgo Gonzalez, I., Zucker, A., & Quoilin, S. (2018). Integrated modelling of future EU power and heat systems - The Dispa-SET v2.2 open-source model (EUR 29085 EN). Luxembourg: European Commission.

[2] Matija Pavičević, Wouter Nijs, Konstantinos Kavvadias, Sylvain Quoilin, Modelling flexible power demand and supply in the EU power system: soft-linking between JRC-EU-TIMES and the open-source Dispa-SET model, ECOS 2019

- Power plant ramping limits
- Reserves up and down
- Minimum up/down times
- Load Shedding
- Curtailment
- Pumped-hydro storage
- Non-dispatchable units (e.g. wind turbines, run-of-river, etc.)
- Start-up, ramping and no-load costs
- Multi-nodes with capacity constraints on the lines (congestion)
- Constraints on the targets for renewables and/or CO2 emissions
- Yearly schedules for the outages (forced and planned) of each units
- CHP power plants and thermal storage
- Integer clustering: a representative unit is considered for each technology and multiplied N times.

The demand is assumed to be inelastic to the price signal. The MILP objective function is therefore the total generation cost over the optimization period.

Run the EU model

A specific config file is provided with the standard Dispa-SET installation (starting from v2.3). After installing Dispa-SET and checking that everything is fine, you can run the EU model in different ways:

From the command line (if Dispa-SET was properly installed):

```
dispaset -c ConfigFiles/ConfigEU.xlsx build simulate
```

Using the Dispa-SET Api and the provided example script:

scriptsbuild_and_run_EU_model.py

Documentation

The general documentation of the Dispa-SET model and the stable releases are available on the main Dispa-SET website: http://www.dispaset.eu

Licence

Dispa-SET is a free software licensed under the "European Union Public Licence" EUPL v1.2. It can be redistributed and/or modified under the terms of this license.

Important results

Main developpers

- Sylvain Quoilin (University of Liège, KU Leuven)
- Konstantinos Kavvadias (European Commission, Institute for Energy and Transport)

- Matija Pavičević (KU Leuven)

References

More details regarding the model and its implementation are available in the following publications

5.9.2 Dispa-SET for the Balkans region

Description

This is input data of the Dispa-SET model, applied to the Balkans Region

Countries included in different scenarios are show in the table[1][2][3][4] :

Countries	2010	2015	2020	2030
Albania	[O]	[O]	[O]	[O]
Bosnia and Herzegovina	[O]	[O]	[O]	[O]
Croatia	[X]	[O]	[O]	[O]
Kosovo	[O]	[O]	[O]	[O]
Macedonia	[O]	[O]	[O]	[O]
Montenegro	[O]	[O]	[O]	[O]
Serbia	[O]	[O]	[O]	[O]
Slovenia	[X]	[O]	[O]	[O]

The model has the ability to describe every single unit, or a cluster of units powered by the same fuel type and technology, with a high level of detail can be modelled together with a large number of RES units with separate hourly distribution curves.

Features

The model is expressed as an optimization problem. Continuous variables include the individual unit dispatched power, the shedded load and the curtailed power generation. The binary variables are the commitment status of each unit. The main model features can be summarized as follows:

- Minimum and maximum power for each unit
- Power plant ramping limits
- Reserves up and down
- Minimum up/down times
- Load Shedding
- Curtailment
- Pumped-hydro storage

[1] Pavičević, M., Kavvadias, K. & Quoilin, S. (2018). Impact of model formulation on power system simulations - Example with the Dispa-SET Balkans model, EMP-E conference 2018: Modelling Clean Energy Pathways, Brussels.

[2] Pavičević, M., Quoilin, S. & Pukšec, T., (2018). Comparison of Different Power Plant Clustering Approaches for Modeling Future Power Systems, Proceedings of the 3rd SEE SDEWES Conference, Novi Sad.

[3] Tomić, I., Pavičević, M., Quoilin, S., Zucker, A., Pukšec, T., Krajačić. G. & Duić, N., (2017). Applying the Dispa-SET model on the seven countries from the South East Europe. 8th Energy Planning and Modeling of Energy Systems-Meeting, Belgrade. https://bib.irb.hr/prikazi-rad?rad=901595

[4] Pavičević, M., Tomić, I., Quoilin, S., Zucker, A., Pukšec, T., Krajačić. G. & Duić, N., (2017). Applying the Dispa-SET model on the Western Balkans power systems. Proceedings of the 2017 12th SDEWES Conference, Dubrovnik. http://hdl.handle.net/2268/215095

- Non-dispatchable units (e.g. wind turbines, run-of-river, etc.)
- Start-up, ramping and no-load costs
- Multi-nodes with capacity constraints on the lines (congestion)
- Constraints on the targets for renewables and/or CO2 emissions
- Yearly schedules for the outages (forced and planned) of each units
- CHP power plants and thermal storage

The demand is assumed to be inelastic to the price signal. The MILP objective function is therefore the total generation cost over the optimization period.

Quick start

If you want to download the latest version from github for use or development purposes, make sure that you have git and the [anaconda distribution](https://www.continuum.io/downloads) installed and type the following:

```
git clone https://github.com/energy-modelling-toolkit/Dispa-SET.git
cd Dispa-SET
conda env create  # Automatically creates environment based on environment.yml
source activate dispaset # in Windows: activate dispaset
pip install -e . # Install editable local version
```

The above commands create a dedicated environment so that your anconda configuration remains clean from the required dependencies installed.

To check that everything runs fine, you can build and run a test case by typing:

```
dispaset -c ConfigFiles/ConfigTest.xlsx build simulate
```

Make sure that the path is changed to local Dispa-SET folder in folowing scripts (the procedure is provided in the scripts):

```
build_and_run.py
read_results.py
```

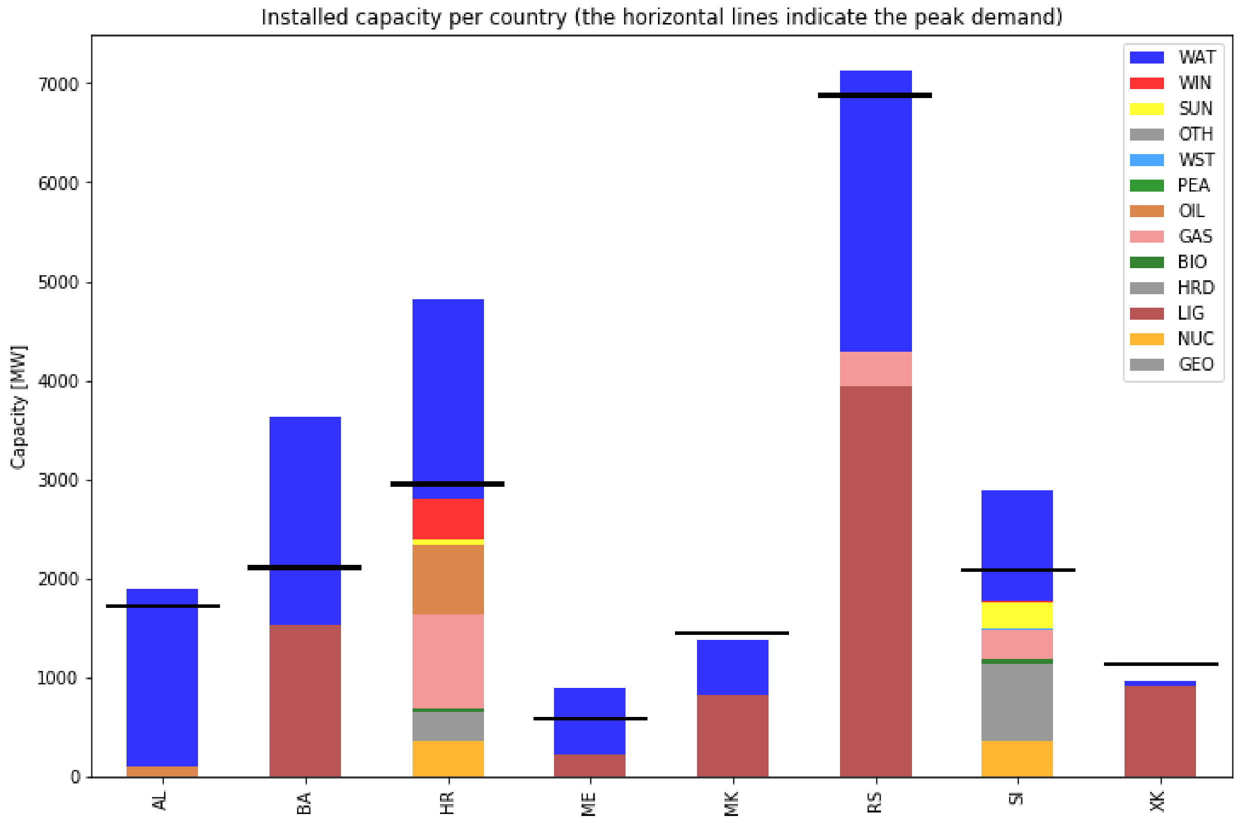
Documentation

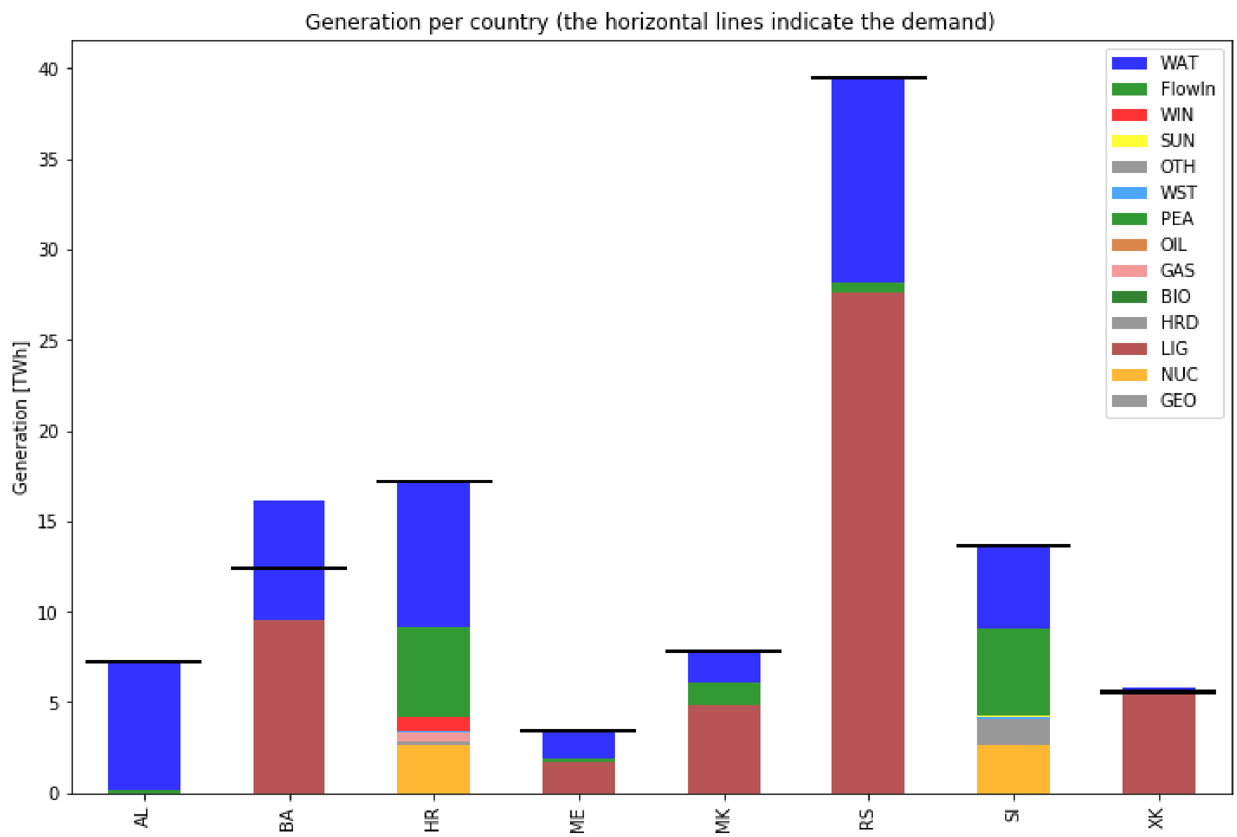
The general documentation of the Dispa-SET model and the stable releases are available on the main Dispa-SET website: http://www.dispaset.eu

Licence

Dispa-SET is a free software licensed under the “European Union Public Licence” EUPL v1.2. It can be redistributed and/or modified under the terms of this license.

Important results





Main developpers

- Matija Pavičević (KU Leuven) - gathered and analysed the data, performed the computations, analysed and verified the results
- Sylvain Quoilin (University of Liège, KU Leuven) - designed the model and the computational framework, verified the results
- Andreas Zucker (Joint Research Centre, European Commission) - supervised the whole process

References

More details regarding the model and its implementation are available in the following publications

Other contributors

- Ivan Tomić (University of Zagreb) - gathered and analysed the initial data
- Tomislav Pukšec (University of Zagreb) - analysed the inital results
- Goran Krajačić (University of Zagreb) - supervised inital project
- Neven Duić (University of Zagreb) - supervised inital project

5.9.3 Dispa-SET for the Belarus

Description

Planning the future: Integrating renewable energy sources in the Belarusian power system

Background

The Belarusian energy sector is mainly running on fossil fuels. Approximately two third of the country's energy production is covered by natural gas, which is mainly imported from Russia. Therefore, increasing the share of renewables in the energy balance has become one of the priority areas of the economic policy of the Belarusian government. In this regard, the objective of this work is the development of smart power and heating systems that can handle increased shares of renewable energy in the Belarusian system. This research focuses on several aspects such as balancing issues, flexibility requirements and congestion management in Belarusian grid.

Methods and Features

The Belarusian energy system has been modeled in Dispa-SET, an open-source unit commitment and optimal dispatch model focused on the balancing and flexibility problems in European grids. A reference and several future scenarios with high share of renewable energy sources were created. This case study analysis provides insights on the ability to integrate as much renewables as possible into the system and to check its impact on the price of runing the system.

The model is expressed as an optimization problem. Continuous variables include the individual unit dispatched power, the shedded load and the curtailed power generation. The binary variables are the commitment status of each unit. The main model features can be summarized as follows:

- Minimum and maximum power for each unit
- Power plant ramping limits
- Reserves up and down
- Minimum up/down times
- Load Shedding
- Curtailment
- Pumped-hydro storage
- Non-dispatchable units (e.g. wind turbines, run-of-river, etc.)
- Start-up, ramping and no-load costs
- Multi-nodes with capacity constraints on the lines (congestion)
- Constraints on the targets for renewables and/or CO2 emissions
- Yearly schedules for the outages (forced and planned) of each units
- CHP power plants and thermal storage

The demand is assumed to be inelastic to the price signal. The MILP objective function is therefore the total generation cost over the optimization period.

Quick start

If you want to download the latest version from github for use or development purposes, make sure that you have git and the [anaconda distribution](https://www.continuum.io/downloads) installed and type the following:

```
git clone https://github.com/energy-modelling-toolkit/Dispa-SET.git
cd Dispa-SET
conda env create  # Automatically creates environment based on environment.yml
source activate dispaset # in Windows: activate dispaset
pip install -e . # Install editable local version
```

The above commands create a dedicated environment so that your anconda configuration remains clean from the required dependencies installed.

To check that everything runs fine, you can build and run a test case by typing:

```
dispaset -c ConfigFiles/ConfigTest.xlsx build simulate
```

Make sure that the path is changed to local Dispa-SET folder in folowing scripts (the procedure is provided in the scripts):

```
build_and_run.py
read_results.py
```

Documentation

The general documentation of the Dispa-SET model and the stable releases are available on the main Dispa-SET website: http://www.dispaset.eu

Licence

Dispa-SET is a free software licensed under the "European Union Public Licence" EUPL v1.2. It can be redistributed and/or modified under the terms of this license.

Results

In this model the potential for integration of renewables in the power system of Belarus was examined. The results of this study indicate that the current grid infrastructure can utilize up to 30% of the energy generated by renewables without causing any balancing and stability issues while applying heat pumps, thermal storage and bio-waste-based technologies.

Conclusions

The analysis performed in this work has demonstrated that the utilization of renewables could greatly reduce the use of fossil fuels and hence reduce the annual CO2 emissions by about 5 million tons in the Belarusian energy sector. This way the high dependence on external energy markets should decrease. The designed scenarios should help to realize a Belarusian Energy and Environmental Policy where the share of renewables should reach 30%.

Highlights

Main developpers

- Matija Pavičević (KU Leuven) - gathered and analysed the data, performed the computations, analysed and verified the results
- Darya Muslina (Belarusian National Technical University) - gathered and analysed the data

- Yuliya Stanetskaya (Belarusian National Technical University) - gathered and analysed the data

References

More details regarding the model and its implementation are available in the following publications

5.9.4 External links

- EU28 DispaSET applied to the EU28 member states
- Balkans Western Balkans and neighbouring countries
- *Belarus* Planning the future: Integrating renewable energy sources in the Belarusian power system
- *Belgium* Coupling a power system model to a building model to evaluate the flexibility potential of DSM at country level
- *Bolivia* Techno-economic assessment of hing renewable energy source penetration in the Bolivian interconnected electric system
- *Central Europe* Evaluating flexibility and adequacy in future EU power systems: model coupling and long-term forecasting
- *Netherlands* Evaluating the impact of EV charging demand on the Dutch energy system

5.10 dispaset package

5.10.1 Subpackages

dispaset.preprocessing package

Submodules

dispaset.preprocessing.data_check module

dispaset.preprocessing.data_handler module

dispaset.preprocessing.preprocessing module

dispaset.preprocessing.utils module

Module contents

dispaset.postprocessing package

Submodules

dispaset.postprocessing.postprocessing module

Module contents

dispaset.misc package

Submodules

dispaset.misc.colorstreamhandler module

dispaset.misc.gdx_handler module

dispaset.misc.str_handler module

Module contents

5.10.2 Submodules

5.10.3 dispaset.solve module

5.10.4 Module contents

CHAPTER 6

Indices and tables

- genindex
- modindex
- search