e-HIGHWAY 2050

Modular Development Plan of the Pan-European Transmission System 2050

Contract number	308908	Instrument	Collaborative Project
Start date	1st of September 2012	Duration	40 months
WP 8	Enhanced Pan-European Transmission Planning Methodology		
D8.2.b	Demand/Generation scenarios prototype		

Revision: 1.1

Due date of delivery: December 2014

		Date & Visa
Written by	Baptiste SEGUINOT, RTE	17/12/2014
Checked by	Patrick PANCIATICI, WP8 Leader Camille PACHE, RTE Jean MAEGHT, RTE Alessandro ZANI, RSE	05/01/2015
Validated by	Brahim BETRAOUI, RTE Gérald SANCHIS, Coordinator	22/01/2015

Project co-funded by the European Commission within the Seventh Framework Program		
Dissemination Level		
PU	Public	X
PP	Restricted to other program participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Document information

General purpose

This document is the deliverable D8.2.b of the e-Highway2050 project. It contains the description of the prototype developed in the framework of task 8.2, named "*Computation of Demand and Generation Time Series*".

Change status

Revision	Date	Changes description	Authors
V0.0	11/12/2014	Initial version	B. SEGUINOT
V1.0	17/12/2014	Corrections by WP8 partners	B. SEGUINOT

EXECUTIVE SUMMARY

This document is the deliverable D8.2.b of the e-Highway2050 project. It contains the description of the prototype developed in the framework of task 8.2, named “Computation of Demand and Generation Time Series”.

The developed methodology is based on four modules:

- A maintenance scheduling heuristic, which plans the maintenance of the thermal power plants.
- A time series generator, which samples new possible behaviours of the uncertain phenomena which affect the power system (intermittent renewable generation, demand, hydrological inflows and thermal plant outages).
- A routine which allocates the hydro resources among the 52 weeks so as to decompose the yearly problem.
- An hourly adequacy model with a complete description of the hydro and thermal power plants, extra-European exchanges and demand response programs which computes the hydro-thermal dispatch of the system.

The prototype has been implemented using both Matlab and AMPL languages.

The method built in the framework of this task is discussed in deliverable D.8.2a, “Enhanced methodology for the computation of demand/generation scenarios”. The present document describes the prototype developed to carry out this work.

TABLE OF CONTENT

List of figures

1. Introduction

1.1. *High level functional description*

The prototype implemented in task 8.2 converts macro-assumptions on the power system into balanced time series of consumption and generation. Typically, gross values of the installed generation capacities of different sources (e.g.: nuclear, gas, wind, etc.) and yearly consumption of different countries are transformed into time series which describe the behaviour of the power system under operation.

To do so, the operation of the power system is simulated via an adequacy model. The optimal (less-costly) dispatch of the hydro and thermal power plants which balances the load is calculated.

A sequential approach is used in order to consider the chronology of the problem. More precisely, the behaviour of the different elements of the power system is described with time series over one year and with an hourly time step.

Moreover, a Monte Carlo approach has been retained in order to take into account the stochasticity of the problem. Simulations are run numerous times with different samples of the uncertain inputs (demand, renewable generation, hydrological inflows and thermal plant outages). This approach requires first to generate new time series of the uncertain phenomena affecting the power system.

The computed time series of generation and demand are then used in the following steps of the TEP of work package (WP) 8. They are reference injections which serve as a base to design the future network expansions.

The method built in the framework of the project is discussed in deliverable D.8.2a, “Enhanced methodology for the computation of demand/generation scenarios”. The present document describes the prototype developed to carry out this work.

1.2. *Modules developed*

The architecture of the developed tool is presented in Figure 1.

- In a first module, the maintenance of thermal units is planned in adequacy with the needs of the power system.
- In a second module, called time series (TS) generator, the time series related to the considered Monte Carlo year are generated.
- A third module allocates the hydrological resources among the 52 weeks of the year. It permits the decomposition of the yearly problem into 52 sub-problems.
- Finally, the hourly hydro-thermal dispatch is computed in a fourth module. A mixed-integer linear program (MILP) is solved for each week of the year. It includes a detailed model of the thermal units with unit commitment (UC) as well as a description of the hydro units, extra-European exchanges and demand response (DR).

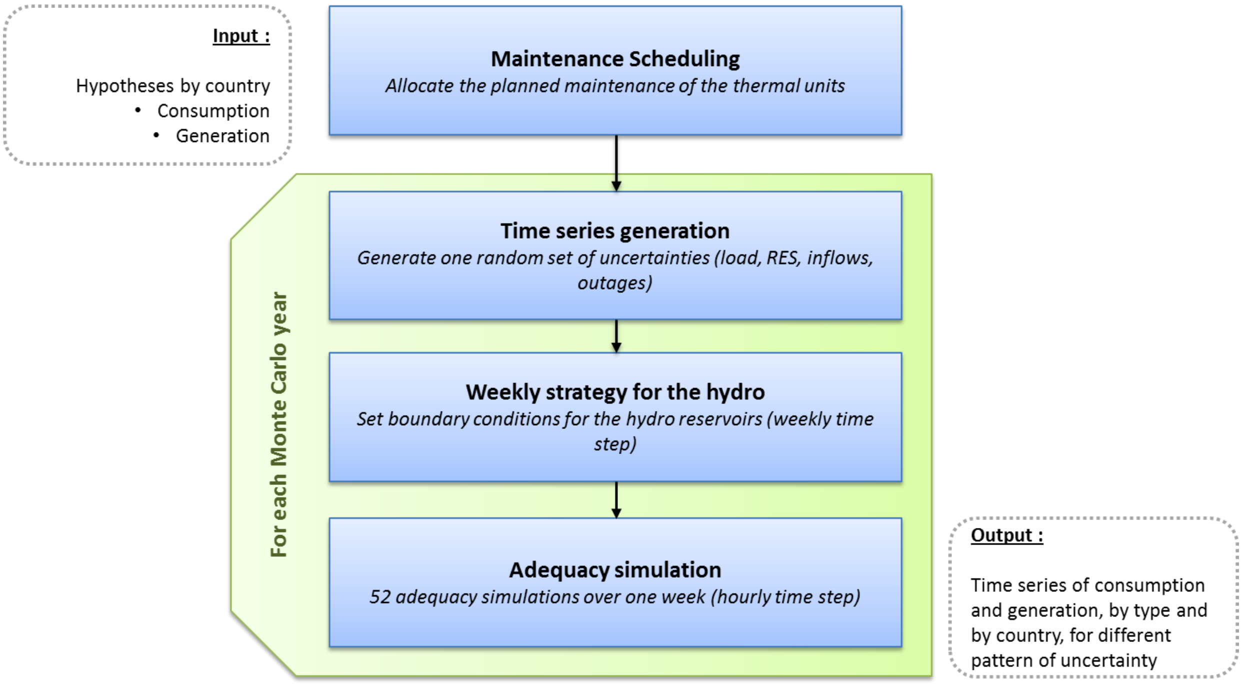

Input :
Hypotheses by country
• Consumption
• Generation
Maintenance Scheduling
Allocate the planned maintenance of the thermal units
For each Monte Carlo year
Time series generation
Generate one random set of uncertainties (load, RES, inflows, outages)
Weekly strategy for the hydro
Set boundary conditions for the hydro reservoirs (weekly time step)
Adequacy simulation
52 adequacy simulations over one week (hourly time step)
Output :
Time series of consumption and generation, by type and by country, for different pattern of uncertainty

Figure 1. Architecture of the adequacy simulator

A Monte Carlo approach has been retained to take into account the stochasticity of the problem and to study the response of the power system to diverse possible events. Consequently, the three last blocks of Figure 1 are repeated numerous times.

1.3. Implementation

The two first modules presented above have been prototyped with Matlab.

Note that the TS generator is also associated to a time series analyser which learns the intrinsic characteristics of each time series from historical data. The TS analyser is as well coded in Matlab.

The two last modules integrate optimisation problems and have been prototyped with AMPL.

2. Data organisation

The methodology of WP8 involves different scenarios, time horizons and Monte Carlo years and therefore necessitates handling large volumes of data.

The folder hierarchy used to classify the data is presented in this section.

First, the two following folders are common to all the study:

- ✓ Models database: This folder contains, for different areas and stochastic phenomena, the models of time series. The models are calibrated on the base of historical data and are used by the TS generator to sample new sets of time series of the uncertainties.
- ✓ General inputs: Contains the references of the model required in the current study and the associated correlation matrices.

Then, each scenario and time horizon is described with the two following folders:

- ✓ Description of the power system: contains the macro-assumptions of the given scenario and time horizon, in *.csv* format. The *.csv* files are the inputs of task 8.2 and contain, for each country, information on their consumption, generation fleet, hydrological resources, external exchanges and DR programs.
- ✓ Power system data: contains the macro-assumption of the given scenario and time horizon, in a format readable by AMPL (*.dat*). This folder includes also the maintenance schedule, which is common to each Monte Carlo year and is therefore unique for a given scenario and time horizon.

Finally, for each scenario, time horizon and Monte Carlo year, the three following folders are used:

- ✓ Monte Carlo year's data: This folder contains the stochastic inputs related to the given scenario, time horizon and Monte Carlo year.
- ✓ Intermediary data: Makes the link between the two last modules of the methodology. This folder contains the weekly volume target for each hydro reservoir of the system.
- ✓ Outputs: time series of generation and demand of the considered scenario, time horizon and Monte Carlo year. Outputs of the methodology of task 8.2.

The paths of the folders are defined in a file (*setup.csv*).

Section 3 presents the TS analyser, whose goal is to supply the database of models. Section 4 describes the algorithm which converts the "*Description of the power system*" (*.csv* files) into "*Power system data*" (AMPL files) and notably schedules the maintenance of the thermal units. The TS generator, which builds the stochastic inputs of a given Monte Carlo year, is presented in section 5. Finally, section 6 describes the prototype of the optimisation problems which simulates the behaviour of the power system and creates the output folder.

3. Fill the time series model database

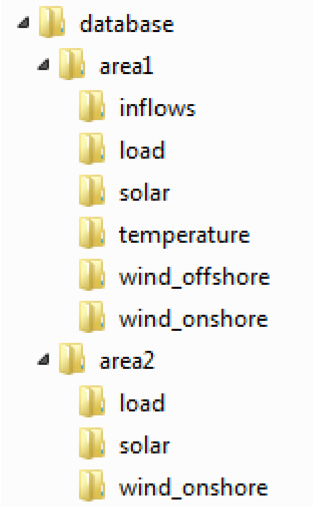
The TS generator requires, as input, intrinsic characteristics of each of the considered time series, and notably:

- Their seasonality: deterministic patterns which describe the average evolution of the time series along the year
- Their stochastic properties: description of their stochastic part with the coefficients of their associated model (either an ARMA-type model or a diffusion-type model)
- The spatial correlations which link them: cross-correlations between time series taken in different places.

Those intrinsic characteristics are learnt from historical data with a module called TS analyser. This module has been prototyped with Matlab and is described in this section.

3.1. Architecture of the time series models database

The hierarchy of the database is illustrated in Figure 2. Information related to a given *area* and *type* of stochastic phenomenon is stored in the folder *database/area/type*.



- database
  - area1
    - inflows
    - load
    - solar
    - temperature
    - wind_offshore
    - wind_onshore
  - area2
    - load
    - solar
    - wind_onshore

Figure 2. Architecture of the model database

In order to build a new model, historical data must first be saved in the folder associated to the considered *area* and *type*. Historical data are saved in *.xlsx* format in a standardised table which will be later read by Matlab.

3.2. Construction of time series models

The construction of the time series model is not fully automatic. The intrinsic characteristics of each model are calibrated on a case-by-case basis. Several models are often tested, updated and improved, until one which properly fits the historical data is found.

A set of functions has been developed in Matlab in order to properly calibrate each model and verify its consistency. The functions which are typically used for this task are listed below.

TS(*zone, type, name*), returns the historical time series saved in the file *name.xlsx* of the given *area* and for the given *type* of stochastic phenomenon. Historical time series are denoted *ts* in the following of this document.

ModelTS(*zone, type, name*), loads (or builds it if the file does not exit) the time series model *name.mat* of the given *area* and of the given *type* of stochastic phenomenon. Time series models are denoted *model* in the following of this document.

model.setTandS(*ts*, *category_trend*, *category_seasonality*, *fourier_order*), calibrates the trend and the seasonality of the *model* based on the historical data *ts.* Different categories of trend and seasonality can be tested. The seasonality is based on Fourier series with dummy variables which distinguish different types of day or hours the day. The order of the Fourier series can be tuned as well. Trend and seasonality are calculated with non-linear regressions.

Examples of categories of trend: *constant*, *linear*, *quadratic*, *spline1year* (regression on spline functions with one knot per year), *spline2years*, etc.

Examples of categories of seasonality: *CosSinHour* (24 Fourier series, one for each hour of the day), *CosSinHourFridaySaturdaySunday* (96 Fourier series, one for each for each hour of the day, Fridays Saturdays and Sundays are distinguished and have their own Fourier series), etc.

An example of regression of historical data on trend and seasonality is given in Figure 3.

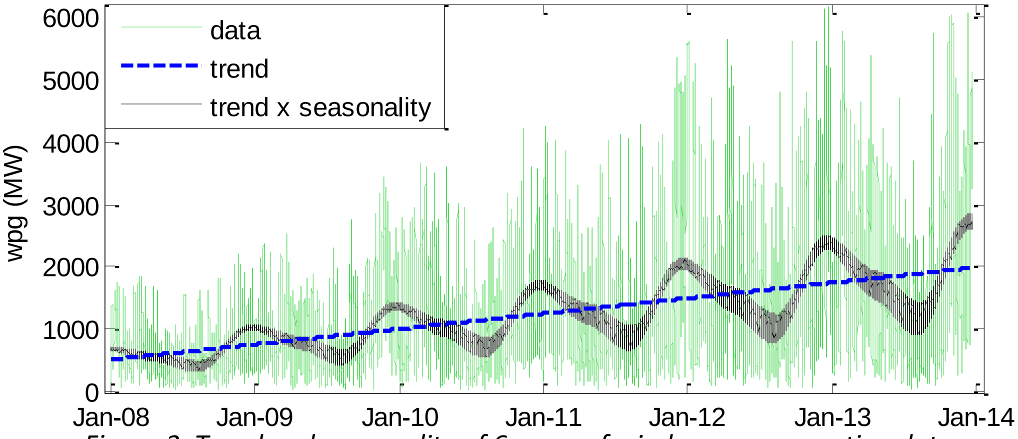


Figure 3. Trend and seasonality of 6 years of wind power generation data.

model.setTandSwithT(*ts*, *category_trend*, *category_seasonality*, *fourier_order*, *temperature_origin*) is used in the case of time series with a thermo-sensitive component. The dependence of the historical data *ts* to the temperature of the zone *temperature_origin* is computed as well.

model.setarima(order, p, d, q, normalized), calibrates an ARIMA(p,d,q) on the residuals of the considered *model.* The last argument allows normalizing first the residuals by dividing them by their monthly standard deviation.

model.setprocdiff(type_of_pdf, normalized), calibrates a diffusion-type model on the residuals of the considered *model*. Different types of probability density function (pdf) can be handled, notably Gaussian, Beta and Gamma distribution functions. The last argument allows normalizing first the residuals by dividing them by their monthly standard deviation.

model.infos, returns general information on the current model.

model.plot, plots diverse charts which compare historical data with the present model. Figure 3 is an example of a graph which can be obtained with this function.

model.getnewMC(new_trend), generates one new Monte Carlo year with the current model and the new trend given in argument.

model.check(n), compares *n* new Monte Carlo years with the historical data available for the *area* and *type* of the considered model. Several characteristics are compared, notably: the yearly averages, the standard deviations between yearly averages, the monthly averages, the pdf and tails of the pdf (5th and 95th percentiles) and the autocorrelation functions.

model.save, saves the model in the folder *database/area/type*.

These functions can be tested with several arguments so as to find a model which better fit the specificities of the processed data.

In order to be further processed by the TS generator, an element of the class model must include a seasonality (fixed pattern over one year learnt with the function *setTandS*) and a stochastic model (either an ARMA-type model learnt with function *setsarima* or a diffusion-type model with function *setprocdiff).* The thermo-sensitive component is facultative.

3.3. Purpose of the folder "general inputs"

The folder *general_inputs* (presented in section 2) contains the list of models which are used during the study, for each area of the studied system and each type of stochastic phenomenon.

Note that the area of the studied system does not necessarily match the historical area on which the models have been learnt. For example, if no historical data are available for a given area and type of stochastic phenomenon (typically: new offshore generation site), the model of a neighbouring area can be used instead.

Correlation matrices are also included in this folder in *.csv* format. They are filled with the correlation coefficient calculated between the residuals of the used models.

4. Build a scenario

The construction of the scenario is performed once for every scenario and time horizon. It:

- Converts the inputs of the task into a new format which will be further read by TS generator and by the adequacy simulator (*.dat*).
- Computes the *expected seasonality* of the stochastic inputs of the problem. The expected seasonality is notably used to schedule the maintenance and allocate the hydrological resources.
- Plans the maintenance of the thermal generators.

These three actions are handled by a Matlab function called *build_scenario*. The inputs/outputs of this function are given in Figure 4. The path of the input and output folders are reported in the file *setup.csv*. *setup.csv* is placed in the current folder (same folder as the function file).

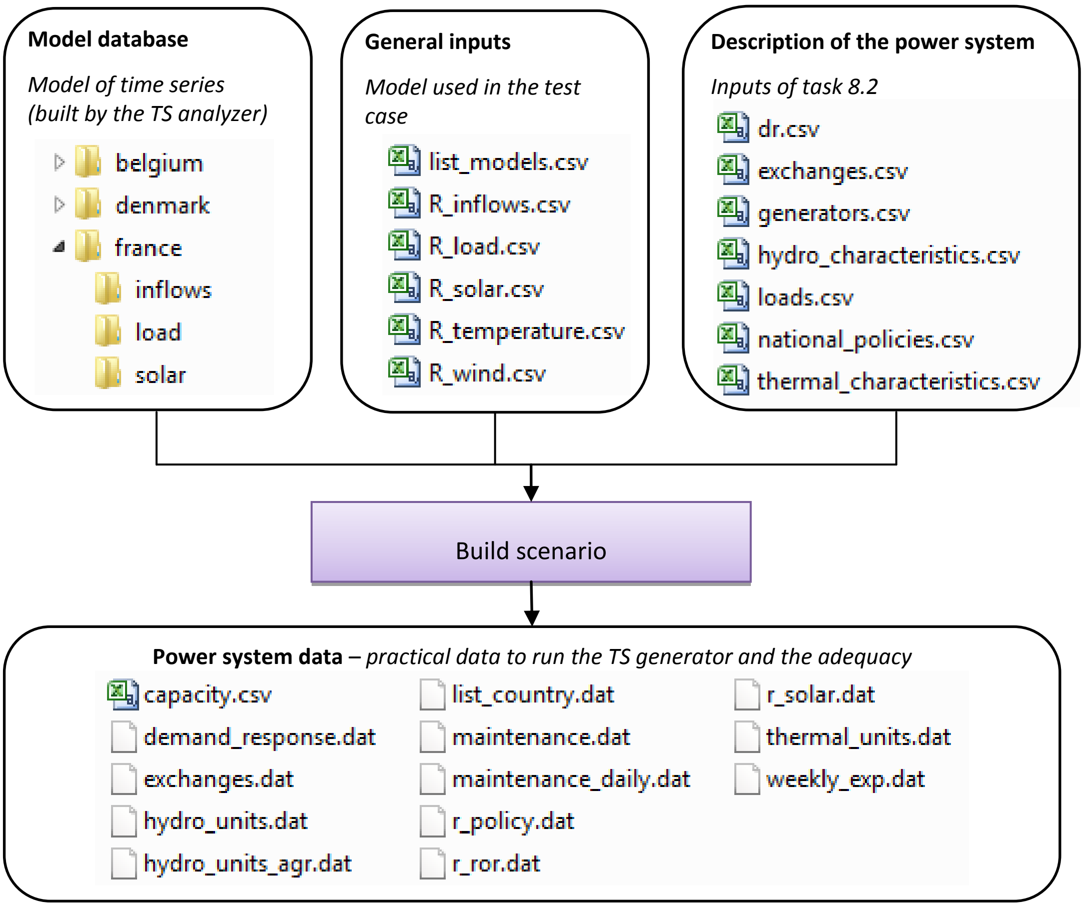


Figure 4. Inputs/output of the scenario builder

The scenario and time horizon is described with the following files:

- *dr.csv*: DR programs of each country (participation rate, delay time, costs)
- *exchanges.csv*: extra-European exchanges possibilities (costs, exchange capacity)
- *generators.csv*: installed power plants of each country (Pmin, Pmax, energy source)
- *hydro_characteristics.csv*: hydrological resources of each country and sizes of the reservoirs
- *load.csv*: demand of each country
- *national_policies.csv*: objective of each country in terms of self-production.
- *thermal_characteristics.csv*: standard characteristics of each type of thermal units (minimum up and down times, maintenance requirements, outage risks, generation costs)

These files are loaded by the function *build_scenario*, some data are re-organised and then converted in a format which can be read by AMPL (*.dat*).

For each country, the expectation of the stochastic phenomena (demand, renewable energy sources (RES) generation and inflows) is computed, based on the time series models previously built and on the inputs of the scenario (yearly demand, installed RES capacities, yearly hydro resources). Expectations of the time series of the stochastic phenomena are written in *weekly_exp.dat*.

The method used to generate the maintenance schedule is a simple heuristic algorithm whose goal is to program the stops of the units when the expectation of the residual load is low. The maintenance program is written in the files *maintenance.dat* and *maintenance_daily.dat*.

5. Sample a new Monte Carlo year

For each scenario and time horizon, several Monte Carlo years are studied. A Monte Carlo year is described by a set of time series of the uncertain inputs of the problem. The uncertain phenomena considered are: the RES generation (solar, onshore wind and offshore wind), the demand, the hydrological inflows and the outages of the thermal units.

The generation of time series is handled by a Matlab function, called *get_new_mc.* The inputs/outputs of this function are given in Figure 5. The paths of the input and output folders are reported in the file *setup.csv*, as well as the seed used to initialise the pseudo-random numbers generator. The seed is integer specific to each Monte Carlo year.Figure 4

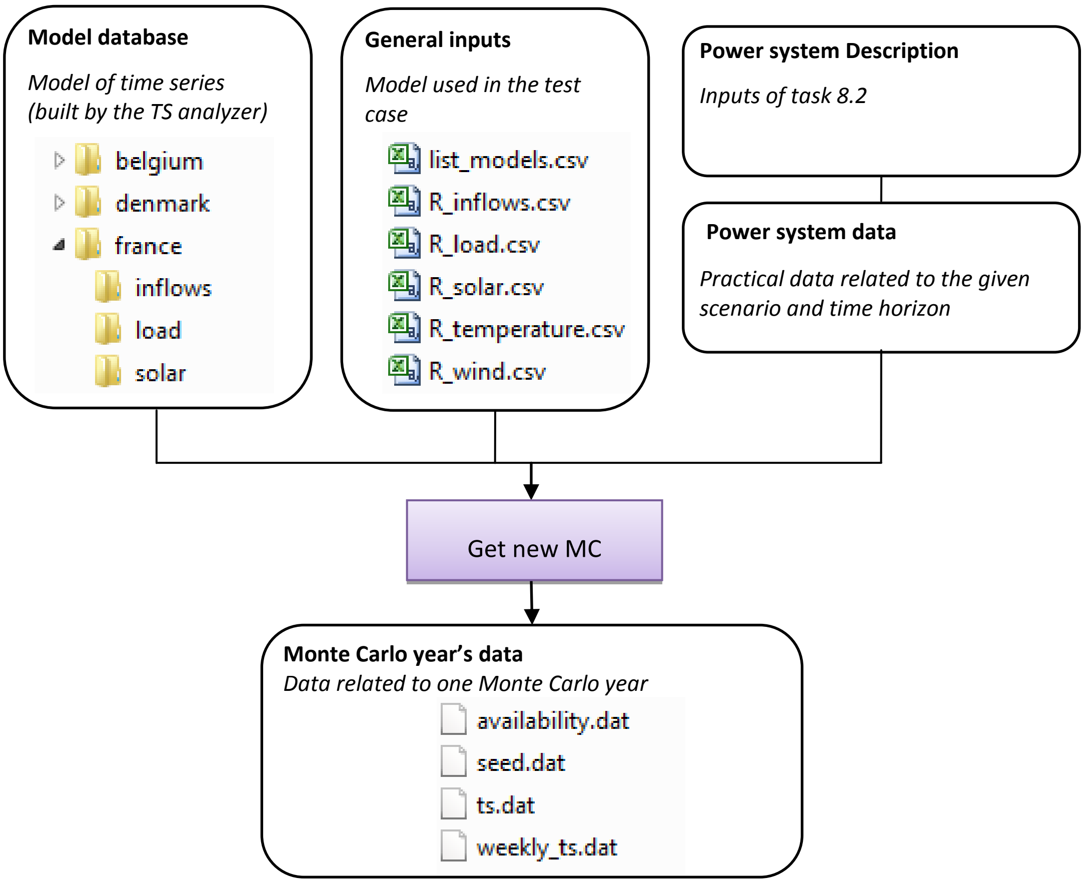


Figure 5. Inputs/output of the sampling of a new Monte Carlo year

The function *get_new_mc* generates:

- For each thermal unit, time series of outages, saved in the file *availability.dat*
- For each area of the studied power system, time series of RES generation, demand and inflows, saved in the files *ts.dat* (with an hourly time step) and *weekly_ts.dat* (with a weekly time step).

The generation of the time series is based on the model referenced in *list_model.csv* and stored in the database. Spatial correlations can be taken into account. Correlation matrices must be placed in the general inputs folder.

6. Simulate the operation of the power system

For each Monte Carlo year, the operation of the power system is then simulated so as to determine the time series of the controllable elements (thermal and hydro power plants, exchanges and controllable demand).

The problem is first decomposed into 52 weekly sub-problems. The adequacy is then solved for each week with an hourly time step.

6.1. Decompose the problem

The decomposition of the problem is performed by the AMPL script *hydro_scheduling.run.* The hydro scheduling problem consists in allocating the hydrological resources among the 52 weeks of the year. The inputs/outputs of this function are given in Figure 6.

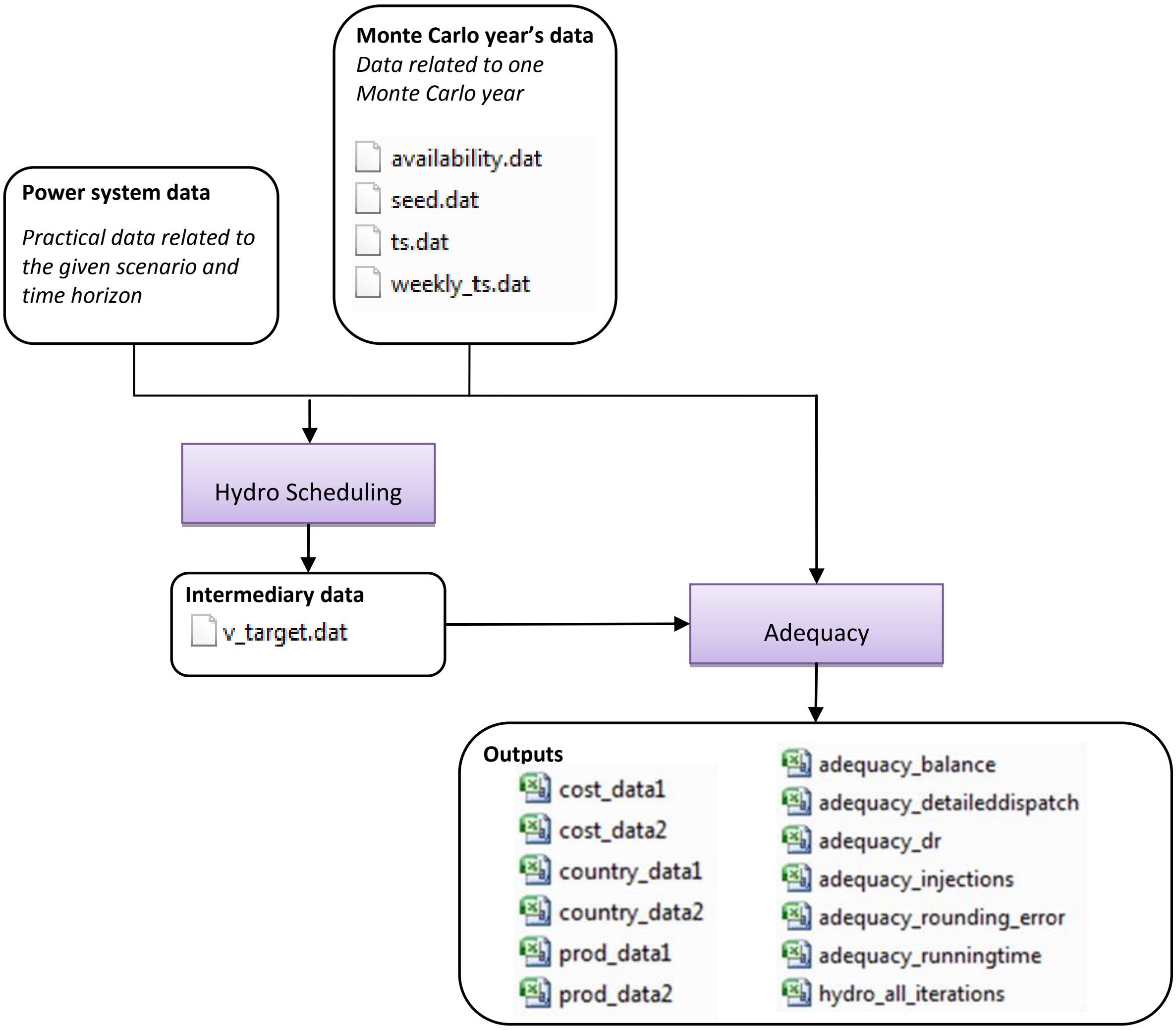


Figure 6. Inputs/outputs of the adequacy problems

The hydro scheduling optimisation problem defines the weekly volume targets for each of the reservoir of the system. They are saved in the file *v_target.dat*.

This step consists of a rough adequacy problem with a weekly time step. One representative hour, with average values of generation and consumption, is used to represent each week. Hydro and thermal resources are scheduled so as to minimize the expected generation costs. Moreover, the allocation of the hydro energy is made through an iterative method with a 1-week foresight rolling planning.

6.2. Solve the hourly adequacy problem

Finally, detailed adequacy simulations, with an hourly time step and a finer modelling of the elements of the system are performed.

The adequacy simulations are performed by the AMPL script *adequacy.run.* The inputs/outputs of this function are given in Figure 6.

52 optimisation problems of 168 time steps are run iteratively. The commitment of the thermal units which are out-of-order or in maintenance is forced to zero. The volume targets obtained in the previous step are used as boundary conditions of the hydro reservoirs at the beginning and end of each week.

The adequacy problem is a large mixed-integer linear program (MILP). It includes a detailed representation of the thermal units, with unit commitment, minimum stable power and minimum on-line and off-line durations. The generation cost includes three components, a linear cost (proportional to the output power of the unit), a constant cost (which takes effect only when the unit is online) and a start-up cost. A linear model is used for hydro and storage. Hydro resources are considered free, but they are limited by the bounds on the reservoirs determined in the previous step. Load can be controlled via Demand Response (DR) programs, which allow to shift a part of the load within a given delay time. Extra-European exchanges are possible and are bounded by the transmission capacities with the neighbouring countries of Europe.

The following outputs are written for each week of the considered Monte Carlo year:

- *prod_data.csv*, contains the output power and the availability of each generator of the system, for each hour of the week
- *country_data.csv*, contains the demand and the aggregated RES of each country of the system, for each hour of the week
- *cost_data.csv*, contains the system cost for each hour of the week

In addition, several output files with aggregated values or reliability indicators are written in order to analyse more easily the results.

7. Conclusion

The four modules of the methodology of task 8.2 have been implemented using AMPL and Matlab. An organised folder hierarchy has been built in order to manipulate the large volume of data of the different scenarios, time horizons and Monte Carlo years. Moreover, scripts have been written in order to automate the simulations of several Monte Carlo years.

The whole adequacy simulator may be run in parallel; tests were made on a server with 16 CPUs and up to 16 yearly Monte-Carlo simulations were successfully run in parallel.