renpass
Renewable Energy Pathways Simulation System
- Manual -

Frauke Wiese

November 16, 2014

Contents

List of Figures

1 Installation of renpass

1.1 License

renpass (Renewable Energy Pathways Simulation System) is a simulation energy model. renpass is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, version 3 of the License, or any later version. renpass is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose.

Licenses for open source software are designed to guarantee the freedom to share and change all versions of a program and to make sure it remains free software for all its users. The GNU General Public License is a free, copyleft license for software and other kinds of works. The latest one is version 3 which was agreed on by the Free Software Foundation on 29th of June 2007 (FSF, 2007). According to this license, everybody can use renpass, adjust it to their needs, and has to use the same license for distributing this variation. The license itself is important to keep renpass open. The reason why the authors have chosen GNU GPL3 is that it fits to the objectives of the development, is widely spread and will be used further by the open source community. Furthermore this license is compatible with the software used. More details on the terms of and the complete GNU GPL 3 license can be fund on `http://opensource.org/licenses/GPL-3.0`.

1.2 Software Requirements

The following software is required for using renpass:

- MySQL - database
- R - programming language
- RMySQL - package for the connection between the MySQL database and the R program

The installation description for this software can be found in the manual for MySQL and R installation (Wiese et al., 2013). In the database, scenario settings, data and results of renpass are stored. The R code consists of several code subroutines which are R script files. For applying renpass, you need four databases, distributed as sql-files and the folder renpass containing the code.

All the files have to be saved on your computer. This is further described in the following.

1.3 Database Import

Import the following MySQL databases, which are distributed as .sql files:

- `pathways.sql`
- `weather.sql`
- `renpass.sql`

- results.sql

You can either import these by phpMyAdmin or use the terminal. For standard setting of phpMyAdmin import, the database files are probably too large. You can either change the settings for phpMyAdmin or simply use the terminal. In windows the terminal can be found in the windows menu / start -> ausführen. Type cmd. If you do not have the entry ausführen, go to Zubehör, Eingabeaufforderung.

In the terminal, you have to set the directory to the folder on your computer where the sql-files for import are located (cd). Then import the file by writing the username, password, database and file name. In the following example, the folder is called Documents/renpass/database_files (This will be different on your computer), the username for the database is root and the password is user123. The databases are called pathways, weather, renpass, results and the particular files are sql-files, thus they have the file extension .sql. You can get out of the folder by just typing cd and then enter.

Type the following and push "enter" after each line.

For **Linux**:

```
> cd Documents/renpass/database_files
> mysql -uroot -puser123 < pathways.sql;
> mysql -uroot -puser123 < weather.sql;
> mysql -uroot -puser123 < renpass.sql;
> mysql -uroot -puser123 < results.sql;
```

For **Windows**:

```
> cd C:\Documents\renpass\database_files
> C:\xampp\mysql\bin\mysql -uroot -puser123 < pathways.sql;
> C:\xampp\mysql\bin\mysql -uroot -puser123 < weather.sql;
> C:\xampp\mysql\bin\mysql -uroot -puser123 < renpass.sql;
> C:\xampp\mysql\bin\mysql -uroot -puser123 < results.sql;
```

For **Mac**:

```
> cd /users/renpass/database_files
> /Applications/MAMP/Library/bin/mysql -u root -p
> "type root-passwort"
mysql> SOURCE pathways.sql;
mysql> SOURCE weather.sql;
mysql> SOURCE renpass.sql;
mysql> SOURCE results.sql;
```

```
> cd
```

Check if all tables are imported:

- pathways : 17 tables
- weather : 7 tables

- `renpass` : 16 tables
- `results` : 14 tables

1.4 renpass Code

The folder `renpass` contains the following sub-folders

- `code_R_renpass` contains all code pieces and R functions for the simulation and plotting.
- `log` is empty. The log-files containing scenario assumptions and running time of the simulations will be automatically generated during simulations and saved in this folder.
- `plot` contains the folders `area_and_region_pngs`, `geodata` and `plot_results`. The first one contains png-files that show the regions of the ten test scenarios, the second one contains shape-files required for spatial result plots. The third one is empty. The result plots will be saved there if the automatic plotting option is chosen during the start of the simulation.

You can save the renpass folder wherever you want on your computer, but you have to tell renpass the correct path to its home folder: Open the file `code_R_start.R`. Write down the path to the folder in which you saved renpass after
`path <-`
and specify if you work with Linux, windows or mac after
`system_software <-`
If you work with a mac, you additionally have to check if
`database_port <- 8888` is the right setting for your mac. You can find the right port for your mac on the start page of MAMP.

Example for **Linux**:

```
path            <- "/home/frauke/Documents"
system_software <- "linux"
```

Example for **Windows**:

```
path            <- "C:/Users/EUM/Energiesystem"
system_software <- "windows"
```

Example for **Mac**:

```
path            <- "/Users/sebastian/Documents"
system_software <- "mac"

# special stuff for macs
if(system_software == "mac"){
  database_host        <- "localhost"
  database_port        <- 8888 # could also be 8889, you have to check
  database_unix.socket <- "/Applications/MAMP/tmp/mysql/mysql.sock"
}
```

2 Choose Pathways

The number of parameters that have to be set before simulating an energy system is large. renpass allows the simulation of different scenarios that combine different pathways in the areas demand, renewable expansion, weather conditions, thermal power plants, storage development, grid development and resource prices. In energy system models there is a trade-off between the flexibility of input parameters and applications on the one hand and the complexity of the code and applicability for users on the other hand. If too many input parameters have to be set for calculating a pathway, the number of people able to use the model shrinks. In renpass, this challenge is dealt with by offering different levels of detail for defining input parameters. For each pathway table either parameters can be defined in detail for a model run or standard scenarios can be used if the respective scenario area is of minor importance for the focus of the planned calculation. Nevertheless, you have to be aware that also the predefined scenarios do influence the simulation results.

In any case, the scenario year and the weather year, the time span, and the length of time steps have to be chosen. The alternatives are hourly or quarterly calculation. With those options renpass is highly flexible with regard to input parameters. Nevertheless, it has to be stated that the determination of standard scenarios contains a lot of decisions that influence the simulation results. Thus you have to be aware what influence this presetting can have on your calculation results. For example, even if you are just interested in the performance of biomass plants in two different biomass scenarios, the grid infrastructure and the capacity and distribution of wind energy plants do have influence on the results.

The renpass input parameter setting is organized in the database `pathways`. Predefined scenarios or combinations of development paths can be chosen. Scenarios and sub-scenarios are identified by their `scenario_name` (= *primary key*). Additionally, each of the sub-scenarios and its parameters can be defined by the user. Figure 1 gives an overview of all scenario parameters, sub-scenarios and their attributes.

2.1 Main table: 1_scenario_parameter

The starting point for setting a new scenario is table `1_scenario_parameter`. The primary key of this table is the `scenario_nr`. Thus this number has to be unique and by this number the results and figures can be identified and connected to the input parameters during and after the scenario run. Here you can either choose from the already defined scenarios one that fits your demands and look up the `scenario_nr` or you can generate a new set of scenario parameters with a new unique `scenario_nr`. Click on `Insert`, then you can add a new line in this table. Figure 2 gives an impression of manually inserting a new line in this table. Each new line in this table with an unique `scenario_nr` defines one scenario setting. It includes many choices. Some choices are simplified by having standard settings or drop-down menus. Standard choices can provide a solid basis for the first calculation of a new reference scenario, but can also be adapted. Ten test scenarios with the scenario numbers 1-10 are already defined in the table `1_scenario_parameter`.

Instead of inserting each new scenario manually via the graphical user interface phpMyAdmin, there are different ways of importing new scenario settings as csv or sql. Scenarios can also be generated and pre-processed in R and written into the database via RMySQL-connection.

The following parameters have to be chosen in for defining a new scenario in the table

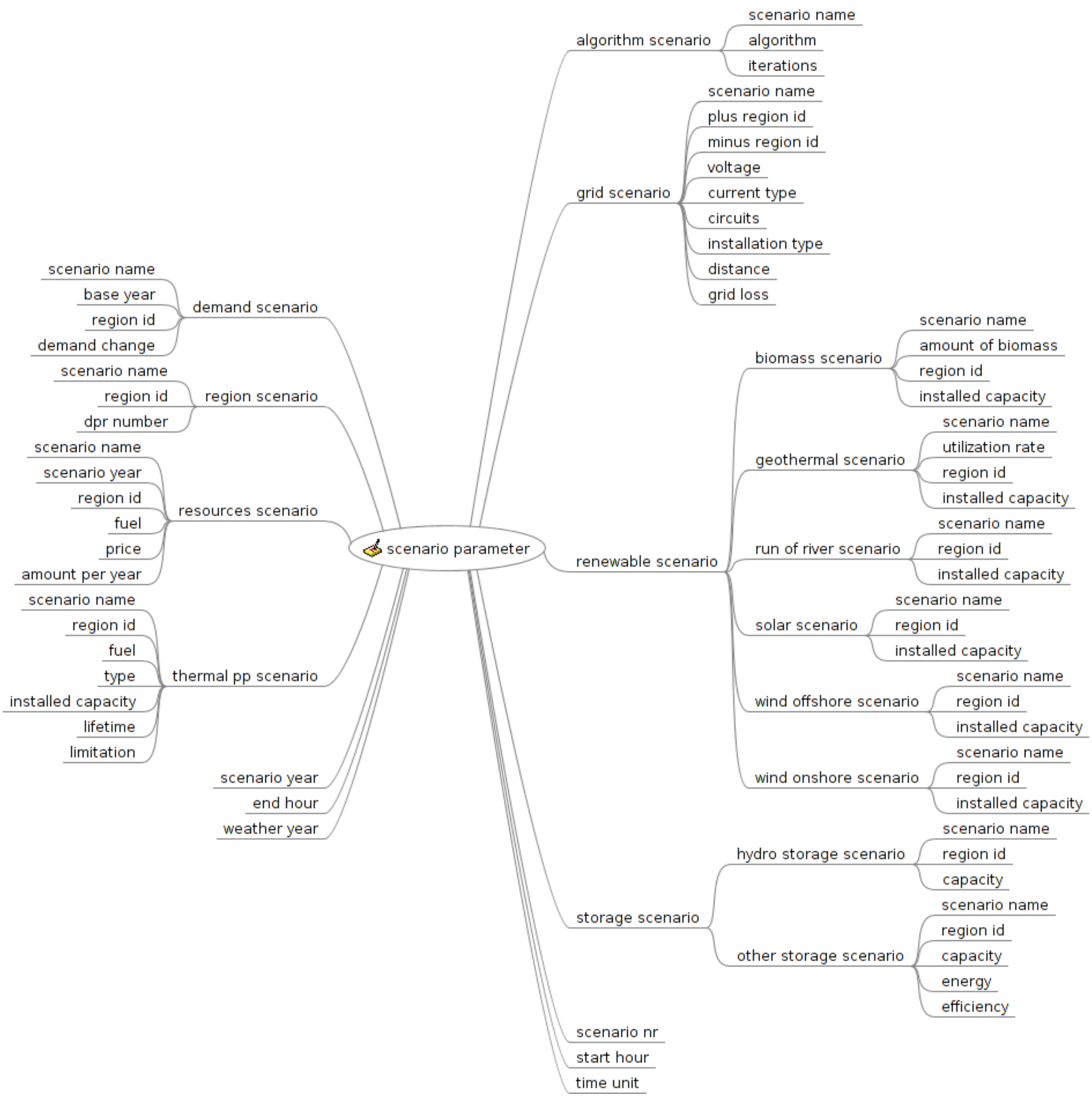


Figure 1: Sub-scenarios and attributes of all pathways tables

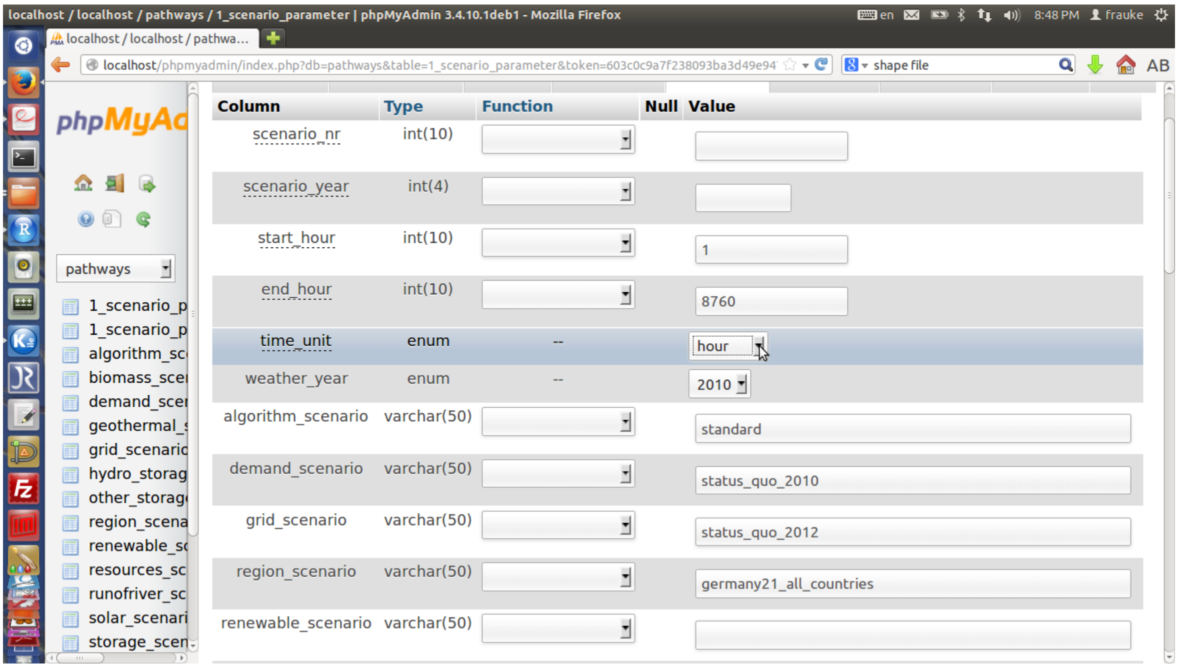


Figure 2: Screenshot of choosing new scenario settings manually

1_scenario_parameter:

- `scenario_nr`
  unique number for the identification of the scenario results

- `scenario_year`
  one year in the range of 2011 to 2050

- `start_hour`
  number in the range of 1-8759, hour of the year in which the simulation starts

- `end_hour`
  number in the range of 2-8760, hour of the year in which the simulation ends

- `time_unit`
  `hour` or `quarter`, defines if the simulation calculates values for each quarter or for each hour of the time span

- `weather_year`
  1998, 2003 or 2010 can be chosen. This defines from which year the historical weather data for wind speed, solar radiation and water levels is used for wind, solar and run-of-river electricity feed-in. Those years differ significantly in their weather pattern:
  1998: high wind / medium solar radiation / high hydro inflow
  2003: medium wind / high solar radiation / medium hydro inflow
  2010: low wind / high solar radiation / low hydro inflow

The parameters for

- `algorithm_scenario`

- demand_scenario
- grid_scenario
- region_scenario
- renewable_scenario
- resources_scenario
- storage_scenario
- thermal_pp-scenario

are clustered in subtables. Only the `scenario_name` has to be specified in the table `1_scenario_parameter`. This name identifies the specifications in the respective scenario sub-table. The pathways database contains one table for each sub-scenario, in which the parameters are defined in detail. Figure 1 gives an overview of all scenario parameters, sub-scenarios and their attributes.

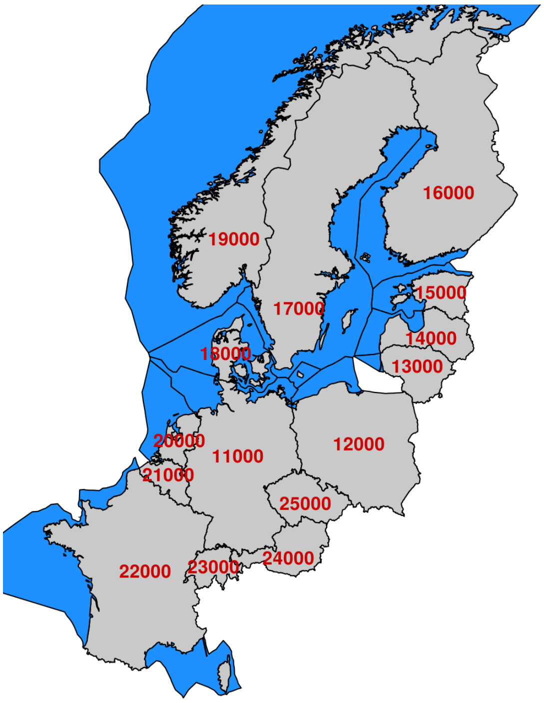
2.2 region_scenario

The region scenario setting is of key importance for defining the other sub-scenarios. Since most of the pathway parameters can be specified by region, the regional resolution has to be defined first. The `region_scenario` determines the simulation area and how this area is divided into regions. The choice of area and the region breakdown is crucial for the definition of sub-scenarios on installed capacities and other parameters as the required resolution of these settings differs according to the `region_scenario`.

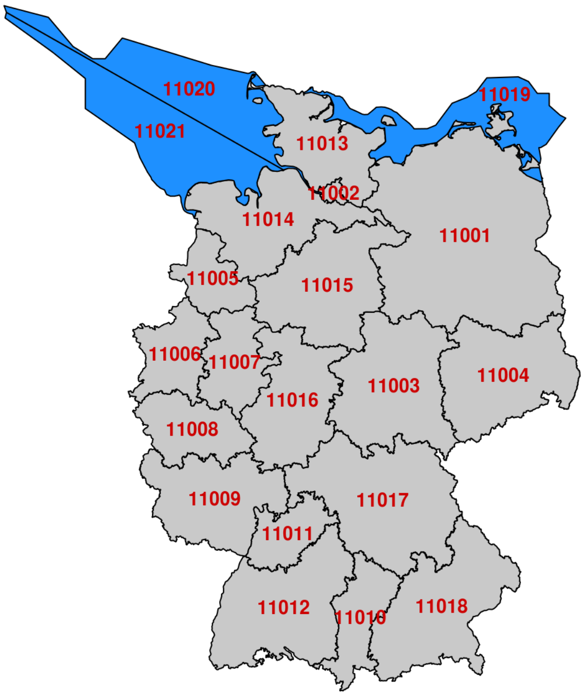
For example, if no region division within Germany is defined, the whole country is modeled as one region without internal grid constraints. In this case, installed capacities and other input parameters can be provided as nationwide aggregations. It would be enough to determine the sum of installed wind capacity in Germany as, for instance, 40 GW. If Germany is modeled in 21 regions, all settings have to be provided for all regions. In this case, the user has to decide upon the geographic distribution of the 40 GW across Germany.

The map in Figure 3a shows all countries for which data is available in the renpass database. Thus, this area can be covered in renpass. In blue, the corresponding offshore regions of the countries (Exclusive Economic Zones [EEZ]) are illustrated. In red, the region IDs utilized in renpass are indicated. Figure 3b shows the renpass sub-regions on- and offshore for Germany.

The regional resolution determines which grid bottlenecks can be discovered. Thus, grid bottlenecks within the country can just be looked at for Germany since the data for this country is stored in the renpass database in a higher regional resolution. Germany is divided in 21 regions: three offshore and 18 onshore regions. The onshore regions are derived from a map the Transmission System Operators (TSO) in Germany published (German TSOs, 2013). The TSO proposed those regions for the analysis of grid bottlenecks and flows between and within the grid control areas. Thus the regions are chosen in a way that bottlenecks appear on the borders. Those regions are also used in the study about the grid infrastructure that is necessary for wind energy by the German Energy Agency (dena, 2011), but since there was no exact geographical data published, the map in (German TSOs, 2013) was the basis for the derivation of the renpass regions. Due to reasons of data availability, the regions for renpass



(a) renpass region IDs of country regions



(b) renpass region IDs of German subregions

Figure 3: renpass country regions (a) and Germans subregions (b) including their region IDs. Sources: GADM (2012),VLIZ (2012),OpenGeoDB (2013), own editing.

are the merged areas of districts. For districts and thus also for the regions, information like inhabitants and GDP is available.

The region id for Germany is 11000. The region ids of the sub-regions start with 11001. Offshore, the Baltic Sea is one region (11019) and the North Sea is divided in region 11020 and 11021. 11020 is the Northeastern and 11021 the Southwestern part of the German EEZ. According to the offshore plan for the North Sea (BSH, 2013) this clustering of the North Sea could fit to the cluster of the onshore grid connection: 11020 to Schleswig-Holstein and 11021 to Lower Saxony coast. In the simulation grid connection to other countries and regions can be added.

The area in the basic version of renpass covers Poland, Lithuania, Latvia, Estonia, Finland, Sweden, Denmark, Norway, the Netherlands, Belgium, Luxembourg, France, Switzerland, Austria, the Czech Republic and Germany. Germany can be divided into up to 21 regions, all other countries constitute one region each.
Some region scenarios are predefined and already included in the table `region_scenario` in the pathways database. Figure 4 shows examples for possible region clusters. Region scenarios already introduced into the pathways database area:

- all_countries_no_split
- germany21
- germany21_all_countries
- germany21_neighbours
- germany21_norway
- germany21_NO_AT_CH
- germany5
- germany5_all_countries
- germany5_neighbours
- germany5_norway
- germanyTSO_neighbours
- germany2
- germany_neighbours_no_split

For those region scenarios with five regions in Germany, the 21 sub-regions of Germany are merged as follows.

- North Sea and North-West Germany: 11002, 11005, 11013, 11014, 11015, 11020, 11021
- Baltic Sea and North-East Germany: 11001, 11003, 11004, 11019
- Rhine and Ruhr Area: 11006, 11007, 11008, 11016
- Saarland and Upper Rhine Area: 11009, 11010, 11011, 11012

- Bavaria without Southwest-Bavaria: 11017,11018

The region scenarios utilized in the ten test scenarios are illustrated as png-files in the folder `plot/area_and_region_pngs`.

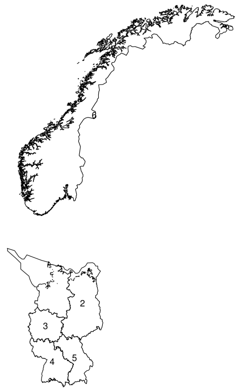
For new region scenarios, for each newly defined modeling region, a dispatch region number (dpr) has to be given. This dispatch number is then assigned to each region that shall belong to the dispatch region. The dispatch regions numbers are
`dpr = 1...n`
with
`n` = amount of regions in this scenario.
If two regions get the same dpr number (dispatch region number) that means that they are aggregated in one region for the simulation. For Germany, all regions from 11001 to 11021 have to be put in the region scenario and given a dpr number if Germany is included in the simulations. If the whole of Germany is taken as one dispatch region for calculation the same dpr number is assigned to every region. For example `germany5_norway` looks like in Table 1 for the `region_scenario` table:

scenario_name	region_id	dpr_number
germany5_norway	11001	2
germany5_norway	11002	1
germany5_norway	11003	2
germany5_norway	11004	2
germany5_norway	11005	1
germany5_norway	11006	3
germany5_norway	11007	3
germany5_norway	11008	3
germany5_norway	11009	4
germany5_norway	11010	4
germany5_norway	11011	4
germany5_norway	11012	4
germany5_norway	11013	1
germany5_norway	11014	1
germany5_norway	11015	1
germany5_norway	11016	3
germany5_norway	11017	5
germany5_norway	11018	5
germany5_norway	11019	2
germany5_norway	11020	1
germany5_norway	11021	1
germany5_norway	19000	6

Table 1: Example for region scenario definition germany5_norway+: Five regions in Germany and Norway as one region

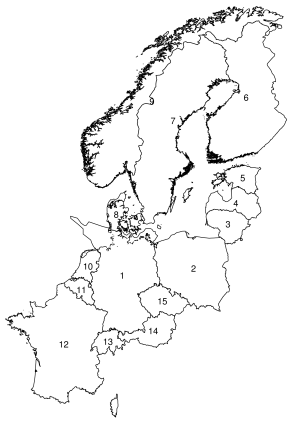
Further information on the regions including their ids, official country codes can be found in the table `region_parameter` in the `renpass` database.

(a) germany2

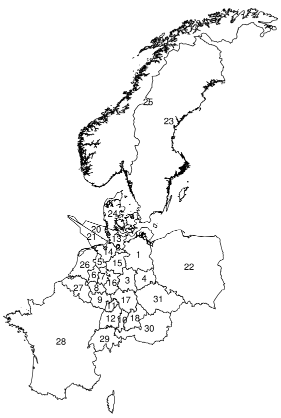


(b) germany5_norway

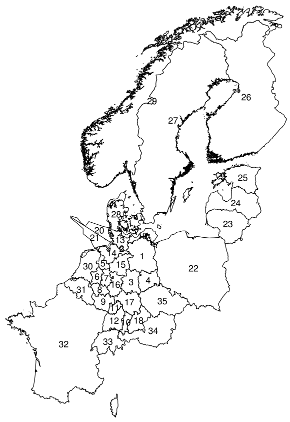
(c) germany5



(d) all_countries_no_split



(e) germany21_neighbors



(f) germany21_all_countries

Figure 4: Predefined region scenarios. Different country compilations and Germany clustered in 2, 5 or 21 regions. Source: Own illustration, based on shape files from GADM (2012)

2.3 algorithm_scenario

This table defines the algorithm that shall be used for the exchange between the regions. Since the simulated annealing function is not activated and does not work yet, the algorithm `standard` should be chosen. `iterations` specifies he number of iteration steps for the exchange between regions. One iteration step means one exchange between two randomly chosen regions. The default value is 3000. This empirical value serves quite well, it produces robust results, without taking too much running time.

2.4 demand_scenario

The demand scenario defines the structure and quantity of the demand per region. Additionally there is a column defining how much of the demand should be supplied by the region itself, but this is not implemented in the code yet.

- `scenario_name` has to be unique and identifies one set of demand parameters from the `1_scenario_parameter` table.
- `base_year` defines which time series for demand is taken. At the moment in the database renpass the demand values of 2010 and 2011 are stored in the `demand_timeseries` table. Thus 2010 or 2011 can be chosen.
- `scenario_year`
- `region_id`: for each region id in the simulation, all the parameters have to be set. The regions are defined by the `region_scenario`.
- `demand_change` is the factor with which the demand values of each hour are multiplied. If it is one, the demand is the same like in the base year.
- `min_self_supply` [MW] indicates how many MW should at least be produced in a region itself for supplying the demand in the respective region, it will not have any effect if set to 0. Anyway, this is not implemented in the code yet.

2.5 grid_scenario

defines all connections between regions with their capacity and other properties. The value of the maximum transfer capacity can be either specfied by th capacity value or by the number of circuits and voltage level. Capacity is the dominant value. If no capacity value is specied the transfer capacity is calculated from the values for number of circuits and voltage. Thus, either `circuits` or `capacity` has to be filled.

Grid loss of each connetion can be specified by defining the length of a line in km in the column `distance` or a %-value for the losses of a connetion has to be specified in the column `grid_loss`. Table 2 gives an overview of the columns of the grid pathway table.

Grid infrastructure is part of the scenario assumptions. Nevertheless, information about the status quo of grid infrastructure is crucial. Since investment cycles of extra-high voltage lines exceed forty years, grid scenarios will be based on today's grid infrastructure. Thus there are predefined scenarios in the pathways table `grid_scenario` for the `status_quo_2012` and for

column name	unit/type	possible values
scenario_name	character	a unique scenario name
plus_region_id	numeric	region the energy comes from if the flow is positive
minus_region_id	numeric	region the energy comes from if the flow is negative
voltage	[kV]	usually 220 or 380
kind_of_current	character	“ac” or ”dc“
circuits	numeric	either circuits or capacity have to be filled
capacity	[MW]	either circuits or capacity have to be filled
kind_of_laying		”land“ or ”sea“
distance	[km]	either distance or grid_loss has to be filled.
grid_loss	numeric	0.03 is the standard value and means 3% grid_loss per connection between two regions

Table 2: Columns and values of the grid pathway table

`planned_2015`. For each of those it can be chosen if the standard grid loss value of 0.03% per connection should be applied or if the grid loss per connection should be calculated based on the distance between the center points of the connected regions. The latter is defined by the suffix `_distance` in the scenario name.

For the `status_quo_2012` grid scenario, information on number of circuits, voltage level [kV] and current type [AC or DC] of extra-high voltage lines (>=220kV) were based on the ENTSO-E grid map (ENTSO-E, 2012b). NTC values are utilized for inter-country connections in renpass. The available capacity between countries is published by ENTSO-E as Net Transfer Capacity (NTC) (ENTSO-E, 2011). If the value differs depending on the direction of the flow direction, the higher value is employed for both directions. NTC values constitute the maximum foreseen magnitudes of exchange programs that can be operated between two areas respecting the n-1 security conditions of the involved areas, taking into account the uncertainties on the assumptions of NTC assessment (ENTSO-E, 2011). These values are provided in MW.

None of the predefined scenarios includes connections to the German offshore regions 11019, 11020 and 11021.Thus, if a `germany21` region scenario is utilized and there is installed offshore capacity, the connections have to be defined, otherwise all electricity generated offshore would be excess electricity.

2.6 renewable_scenario

This is an overview table for the different pathway definitions of the different renewable technologies which are:

- `wind_onshore_scenario`
- `wind_offshore_scenario`
- `solar_sceario`
- `runofriver_scenario`

- geothermal_scenario
- biomass_scenario

Each of those is one table consisting of the columns

- scenario_name
- region_id
- installed_capacity [MW]

The installed capacity has to be given in MW per region and includes the capacity that is already installed today. The biomass scenario table has an additional column, since the amount of this resource, which is made available for electricity generation is another decision that has to be taken by the simulator. Thus the amount_of_biomass has to be given in GWh per region and indicates the amount of biomass available for the whole year in this region. The geothermal_scenario has the additional column utilisation_factor. 0.9 is the standard value, which means 90% availability of geothermal power plants.

Information about already installed capacities and distribution of renewable energy plants is important for the creation of scenarios, since renewable scenarios are often not be completely independent of existing installations. Furthermore, information on today's distribution of installed capacities can be a useful indicator to extrapolate the distribution of installed capacities between regions for consistent future scenarios.

The status_quo_2012 renewable scenario, already inserted in this table is derived from ENTSO-E (2014) for all countries in renpass except Germany. Since this source has aggregated the installed wind energy capacity, the offshore share is taken from EWEA (2013, p.13).

In Germany, although the number of renewable plants is huge, the data availability is much better than for fossil power plants. The TSOs have the obligation to publish a register of all plants remunerated under the EEG mechanism. With the objective to provide an easier access to renewable plant data and to point to possible improvements in data provision, the German Section of the International Solar Energy Society (DGS) merges and verifies the registers of the TSOs and distributes the result as a csv-file of all renewable electricity plants except large hydro for free download on their website (DGS, 2013). Thus the German numbers of the status_quo_2012 renewable scenario in renpass are based on the downloaded register of February 2013.

2.7 resources_scenario

This scenario indicates the prices for the fossil resources hard coal, lignite, oil, gas and biomass in €/GJ, and the price for CO_2 certificates in €/t. Table 3 shows the columns of this table.

Three different resource scenarios are predefined in this table: high, moderate, low.

column name	unit/type	values
scenario_name	character	a unique name
scenario_year	numeric	one year in the range 2011 to 2050
region_id		5-digit number
fuel	character	CO_2, gas, hard_coal, lignite, oil, refuse, uran
price	[€/t], [€/GJ]	per ton for CO_2, per GJ for all other

Table 3: Columns and values of the resources scenario

2.8 storage_scenario

This is an overview table for the pathway specification of different storage technologies. In this version of renpass, these are:

- `hydro_storage_scenario`
- `other_storage_scenario`

Both scenarios are defined in sub-tables in which the installed capacity per region can be specified.

In the `other_storage_scenario` table additionally to the `capacity` [MW], the `energy` of the storage utility in MWh and the `efficiency` as part of 1 has to be specified. The number 1 in the column `efficiency` would mean 100% efficiency. These three columns can be defined per region.

For all countries except Norway (`region_id` 19000), the specified installed capacity in the `storage_scenarios` specifies the total installed pump capacity for the scenario including already existing installations. This number also defines the total installed turbine capacity. Thus it is assumed that the installed turbines in the scenario are all suited for pumping and generating electricity. For Norway, this is treated differently because a huge system of water turbines is already in operation in this country. This system is set for each simulation including Norway and the specified number of MW just indicates the additional pump capacity in this country.

2.9 thermal_pp_scenario

This table defines the fossil thermal power plants. There are two different possibilities to set the limit of fossil power plants: Either the `lifetime` for each technology and each fuel is given, then the installed capacity per region is calculated from the power plant register, or the `installed_capacity` in the year to be simulated can directly be specified in this table. This can be defined per fuel, per technology and per region.

If the limitation is `lifetime`, 0 has to be written into the column installed capacity column no matter which type or region, since the installed capacity results from the lifetime. A standard lifetime of all power plant types would be 40 years. If lifetime is the indicator, no power plants are added and the existing power plants from the power plant register fade out in accordance with their start-up year and lifetime. This has the advantage that very little has to be determined for the standard scenarios.

Thus the resolution of definition can be as detailed as region, fuel and type if necessary, but this has to be defined by the user. If a region is not mentioned in the `thermal_pp_scenario` but included in the simulation, it is assumed that there is no fossil power plant capacity at all. If the installed capacity or lifetime should not be defined further per type, then put 0 in the column `type`.

There are four predefined scenarios in the pathway table `thermal_pp_scenario`:

- `status_quo_2012`: All existing power plants in the year 2012. For German power plants based on BNetzA (2013), which includes types and parameters of existing power plants in Germany with an installed capacity of more than 10 MW installed capacity. For countries other than Germany, the values are derived from the Yearly Statistics & Adequacy Retrospect 2012 provided by ENTSO-E (2014). Since there is no information about the type of plants, lignite and hard coal fired power plants are assigned to the type steam turbine, oil fired power plants are assigned to gas turbine and the installed capacity of natural gas fired power plants was divided equally between gas turbine and combined cycle. Since there is no information about the age distribution of the plants, the start year is distributed evenly between today and the scenario year. Other sources for power plant capacity are EC Energy (2012), EC Energy (2013) and EIA (2014) but those only provide aggregated numbers for fossil fuel plant capacities.

- `lifetime_40years`: All existing power plants (same sources as the status quo 2012 table) have a lifetime of 40 years.

- `no_fossils`: Fossil power plant capacity is set to zero for all regions.

- `status_quo_2012_DE_lifetime_40years`: Power plants in Germany according to the BNetzA (2013) register have a lifetime of 40 years. Fossil power plant capacity in other countries equals the status quo of 2012 as provided by ENTSO-E (2014).

For defining own thermal power plant scenarios, one source of information regarding estimated future capacity could be the ENTSO-E System Outlook and Adequacy Forecast Reports (ENTSO-E, 2013).

Table 4 and Table 5 show examples what the `thermal_pp_scenario` could look like in the case of specifying the installed capacity (Table 4) or the lifetime (Table 5).

3 Start simulation

Although you have not even started the simulation, half of the work for you is done already by choosing all this pathway stuff! Now as you have got a `scenario_nr` it is the time to lay back and let the computer calculate and simulate the energy supply based on your assumptions. There are just a few last steps left before you can let it run....
In the folder `renpass/code_R_renpass` you can find the code file `code_R_start`. Open it with an editor, for example RStudio. At the beginning of the file you have to define `scenario_nr_vector`, `plot`, `path` and `system_software`. How to define `path` and `system_software` is described in 1.4. Normally you do not have to change anything about `database_host`, `database_port` and `database_unix.socket`. Different specifications for

scenario name	region_id	fuel	type	installed capacity [MW]	lifetime	limitation
standard	11001	gas	0	40.00	0	installed_capacity
standard	11001	hard_coal	0	100.00	0	installed_capacity
standard	11001	lignite	0	30.00	0	installed_capacity
standard	11001	oil	0	5.50	0	installed_capacity
standard	11001	refuse	0	0.00	0	installed_capacity
standard	11001	uran	0	0.00	0	installed_capacity
standard	11002	gas	0	1000.00	0	installed_capacity
...						

Table 4: pathway for thermal power plants defined by installed capacity

scenario name	region_id	fuel	type	installed capacity [MW]	lifetime	limitation
...						
standard	17000	hard_coal	0	0.00	40	lifetime
standard	17000	oil	0	0.00	40	lifetime
standard	17000	uran	0	0.00	40	lifetime
standard	18000	gas	0	0.00	40	lifetime
standard	18000	hard_coal	0	0.00	40	lifetime
standard	18000	oil	0	0.00	40	lifetime
standard	19000	gas	0	0.00	40	lifetime
standard	19000	oil	0	0.00	40	lifetime

Table 5: pathway for thermal power plants defined by lifetime

macs are already included in the code. If you followed the installation description for MySQL and R (Wiese et al., 2013) you do not have to change anything about

```
database_username <- "root"
database_password <- "user123
```

mac users sometimes have to choose "root" instead of "user123" for the database password. If you have a different username and password for your database you have to change that.

You have to give the `scenario_nr` of the scenario to be calculated. This `scenario_nr` defines all the pathway specifications in the table `1_scenario_parameter` in the `pathways` database. It can also be a vector of scenario numbers that are then calculated one after the other.
For example if you want to calculate scenario 1 and 13 and 7, this can look like this in the code.

```
scenario_nr_vector   <- c(1,13,7)
```

You can choose which result plots should be generated. Write `"none"` if no results plots at all should be generated, `"all"` if they should and `"no_spatial"` if just those plots should be done that do not require additionally installed R-packages for spatial plotting.

Example:

```
plot                  <- c("no_spatial")
```

For the additional packages and other information about plotting, see 5.1. The plot code, especially the code for the spatial plots could still contains some bugs.

When those setting are done, run the whole script. All other subroutines/code files are called by the start script. As soon as the > sign appears at the beginning of the line and the little red "stop" button (in RStudio) disappears, the simulation is finished. If there is an error, the simulation stops and an error message is displayed. Read the error message carefully since it gives you the necessary information to solve this bug! Warning messages are worth reading as well, but they do not stop the simulation.
On a computer with several kernels, you can run different scenarios in parallel. You can open an additional RStudio window by typing `rstudio` in a terminal. Don't let your computer work to full capacity, there is a danger of overheating. Thus, if you have four kernels, run three RStudio terminals maximal.

4 Logfile

You can have a look at the progress of the simulation by looking at the respective log-file in the folder `/renpass/log`. It is a text file that can be opened with an editor. The name of the log-file contains the scenario number and the date of calculation. For example `scenario-8-2013-04-30.log` The log-file itself contains some basic information about the input parameters of the scenario. As soon as one time-step is finished during simulation, the number of the time-step is written down in the log-file. Thus for checking out which point in the simulation renpass is and for estimating how long it will take, just have a look at the respective log-file. At the end of the simulation, the run time in total and for the different parts is written at the end of the log-file.

5 Results

5.1 Plots

Some automatic result figures are plotted at the end of each simulation run in the folder `renpass/plot/plot_results`, if the plot output option is chosen in the code start settings (in the code file `code_R_start_renpass.R`). These plots give a first impression of the scenario run and can be a starting point for deeper analysis of the results. The options area "none" "all" or "no_spatial". For the spatial plots the required shape-files stored in the folder `renpass/plot/geodata`. These shape-files as well as the following additional R-packages are required for the option "all", thus including spatial plots:

- maptools
- colorRamps

If the plot code should - instead of automatically subsequent to the scenario calculation - be run for results already stored in the database and defined by a `scenario_nr` , the following has to be defined in the file `plot_code/code_R_plot_core.R`

```
scenario_nr          <- c(1)
plot                 <- "no_spatial"
path                 <- "/home/frauke/Documents"
system_software      <- "linux"
```

Then, the file `plot_code/code_R_plot_core.R` can be run.

5.2 Results Database

The results that need to be saved from a simulation depend on the type of research question a model ought to address. For renpass, the aim was to obtain information on the operation of energy system components with a high spatial and temporal resolution. Table 6 provides an overview of all tables in the `results` database which are filled out during each scenario run.

result table name	unit	per time step	per region	Σ all regions	per technology
co2	*ton*	x	x		
demand	MW	x	x		
electricity_production	MW	x	x		x
excess_vre_after_exchange	MW	x		x	
excess_vre_after_storage	MW	x		x	
exchange	MW	x			
exchange_after_storage	MW	x			
filling_level_indicator					
filling_level	GWh & $mio\ m^3$	x	x		x
over_demand	MW	x	x		x
price_before_exchange	€$/MWh$	x	x		
price_after_exchange	€$/MWh$	x	x		
residual_load_before_exchange	MW	x	x		
storage_consumption	MW	x	x		x

Table 6: renpass result tables

Figure 5 shows at which points, during the simulation, figures are extracted and saved in the associated result table. The flowchart is a summary of the detailed flowchart in Figure 7 in form of colored code clusters.

The results in the various result tables can be linked to the input parameters by the `scenario_nr`.

Some of the fourteen result tables are clarified below:

Electricity production comprises of the amount of electricity produced by dispatchable power plants (geothermal, biomass, thermal, hydro, storage turbines) and the maximal

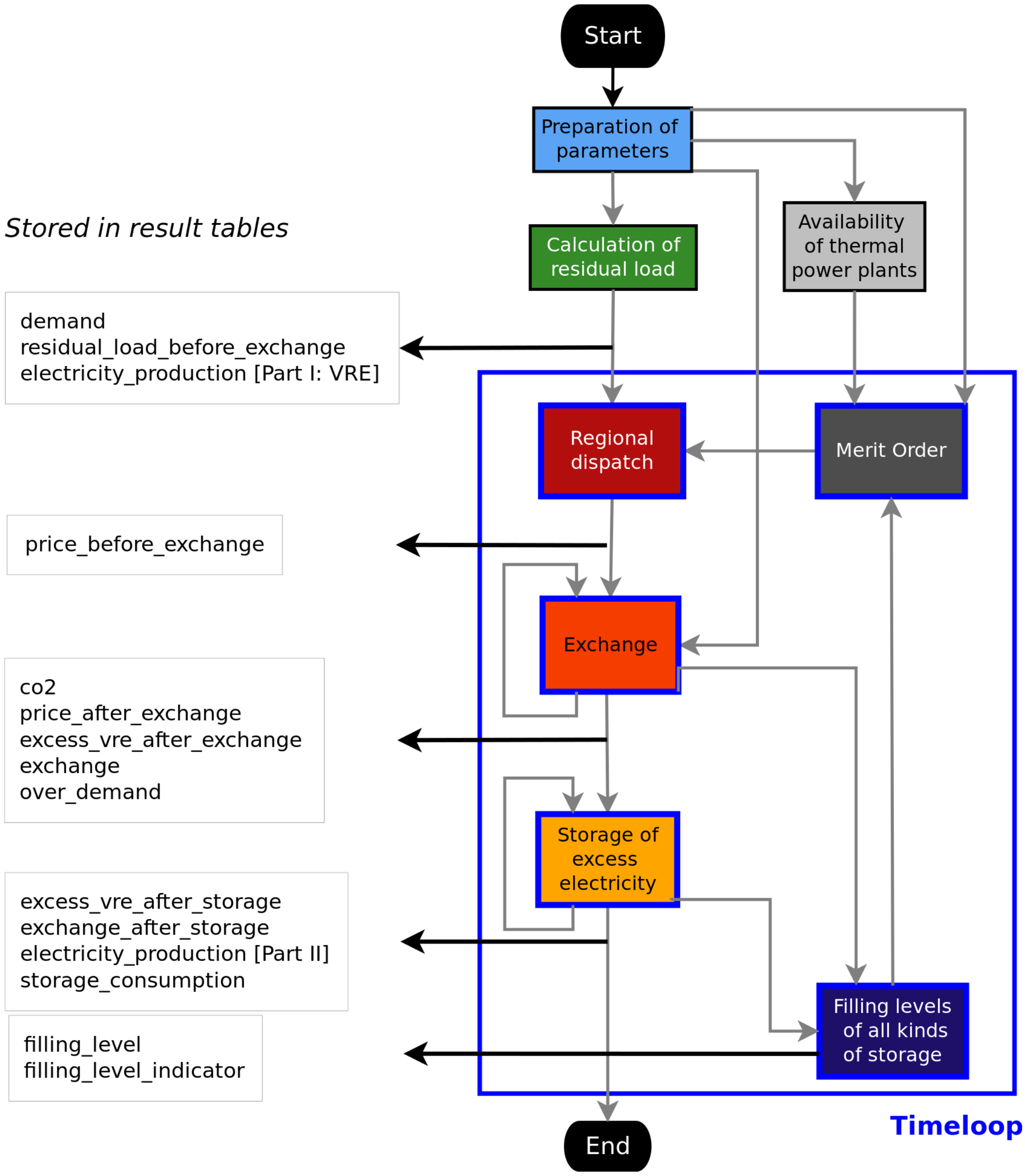


Figure 5: Overview flowchart of renpass showing at which steps in the simulation result parameters are extracted for the result tables.

amount of electricity that could be produced by VRE (solar, wind, run-of-river) if transmission and storage infrastructure allows for it. The generation side of all storage plants is included in this table, too.

Excess electricity of variable renewable generation (excess_vre) is the amount of electricity that cannot be used because there is (a) no demand or storage capacity to absorb it, or (b) there is not enough grid capacity to transport it to demand or to available storage capacities. It is saved at two different code steps in each time step of the simulation: Before storage capacities are used to absorb excess electricity, and afterwards. Of course, after storage, values of excess electricity cannot be higher than before storage. All values are saved as positive values.
`excess_vre_after_exchange` is the electricity generated by VRE that cannot be used either due to a lack of transport capacity or because the demand in the entire simulation area is already satisfied.
`excess_vre_after_storage` is defined as the electricity generated by VRE that cannot be used due to a lack of demand, transport and storage capacity.

The difference between the two values represents the excess electricity absorbed by storage utilities. The results concerning excess electricity are not assigned to a type of VRE, but represent the amount of VRE that have to be curtailed. Excess electricity cannot be clearly assigned or related to a region because it is shifted around during the exchange step the region this electricity is assigned to after the exchange is chosen randomly. Thus, only sums for the whole calculation area are saved.

Prices in renpass are derived from the marginal costs of the marginal power plant in the merit order. For fossil thermal power plants, they reflect short term costs, but for biomass plants and storage utilities they reflect opportunity prices and are indicators to place them in the merit order rather than real prices to cover costs. Thus, especially in scenarios without fossil power plants, prices do not necessarily reflect real costs. Instead, they can be used for comparison between scenarios rather than as absolute values. Prices are also an indicator of how often the scarcity price appears.
The average `price_before_exchange` of all regions should be higher than the average `price_after_exchange` because during exchange the total short term costs are minimized. Exporting regions can have higher prices after exchange. During storage, the price does not change, since only excess electricity with a price of 0 is stored.

Exchange result tables reflect the usage of the transmission grid. Exchange numbers are stored per connection and time step. Connections are always between two regions which are indicated by `plus_region` and a `minus_region`. The determination of plus and minus region is important to identify the direction of the flow. The leading sign of the numbers in the column `capacity_used` identifies the direction of the flow: If the value is positive, the electricity flows from the `plus_region` to the `minus_region` and vice versa.

Additionally, there are two columns for `grid_loss` in each exchange table. Grid losses of transmitted electricity between two regions are given in absolute $[MW]$ and relative values. They are saved as positive values.
The values in the table `exchange_after_storage` include the capacity usage status of the first exchange step before storage. Thus, the difference between the two exchange tables gives a rough estimation of the amount of electricity transferred due to storage. `exchange` is an

additional table and `exchange_after_storage` is the main exchange result table because it states the final transport needs.
There is an inaccuracy in `exchange` since the excess electricity after the exchange step is not necessarily assigned to its region of origin.

Filling levels of storage utilities are summed per region and type in the table `filling_level`. Filling levels are saved in energy values $[GWh]$ and volume $[million\ m^3$ for hydro, not yet defined for other storage mediums] per time step. `filling_level_indicator` contains indicators of the operational pattern of each reservoir during the simulation time. See Bökenkamp (2014) for a more detailed explanation of the filling level result tables.

6 What happens in there

Some more information on renpass and its applications can be found in Wiese et al. (2014) and Wiese (2015).

6.1 Overview: Functionality

Figure 6 illustrates the functionality of **renpass**. The installed capacities and expansion pathways of the different energy sources are set exogenously for the period to be analyzed. For each time step, the production of wind, solar and run-of-river electricity is subtracted from the demand. The so-called residual load is then supplied by the least expensive combination of the fully controllable production plants, storage units and grid utilization. The utilization of controllable capacity in **renpass** is based on regional dispatch in each grid region followed by balancing between the regions within the limits of the grid capacity.

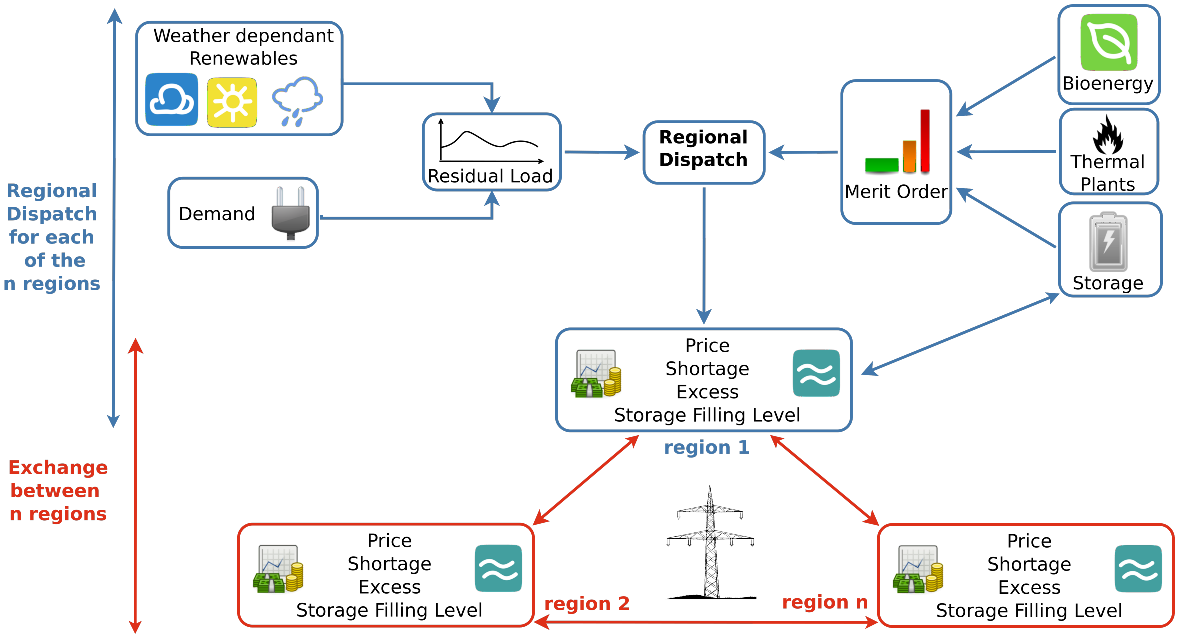


Figure 6: Schematic diagram of the regional dispatch and the exchange in renpass. Source: own illustration, icons from Open Icon Library and EnergyMap

6.2 Overview: Flow Chart

A possibility to gain an insight into the program sequence is the flow chart. Figure 7 shows the overview flowchart. All boxes with double lines at the sides stand for underlying subroutines. Blue framed boxes indicate which routines are repeated in each time step. Cluster of code are indicated in colors:

→ Parameter preparation (light blue)

→ Residual load calculation (green)

→ Thermal power plant availability (light gray)

→ Merit Order (dark gray)

→ Regional Dispatch (dark red)

→ Exchange (orange)

→ Excess electricity storage (light orange)

→ Adapting storage filling levels (dark blue)

6.3 Parameter preparation

This is displayed in light blue in the flowchart. All system and scenario settings, assumptions and input parameters from the database pathways for the given `scenario_nr` are read from the database and brought into shape for the following calculations. The region scenario is very important for the setup of all other pathway assumptions.

6.4 Residual load calculation

This part is displayed in green in the flowchart. The fluctuating renewables are used first to cover the demand. Thus, the demand for each time step per region is read from the database and adjusted when required, wind, solar and run-of-river production is calculated and then the so called must-run feed-in is subtracted from the demand for each region and each time step.

Demand is the consumption data from (ENTSO-E, 2012a). For Germany according to the description of the data from ENTSO-E, 91% of the whole electricity demand is covered (ENTSO-E, 2010). The rest is covered by industrial power plants not producing for the common grid. This minor part of electricity supply is not included in renpass.

Weather Data Raster weather data exists for the area shown in Figure 8. The CoastDat2 data set is from the Helmholtz-Zentrum Geesthacht (Geyer and Rockel, 2013). The Helmholtz COSMO-CLM data set is an extensive climate model set of historic climate data on an hourly basis. The raster covers the following area:
longitude: W 050.125 ° - E 051.125 ° – 234 entities

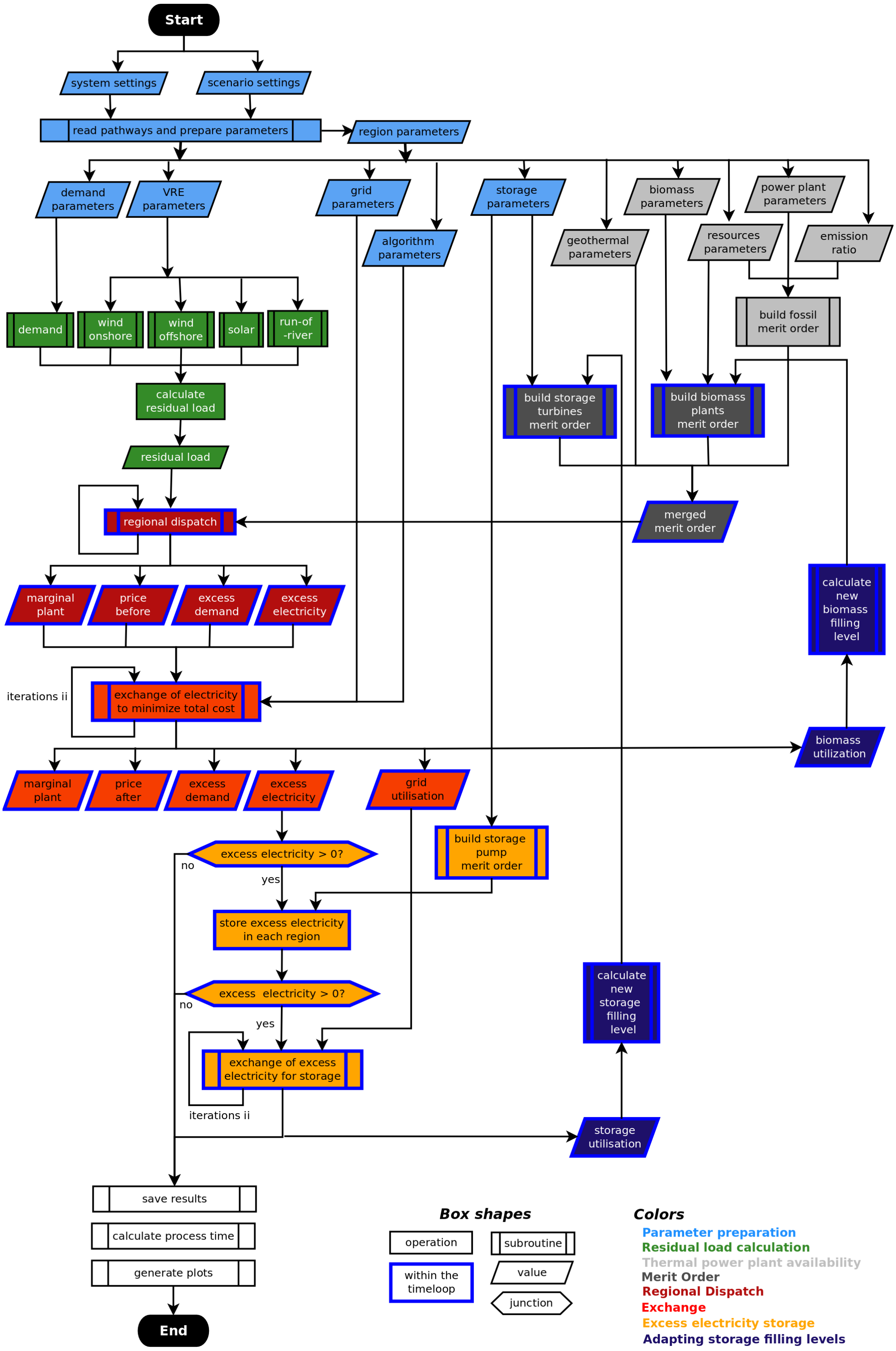


Figure 7: Program Flowchart of renpass core, the frame code of the model. Clusters of processing are indicated in colors. The shapes of the boxes indicate operations, underlying subroutines, values or junctions. Blue framed boxes are within the time loop.

latitude: N 30.125 ° - N 79.875 ° – 228 entities
Thus the resolution is 0.2 °.
Figure 8 shows the area that is covered by their data set and Figure 9 illustrates the utilized raster data points of this data set for renpass.

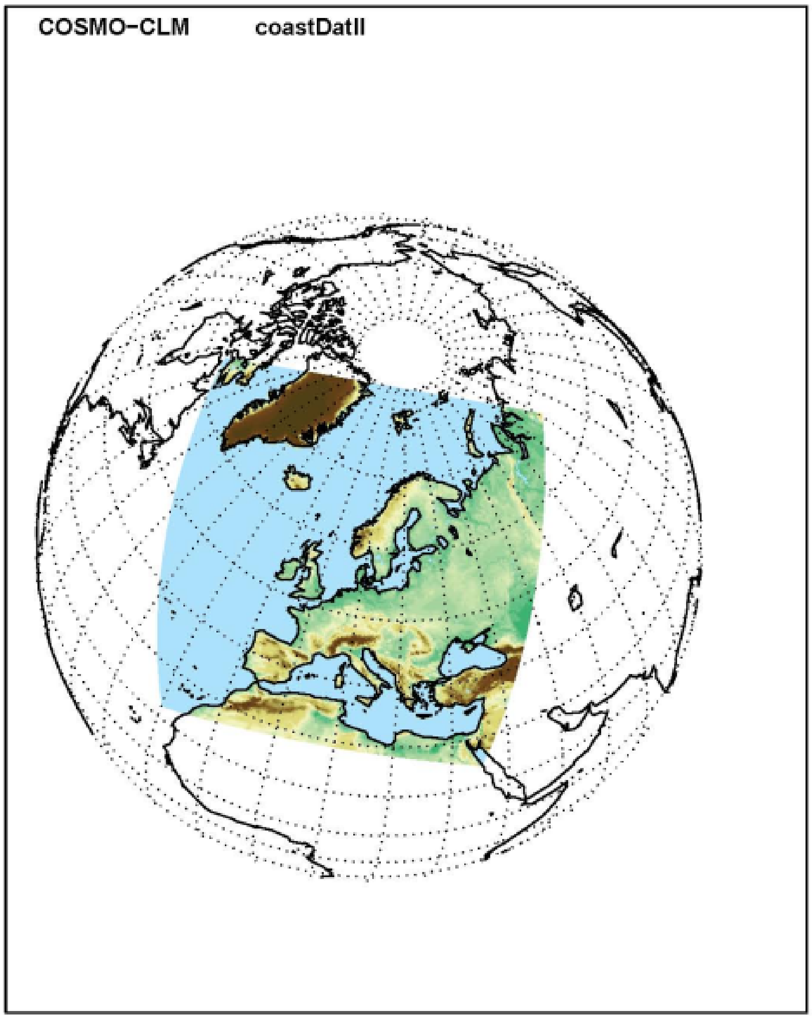


Figure 8: Area of Data from the Climate Model CoastDat2, (Geyer and Rockel, 2013)

Wind onshore Since open measuring data for wind speed is difficult to get for the area covered by **renpass**, wind speed raster data in combination with roughness values is used. Since the grid of (Geyer and Rockel, 2013) is very fine, the calculation time for the weather data would be very high if the whole data set was used. To keep the quality of feed-in calculation but also keep the runtime workable, only every third row and every third column of raster data is chosen for the wind time series. In the table `wind_onshore_raster_register`, raster_id, region_id which indicates the affiliation to the region, longitude, latitude, height above surface, and roughness of all raster points is stored. The column category indicates if the wind speed time series of the raster point is stored in the table `wind_onshore_raster_timeseries` (category = 1) or if it is an available raster point in the source data but not stored in the time series table and not utilized for the feed-in calculation (category = 2). The chosen installed capacity per region (that is chosen by you in the table `wind_onshore_scenario` in the pathways database) is evenly distributed to all available time series in the region. In comparison

to reality, this leads to rather an underestimation of wind feed-in and a smoothing of the total feed-in because the wind plants would rather be installed in the best wind locations. Figure 9 shows the chosen raster points for which there are time series in the renpass weather database in blue.
The data given is for 10 m above sea level. In the wind onshore code it is projected to the hub height of the utilized standard onshore wind turbine. The name of the power curve utilized can be looked up in the `code_R_wind_onshore.R`. The power plant curve name is the primary key of the wind-speed/electricity-output values in the table `wind_pp_parameter` in the renpass database. Standard is the power plant curve `Enercon_Vestas_REpower_3MW`. The turbine is a mixture of the wind power plants installed in Germany with a maximum output of 3 MW and a hub height of 80 m. Start wind speed is 3 m/s and stopping wind speed is 25 m/s.

Wind offshore For offshore wind, there are not many measuring stations. The German Weather Service offers Helgoland wind time series (DWD, 2013) and the Federal Maritime and Hydrographic Agency offers wind time series from three FINO stations (BSH, 2012) for German offshore areas. Thus like for onshore wind, European raster data from (Geyer and Rockel, 2013) is utilized.

The raster points were blended with the Exclusive Economic Zones (EEZ) of the countries in renpass. The EEZ shape files are taken from (VLIZ, 2012, version 7). For Norway, the offshore area was cut at 60 nm (nautical miles) north of the northernmost point of the Norwegian mainland at 72.171 °C N, since it is not very likely that there will be offshore wind farms around the icy and sparely inhabited area around Svalbard.

Figure 9 shows the chosen raster points for which there are time series in the renpass weather database in blue. Like for onshore wind, every third row and every third column of the available raster is chosen. All points are registered in the **renpass** weather database in the table `wind_offshore_raster_register`. In order to save storage space and processing time, only every third row and every third column of the available raster points is utilized in renpass. Additionally one raster point in Lithuania is taken since no raster point of the third row and column is located in the offshore area of Lithuania but the country does have an offshore wind potential. Figure 9 illustrates all renpass weather data points.

For Germany, the offshore area is divided into three regions:

- 11019: Baltic Sea
- 11020: North Sea, northeastern part off the coast of Schleswig-Holstein
- 11021: North Sea, southwestern part off the coast of Lower Saxony

Like for onshore wind, the installed offshore wind capacity in each region (chosen by you in the table `wind_offshore_scenario` in the pathways database) is evenly distributed to the raster points / time series of the respective region. The data given is for 10 m above sea level. In the `code_R_wind_offshore.R` the time series it is projected to the typical offshore wind turbine hub height. A offshore turbine standard power curve called `SiemensSWT36_Vestas112_REpowerM5` with a hub height of 100 m and 5 MW installed capacity is utilized. The values are stored in the table `wind_pp_parameter` in the renpass database. The roughness offshore is assumed to be 0.0002. This value can be adjusted in the `code_R_wind_offshore`. For further refinement, the wind data points could be restricted to the regions where water depth or any other circumstance that could encourage or hinder off-

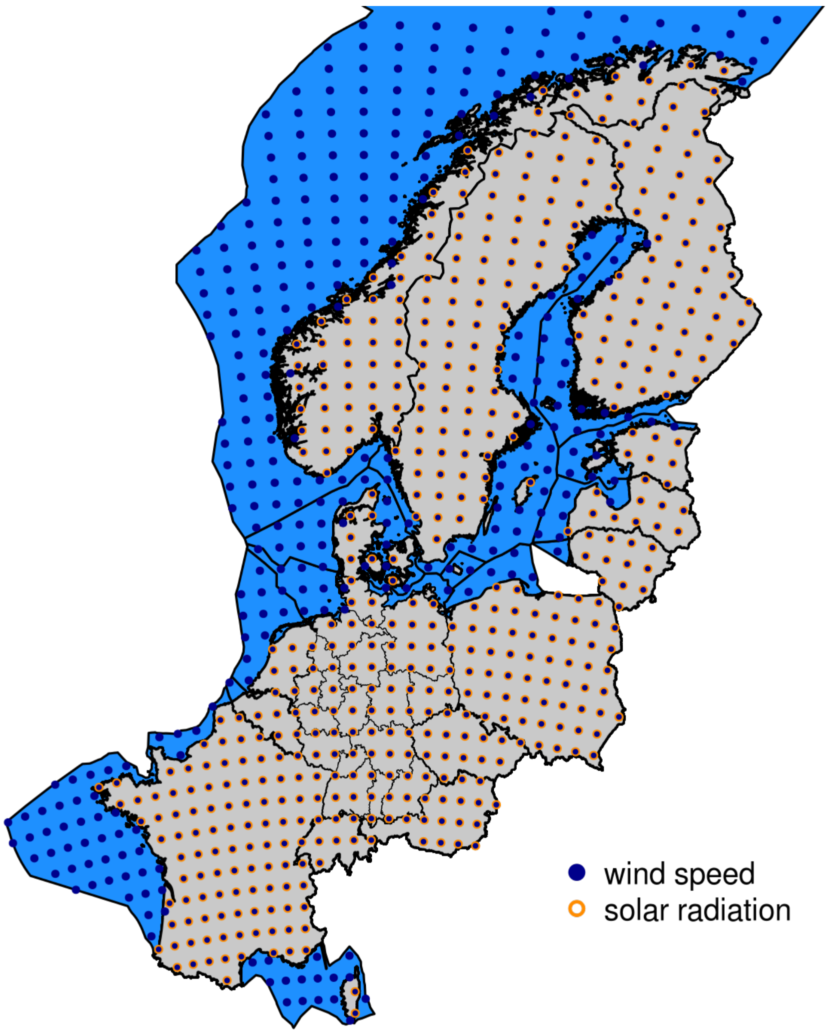


Figure 9: Weather raster data points wind speed offshore/onshore and solar radiation onshore. Source: own illustration based on Geyer and Rockel (2013); GADM (2012); OpenGeoDB (2013); VLIZ (2012)

shore wind farms is taken into account. But for now the method is considered to be accurate enough to be able to picture the offshore wind feed-in.

Solar Like the wind data, the solar data is provided by Geyer and Rockel (2013). Figure 9 shows the chosen solar raster points with an orange circle. Those are the same raster points as for onshore wind. Each region has at least one raster point, including Hamburg, which is the smallest region. All points are registered in the renpass weather database in the table `solar_raster_register`. Two hourly time series are stored for solar: diffuse and direct radiation on the surface $[W/m^2]$. In the solar code of renpass, those are summed up, which represents the global radiation. Based on a comparison of the modeled solar feed-in and real solar feed-in in 2010 a correction factor of 0.9 was introduced. That means 90 % of the mean global radiation for each region are converted into electricity. This value can be changed in the `code_R_solar.R`. With a more precise code and data availability about the orientation and technical properties of solar cells in the scenario-settings, the solar feed-in could be calculated more in detail.

Run-of-river For the compilation of run-of-river plants, several sources were used. The German run-of-river plants which are under the EEG are taken from DGS (2013), the older and bigger ones are not in there since they do not operate under EEG. Those were added mainly by a study made by Fichtner on behalf on the State Ministry of Economic Affairs and Employment (BMWI, 2003). Norwegian run-of-river plants are published by Norges Vassdrags- og Energidirektorat (2010). For other countries the run-of-river production is not modelled based on individual plants. Sources and setup of the German run-of-river register in detail can be found in Bökenkamp (2014).

For all countries except Germany and Norway the run-of-river production is modeled to be constant. The utilization of the assumed capacity and thus the level of production is determined by the selected weather year and varies between 0.45 and 0.65. See the file `code_R_run_of_river.R` for details. For Germany and Norway the run-of-river production curve is based on flow data.

For German run-of-river plants the average capacity utilization is determined by the selected weather year as for the constant production of the other countries. Inflow data from 40 level meters in German rivers is selected from the German hydrological yearbook (Deutsches Gewässerkundliches Jahrbuch). The run-of-river plants are assigned to level meters from the same river system or region. For each power plant the flow curve is transformed so that the maximum production is at installed capacity, the minimum is zero and the capacity utilization is at the determined value. The production of the plants is then aggregated to form run-of-river feed-in for every dispatch region. See the file `code_R_run_of_river_de` for details.

Norwegian inflow data was supplied by NTNU and SINTEF (2010). The power plant data includes information on the the total annual inflow and energy yield per m^3 for each plant. The inflow curve is scaled to represent the inflow for each plant that can be converted to electricity production with the energy yield. The production level of each time step is restricted by the installed capacity of the plant. See the file `code_R_run_of_river_no` for details.

Examples residual load The fluctuating renewable energies wind, solar and run-of-river are used first to cover the demand. The residual load is calculated by subtracting the so called must-run feed-in of wind, solar, and run-of-river from the demand. This is illustrated in

Figure 10.

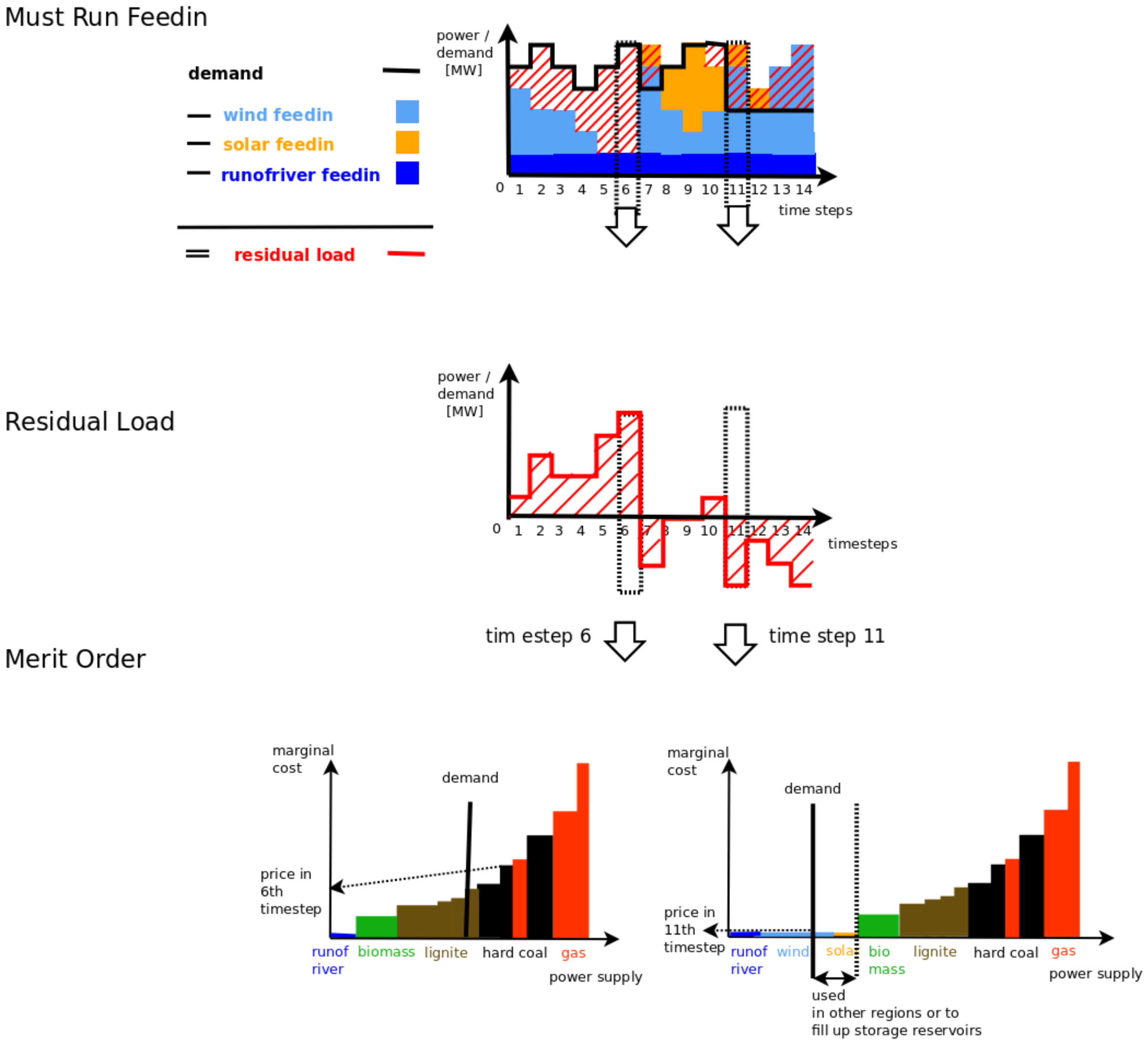


Figure 10: Regional dispatch in renpass

A schematic demand curve for 14 time steps is shown by the black line in the uppermost graph of Figure 10. The blue and yellow area reflect the must-run feed-in of wind, solar and run-of-river. Those three fluctuating electricity sources are dependent on meteorological conditions. Since they have marginal costs close to zero, they are utilized first. The shaded red area equates to the residual load: The difference between demand and must-run feed-in. In some time steps, the sum of the weather dependent feed-in wind, solar and run-of-river is higher than the demand. In those time steps, the residual load is negative. In the shown example this is the case in step 7, 11, 12, 13 and 14 (see Figure 10 middle graph).

6.5 Availability of thermal power plants

Geothermal power plants The installed geothermal capacity is multiplied with the utilisation factor that can be provided by the user in the geothermal scenario in the pathways database. A normal value is 0.9 that means 90% of the capacity installed, are all the time available.

The priority status of geothermal (utilized before other dispatchable power plants) is achieved by setting the geothermal price very low. The standard value in renpass is 0.1 €/MWh. This can be changed in the file `code_R_geothermal.R`.

Biomass Biomass plants are treated as thermal power plants and are dispatched via the merit order. The price depends not only on the price of biomass but also on the amount of biomass left. There is a certain amount of biomass given in GWh per year. If this amount is evenly distributed for each time step of the year, biomass bids with the average price. If more has already been used, the scarcity of biomass raises the price. The scarcity factor decides how fast the biomass is used up. The higher the scarcity factor, the higher the marginal cost and the slower the biomass gets used up since it gets very expensive to use. This scarcity factor is crucial for a good usage of the available biomass throughout the year. The standard scarcity factor in the code is 100. This can be changed in tht file `code_R_biomass_merit_order.R`.

The function to define the marginal cost can be found in `biomassMeritOrder` and looks like this.

```
if(amount_real <= amount_average){

     marginal_cost <- average_price *
                      (1 + (amount_average - amount_real) /
                      amount_average *
                      scarcity_factor)
 }
```

If less biomass than average would be is used until this time step, the price is lower than average biomass price.

```
if(amount_real > amount_average){

     marginal_cost <- average_price *
                      amount_average /
                      amount_real

}
```

After the dispatch the filling level of the available biomass has to be adapted. The biomass filling level cannot get negative. To be sure to fulfil this, in the `code_R_filling_level_tplus_bio` the filling level is set to zero if it gets negative. Thus no more biomass can be utilized.

Fossil power plants The availability of fossil power plants defines which plants bid for the merit order. There can either be the restriction by the installed capacity or by the lifetime, which was chosen in the thermal power plant scenario in the pathway database. All fossil thermal power plants in all regions need to have the same limitation parameter: either all are restricted by lifetime or all are restricted by installed capacity. The values per region and per fuel can of course differ. If the restriction chosen by the user is lifetime, the thermal power

plant register is reduced: All plants older than the chosen lifetime are taken out of the fossil power plant merit order for this calculation. If the installed capacity per fuel and region is defined by the user, the existing power plant register is either reduced or expanded by the amount of installed capacity needed to reach the values given by the user. The register of existing power plants in Germany is taken from BNetzA (2013). This register includes types and parameters of German power plants of more than 20 MW installed capacity. Based on the report 'Analysis of the unavailabilities of thermal power plants 2002 - 2011' published by VGB PowerTech (VGB, 2012), a general availability of thermal power plants of 85 % is considered. That means only 85 % of the installed capacity defined in the thermal power plant scenario can be utilize. This number can be changed in the `code_R_merit_order.R`.
For the merit order power plants are modeled individually but for power plant parameters like auxiliary consumption, mean efficiency, variable and fixed costs, they are clustered per fuel and type. The parameters of fossil power plants are mainly taken from Grimm (2007), the efficiency from dena (2008, p.54).
The efficiency depends on the age and the type and fuel of the plant. If no start up year is available and there is a zero in the database, the average age of all plants will be used instead. The mean efficiency is stored in the parameter table for each type of fossil plant (Table `thermal_pp_parameter` in the `renpass` database). This value is assumed for power plants that went into operation in 1980. The efficiency is changed by 0.3 percentage points (thus 0.003) for each year the power plant went into operation earlier or later. Younger power plants have a better efficiency. These values can be changed in `code_R_merit_order`.

The efficiency is calculated in the function `calculateEfficiency`.

```
efficiency <- measured_efficiency -
   auxiliary_power +
   ((scenario_year - efficiency_year) - age) * change_factor

 efficiency[measured_efficiency == 0] <- 0
 efficiency[is.na(measured_efficiency) == TRUE] <- 0
```

The calculation of the marginal costs of thermal power plants is done by the function `marginalCosts.R` like this:

```
# fuel_cost
fuel_cost                    <- (1 / efficiency) *
                                   3.6 *
                                   fuel_price

# co2_cost
co2                          <- (1 / efficiency) *
                                    3.6 *
                                    co2_emission_factor

fuel_cost[efficiency == 0]   <- 0
co2[efficiency == 0]         <- 0

co2_cost                     <- co2 * co2_price

# marginal_cost
```

```
marginal_cost              <- fuel_cost +
                              co2_cost +
                              variable_cost
```

For nuclear power plants, the marginal costs are set to 10.8 €/MWh, which is taken from a study that was done on behalf of the German ministry of Economic Affairs and Employment (BMWI, 2010, p.44). This value can be changed in the column `Cvar` in the table `thermal_pp_parameter` in the `renpass` database.

6.6 Merit Order

In each time step, the merit order has to be set up once again because the marginal costs of biomass plants and turbines have to be recalculated. Since their marginal costs for bidding into the merit order and the available capacity depends on the filling level of their reservoir or the still available biomass, this has to be done in the time step loop (tt) each time step again. The marginal costs for biomass plants, hydro plants and fossil power plants are calculated differently. Thus those merit orders have to be merged to one unique merit order (`code_R_merge_merit_orders.R`). The information passed to the merged merit order are the power plant number (`pp_nr`) for the identification of the power plant, the available capacity (`available_capacity`) that is offered by this plant, the `marginal_cost` in €/MWh and the CO_2 emissions in t CO_2/MWh. Each region has its own merit order ordered by the marginal costs and they are pooled as a list.

6.7 Regional Dispatch

Regional dispatch indicated in the flowchart (Figure 7) in dark red, matches the residual load with the merit order and finds a marginal power plant for each region, as well as a price [€/MWh]. Also it is determined if there is excess electricity [MW] or demand that cannot be met [MW].

Simulation In each time step, when there is remaining positive residual load, it is then covered by the dispatchable power plants: thermal power plants, hydro storage plants, and biogas plants are then used in the order of their marginal costs of operation. This is illustrated exemplary in the two merit order graphs in Figure 10. The merit order on the left shows the dispatch in time step six. In this time slice, there is not much must-run feed-in, just a bit of run-of-river which stands at the beginning of the merit order due to its marginal costs of close to zero. Then biomass, lignite, hard coal and gas power plants follow with their capacity on the x-axis in ascending order of their marginal costs of operation. To cover the demand, all the capacity of the biomass plants and some of the lignite power plants are utilized, they are to the left of the demand line. The point of intersection between demand and merit-order defines the price in the respective time step. In time step eleven, the must-run feed-in is much higher, thus the thermal power plants are shifted to the right. The order of wind and solar and run-of-river is just exemplarily in this picture. The demand intersects the merit order in the area of the must run capacities with practically zero marginal costs, therefore the price in this time step is zero.

The described dispatch is done for each region. Since the weather and the merit order can be different in every dispatch region, different prices can be observed after this regional calculation

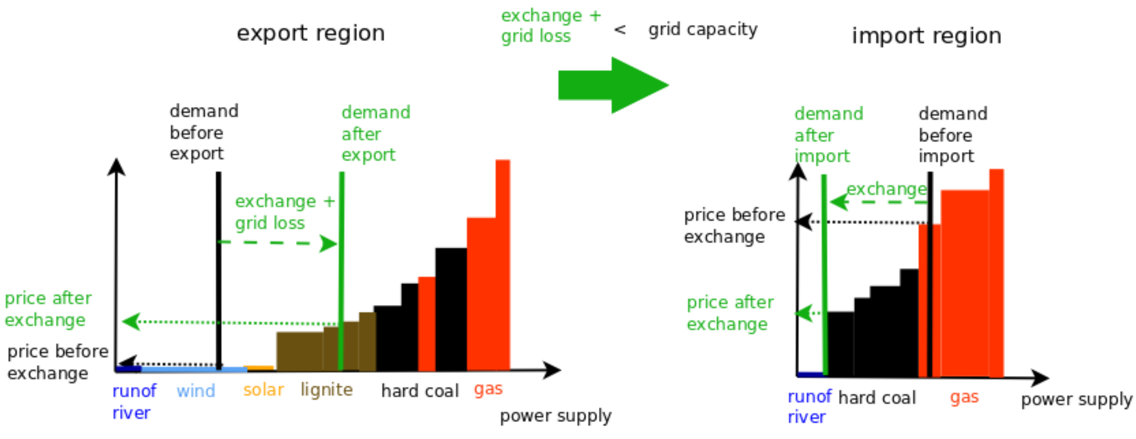


Figure 11: Exchange between Regions in renpass

step in renpass. The two example merit orders of two regions are shown in Figure 11 and their respective demand results in a price of zero in the first region and a higher price in the second region. In the first region, the must-run feed-in suffices to cover the demand, in the second one a gas power plant is the marginal power plant and sets the price. The black vertical demand line illustrates the demand in this region.

Then the exchange between the regions starts.

6.8 Exchange

In the schematic Figure 11, that illustrates the exchange mechanism, the first region becomes an export region, since the price in this region is lower than in the other. Thus, demand from the more expensive region is shifted to the region where electricity is cheaper in this time step. The green vertical line symbolizes the demand in each region after the exchange has taken place. In the export region the additional electricity needed to cover the grid loss has to be provided as well. The grid capacity between regions is the limiting factor for the amount of electricity that can be shifted. The n-1 criterion for grid stability is met by utilizing only 70% of the grid capacity of each line. This can be changed in the function file `capacityFromCircuits`. If the number of circuits is given, the capacity is calculated as follows:

```
capacity <- circuits *
            maximal_power *
            voltage *
            sqrt(3) *
            utilisation_factor /
            1000
```

Grid capacity is summarized for all circuits between each two regions. Figure 12 illustrates this principle.

For each connection line, grid loss is defined as a percentage of exchange, either the percentage number is directly provided in the grid scenario or calculated based on the length/distance of the connection provided in the grid scenario. Then the grid loss is calculated in the function `gridLossFromDistance.R`.

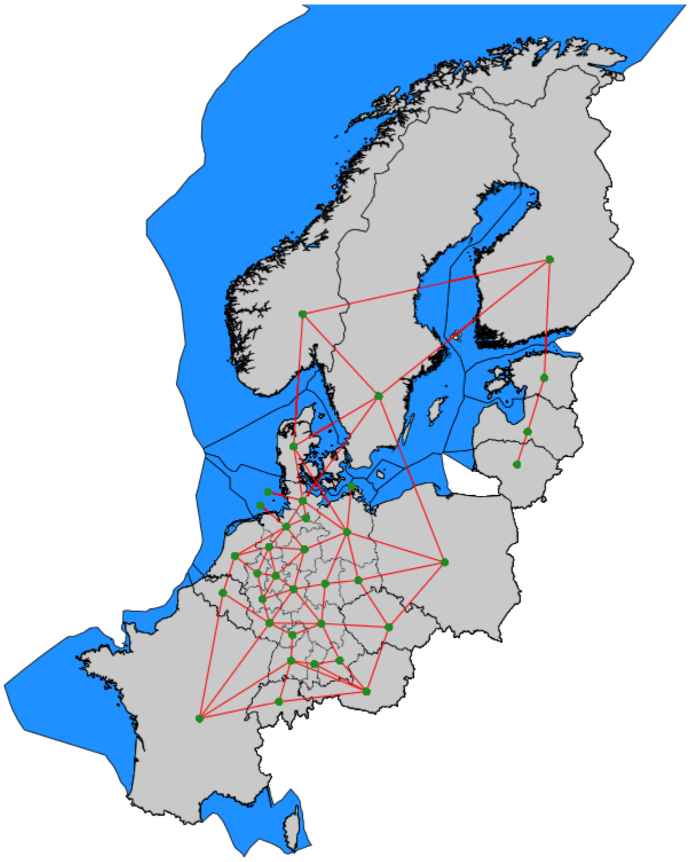


Figure 12: Summarized grid connections ENTSO-E (2012b); GADM (2012); VLIZ (2012) between regions

At the end of the exchange, the price will be the same in each region if the grid capacity is sufficiently large.

Optimization Problem The optimization problem for each quarter is to minimize electricity supply cost with the existing power plants for all regions. Thus after the regional dispatch, the exchange between the regions is carried out and restricted by the grid capacity. Since the problem is a very complex one, it is not calculated deterministically but solved with a heuristic procedure. The input is merit order, marginal power plant, excess energy (renewable energy that has to be cut off) and over demand (demand that cannot be supplied) for each region. Furthermore, the grid capacity between all regions is listed. With this information, the exchange algorithm has to find the least cost for all regions, thus the target function is to minimize the total costs of all regions with the existing infrastructure of grid, storage and installed capacities. The total cost is defined as the residual load times the price in each region summed up for all regions. This is implemented in the function `exchangeStandard`.

In `algorithm_scenario` in the `pathways` database, theoretically, another algorithm could be written, but so far only the standard one is testes and fully implemented so far. For another based on simulated annealing, a function called `exchangeSimulatedAnnealing` is already in the function files folder, but it is not tested yet. For further information about these two algorithms and their performance, see Hilpert (2012). If other algorithms should be added, a function has to be written and the name added in the file `code_R_exchange.R`.

In the standard exchange algorithm, there are no defined steps what is exchanged first. Randomly, two regions are chosen. Then an amount of electricity to be shifted is selected randomly as well. The upper limit of the amount of electricity exchange is the grid capacity between the two chosen regions. The exchange is executed by subtracting the amount of available energy in one region and adding it to the other region. Due to grid loss, the amount additionally available in the receiving region is smaller than the electricity leaving the exporting region. If the direction of electricity was in the opposite direction before, the grid loss already paid would have to be taken into account. This is dealt with in the function `grid_loss`. The amount of grid loss does not reduce the available capacity of the connection.

Due to the electricity exchange, the utilization of power plants in the two regions and the price can change. If the total cost for fulfilling the energy needs rises after the exchange, this exchange is not done and the next step starts. If the cost stays the same or gets less, the step is done, thus the electricity transfer takes place. Over demand will not be raised in any of the regions, because the aim is to supply the demanded electricity. Since the total cost is a product of electricity price and produced energy, it could also happen that the price in one region rises significantly, but the supplied energy is so small that it does not have a high influence on the total costs in this algorithm logic. The residual load is the decisive amount of energy, thus the energy supplied by renewables with 0 € marginal cost is not taken into account in the framework of this algorithm.

Depending on the number of regions and their interconnections, one can choose a number of iteration loops, which delivers sufficiently precise results. Figure 13 shows an algorithm test of the heuristic algorithm approach. The blue line indicates the optimal result. For test reasons, the heuristic renpass algorithm was started ten times with the same start conditions for an iteration loop of 5000 random exchange steps. In all cases, the result reached after 5000 iteration loops is in the range of results close enough to the real optimum.

The simulated annealing algorithms can be rahter used for the intensive study of similar scenarios. It costs more running time due to a higher number of calculated objects but the precision as implemented and tested in renpass is up to 10 % higher (Hilpert, 2012). This solver should rather be used if the exact result is more important than reasonable running time since it requires a time intensive setting of specific parameters (cooling function).

For a deeper explanation and formulas of the optimization problem see Wiese et al. (2014).

Since the algorithm for the exchange needs most computing power, it is crucial for the practicability of renpass to solve the trade-off between computing time and exactness of the results in a way that is best suited for the research question. If a high spatial resolution is for example not necessary, each country can be modeled as one region. In this case one assumes that grid bottlenecks within countries will not affect the outcome of the analysis.

6.9 Store Excess Electricity

Regional Storage The "left overs", thus, the must-run electricity with marginal costs of 0 € that cannot be used directly in the respective time step, are stored. This is done on a regional level first and then other regions are examined with respect to their availability of storage capacities. No additional power plants that have marginal costs and use fuel are started for storing energy.

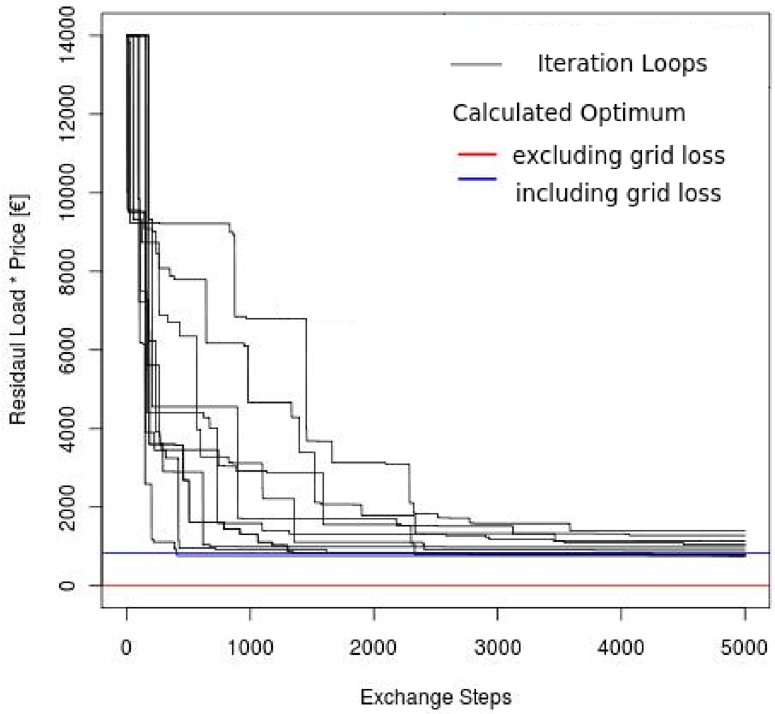


Figure 13: Example of Optimization loop with six regions and 8% grid loss

Exchange for Storage The remaining grid capacity can be used to transfer excess electricity that cannot be stored within the region to other regions with available storage capacity. The distribution of excess energy is similar to the exchange algorithm problem. Excess energy should be transferred to neighboring regions where it can fill up storage reservoirs. The algorithm can be simpler, but there is still a component of trial and error, since it can still happen that a region has to be used as transfer region and that there are different travel ways for the electricity. Furthermore the algorithm has to make sure that the excess energy is not thrown around meaningless and reduced just by being sent around and causing grid loss. The function `storeExcessElectricity.R` brings together excess electricity and storage capacity. It chooses one region that has excess energy, then chooses one region that has storage capacity. There a four stop criteria.

- `while(ex < exchange_loops){...}`
  number of max iteration steps is reached

- `if(sum(pump_temp) == 0){break}`
  if there are no storage possibilities left, the optimum is reached, since as much excess electricity as possible has been stored

- `if(sum(over_ee_temp) == 0){break}`
  if there is no excess electricity left, it can stop

- `if(0 %in% over_ee_temp == FALSE){break}`
  if there is excess electricity in each region, it is the same as if there is no pump anywhere, but to be safe, this criterion is additionally introduced

By this method the number of steps needed is limited and takes less running time than the main exchange algorithm.

If there is still excess electricity left after exchange and utilizing electricity for filling up storage reservoirs, fluctuating renewable plants have to be shut down. In the modelling no decision is taken which plant or technology has to be shut down. Shutting down of zero marginal cost renewable generation could be an indicator of an excess of installed capacity, a lack of transmission capacity or a first indicator of a lack of storage capacity.

6.10 Adapting storage filling levels

Indicated in dark blue in the flowchart (Figure 7), adapting storage filling levels takes place at the end of each time step as a preparation for the next one. For biomass plants the available biomass for the whole year of each region is reduced by the amount that was used for electricity production in the time step. For hydro and other storage plants the production of electricity and the storage of electricity have to be taken into account. Filling levels of reservoirs or other storages are accordingly increased or reduced. For hydro storage plants natural inflow is added to the filling levels where relevant. The new filling levels determine the availability of production and the marginal costs in the next time step. The initial filling level value for hydro storage is determined in `code_R_prepare_storage_connection` by the average filling state at that time of the year. Other storage reservoirs are set empty for the beginning of the calculation to make sure that the system does not get additional energy by emptying the reservoir without filling it up at a later point of time again. This can be changed by adapting `code_R_other_storage_scenario`.

7 Information for Interpretation

This idea of this section is to share gained experience with using renpass. It should be extended and improved step by step by your findings and experience with interpreting renpass results and your ideas for improving renpass. Some more information on renpass and its applications can be found in Wiese et al. (2014) and Wiese (2015).

7.1 Scarcity and overabundance prices

In the function for finding the marginal power plant (`marginalIndex`), the prices for extremes have to be defined. If the demand can be supplied with the given power plants, the price is in the range of the marginal costs. If there are no power plants at all in a region and there is excess demand (demand, that cannot be supplied), the price 1000 €/MWh is defined in the funciton `marginalIndex`. This is a randomly chosen price, with the condition to be much higher than the highest marginal costs of offering power plants. If the demand is too high, the price should be high as well, to give a signal for import need. The price does not change depending on the amount of demand that cannot be supplied in the current standard version of the renpass algorithm.

In the future, negative prices should also be implemented with the objective to serve as an indicator for excess electricity (= negative demand). Then the code for the objective of the target function would have to be calculated differently, so that the total cost (= price times

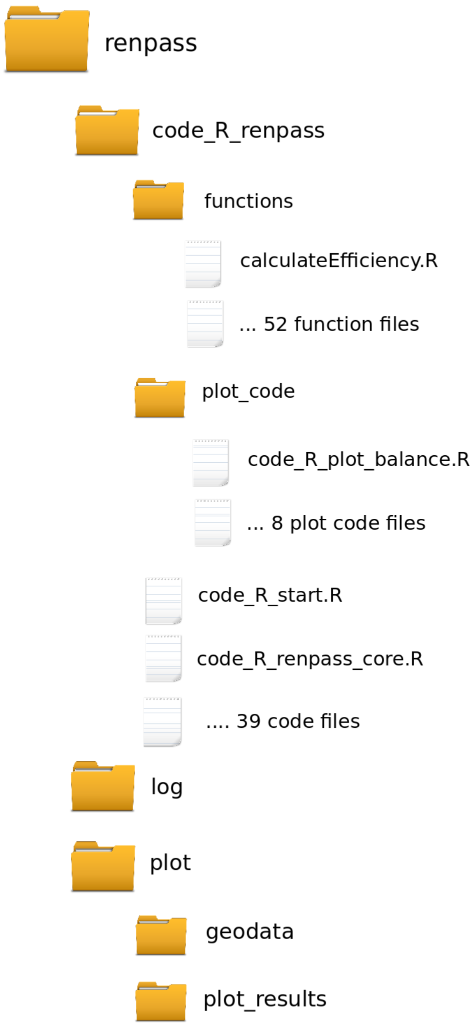


Figure 14: Folder structure. Icons from gettyicons.com, graphicsfuel.com

residual load) does not shrink again due to negative prices, which would send the wrong sign. The negative price will get important as a sign of expensive start-up costs of power plants. But this is not included in the model yet.

8 Broadening renpass

If you just want to run renpass you don't have to look into this section, but if would like to help to improve this model by finding bugs and to broaden renpass by adding new code pieces, better data, more clever algorithms, or whatever, you are very welcome! The following gives some important hints on database structure, source code, etc. This will hopefully be extended along with gained experience on collaboratively working on the model.

Functions and Subroutines The core of the model is the code processing the input data, simulating residual load, optimizing the dispatch and writing the results back into the database. It is written in the programming language R and organized in 39 files. Figure 14 illustrates the folder and file structure. Since the R language is organized in functions, 52 functions were defined for renpass. They are stored in the folder `/renpass/code_R_renpass/functions`.

They can be distinguished from code files containing more than just one function by the file name: functions are named with camelCasing and subroutine code files are named `code_R_...` and stored directly in the main folder.

Each code file starts with the short form of the license, required R-packages to run the respective code-file and information which renpass functions are applied in this piece of code. Each function-file additionally contains a standardized header including name, title, usage, arguments and resulting value of the function. This is the prearrangement in preparation for maybe publishing renpass as an R-package.

References

BMWI (2003). *Die Wettbewerbsfähigkeit von großen Laufwasserkraftwerken im liberalisierten deutschen Strommarkt.* Technical Report, Federal Ministry of Economic Affairs and Employment (Germany).

BMWI (2010). *Energieszenarien für ein Energiekonzept der Bundesregierung: Studie.* Technical Report, Prognos AG, EWI, GWS, on behalf of the Federal Ministry of Economic Affairs and Employment (Germany).

BNetzA (2013). *Kraftwerksliste der Bundesnetzagentur (27.03.2013).* Federal Network Agency (Germany). `http://www.bundesnetzagentur.de/DE/Sachgebiete/ElektrizitaetundGas/Unternehmen_Institutionen/Versorgungssicherheit/Erzeugungskapazitaeten/Kraftwerksliste/kraftwerksliste-node.html`, accessed April 12, 2013.

BSH (2012). *Fino Database.* Federal Minsitry of the Environment (Germany), Federal Maritime and Hydrographic Agency (Germany), Project Management Jülich (Germany), German Wind Energy Institute. `http://fino.bsh.de`, accessed May 28, 2012.

BSH (2013). *Bundesfachplan Offshore für die deutsche ausschließliche Wirtschaftszone der Nordsee 2012 und Umweltbericht.* Technical Report, Federal Maritime and Hydrographic Agency (Germany).

Bökenkamp, G. (forthcoming, 2014). *The role of Norwegian hydro storage in future renewable electricity supply systems in Germany: Analysis with a simulation model.* PhD thesis, University of Flensburg.

dena (2008). *Kurzanalyse der Kraftwerks- und Netzplanung in Deutschland bis 2020 (mit Ausblick auf 2030). Annahmen, Ergebnisse und Schlussfolgerungen.* Technical Report, German Energy Agency.

dena (2011). *Grid Study II. Integration of Renewable Energy Sources in the German Power Supply System from 2015-2020 with an Outlook on 2025.* Technical Report, German Energy Agency.

DGS (2013). *EEG Anlagenregister (Feb 2013).* International Solar Energy Society, German Section. `http://www.energymap.info/download.html`, accessed March 04, 2013.

DWD (2013). *Climatological time serie: hourly means of wind speed (in m/sec) de.dwd.nkdz.FFHM.* German Weather Service. Web Werdis Weather Request and Distribution System. `https://werdis.dwd.de/werdis`, accessed January 18, 2013 – May 16, 2013.

EC Energy (2012). *Market observatory and Statistics. Energy figures by country. 2012 country factsheets - EU 27 Member States.* Technical Report, European Commission Energy.

EC Energy (2013). *EU Energy in Figures. Statistical Pocketbook 2013.* European Commission Energy. `http://ec.europa.eu/energy/publications/doc/2013_pocketbook.pdf`, accessed February 11, 2014.

EIA (2014). *International Energy Statistics. Electricity Installed Capacity by Type 2011.* U.S. Energy Information Administration. `http://www.eia.gov/cfapps/ipdbproject/`, accessed January 07, 2014.

ENTSO-E (2010). *Load and consumption data: Specificities of member countries.* European Network of Transmission System Operators for Electricity. https://www.entsoe.eu/data/data-portal/glossary, accessed May 16, 2013.

ENTSO-E (2011). *Indicative values for Net Transfer Capacities (NTC) in Continental Europe. Winter 2010/11, working day, peak hours (non-binding values).* European Network of Transmission System Operators for Electricity. https://www.entsoe.eu/publications/market-reports/ntc-values/ntc-matrix/, accessed November 14, 2013.

ENTSO-E (2012a). *Consumption Data.* European Network of Transmission System Operators for Electricity. https://www.entsoe.eu/data/data-portal/consumption, accessed August 10, 2012.

ENTSO-E (2012b). *Interconnected network of ENTSO-E. Grid Map 1:4000000.* European Network of Transmission System Operators for Electricity. available at: https://www.entsoe.eu/publications/order-maps-and-publications/electronic-grid-maps/.

ENTSO-E (2013). *Scenario Outlook and Adequacy Forecast 2013 – 2030.* Technical Report, European Network of Transmission System Operators for Electricity, Brussels, Belgium.

ENTSO-E (2014). *Yearly Statistics & Adequacy Retrospect 2012. Background data tables. NGC.* European Network of Transmission System Operators for Electricity. Available at: https://www.entsoe.eu/publications/statistics/yearly-statistics-adequacy-retrospect/, accessed April 09, 2014.

EWEA (2013). *The European offshore wind industry - key trends and statistics 2012.* www.ewea.org/fileadmin/files/library/publications/statistics/European_offshore_statistics_2012.pdf, accessed April 22, 2014.

FSF (2007). *GNU General Public Licence Version 3 (June 29, 2007).* Free Software Foundation. http://opensource.org/licenses/GPL-3.0, accessed April 07, 2014.

GADM (2012). *GADM database of Global Administrative Areas (Version 2.0).* Boundaries without limits. www.gadm.org, accessed November 12, 2013.

German TSOs (2013). *Szenariorahmen für die Netzentwicklungspläne Strom 2014 - Entwurf der Übertragungsnetzbetreiber.* German Transmission System Operators 2012: 50Hertz, Amprion, Tennet, Transnet BW. http://www.netzausbau.de/DE/BundesweitePlaene/Charlie/SzenariorahmenCharlie/SzenariorahmenCharlie-node.html, accessed April 01, 2014.

Geyer, B. and Rockel, B. (2013). *coastDat-2 COSMO-CLM Atmospheric Reconstruction. World Data Center for Climate.* Institute of Coastal Research, Helmholtz-Zentrum Geesthacht, Germany. http://dx.doi.org/doi:10.1594/WDCC/coastDat-2_COSMO-CLM, accessed August 15, 2013.

Grimm, V. (2007). *Einbindung von Speichern für erneuerbare Energien in die Kraftwerkseinsatzplanung – Einfluss auf die Strompreise der Spitzenlast.* PhD thesis, Faculty of Mechanical Engineering, Ruhr University Bochum.

Hilpert, S. (2012). *Analyse und Optimierung des renpass-Modells.* Master's thesis, University of Applied Science Flensburg.

Norges Vassdrags- og Energidirektorat (2010). *Power Plants, Reservoirs and Inflow Fields.* Available at: http://www.nve.no, Norwegian Water Resoruces and Energy Directorate.

NTNU and SINTEF (2010). *Hydro inflow Norway.* Norwegian University of Science and Technology and SINTEF Energy Research.

OpenGeoDB (2013). *Geo Data Set of Germany.* General German Bicycle Association. http://fa-technik.adfc.de/code/opengeodb, accessed March 04, 2013.

VGB (2012). *Analyse der Nichtverfügbarkeit von Wärmekraftwerken 2002-2011, Ausgabe 2012, VGB-TW-103A.* Technical Report, VGB PowerTech e.V. and eurelectric. Available at: http://www.vgb.org/vgbmultimedia/KW_Statistik/TW_103+A_+2012_+Kurzinfo.pdf.

VLIZ (2012). *Maritime Boundaries Geodatabase, version 7.* Flanders Marine Institute. http://www.marineregions.org, accessed August 15, 2013.

Wiese, F. (forthcoming, 2015). *renpass - Renewable Energy Pathways Simulation System - Open Source as an approach to meet challenges in energy modeling.* PhD thesis, University of Flensburg.

Wiese, F., Bökenkamp, G., and Wingenbach, C. (2013). *Modelling with R and MySQL.* Center for Sustainable Energy Systems. Available at renpass.eu.

Wiese, F., Bökenkamp, G., Wingenbach, C., and Hohmeyer, O. (2014). *An open source energy system simulation model as an instrument for public participation in the development of strategies for a sustainable future. WIREs Energy Environ*, 3:490–504. doi: 10.1002/wene.109.