Hands On Exercise 4: Adding a technology

Learning objectives

- Learn how to add a new technology in MUSE

Addition of solar PV

Hands-on accompanying video: https://youtu.be/d_KlS4QL5mw

In this section, we will add solar photovoltaics to the default model. We will be starting from scratch and not continuing with the examples from hands-on 2 and 3. Therefore, to achieve this, we must modify the input files in the default example (default.zip) which can be found in the zenodo link provided below.

https://zenodo.org/record/6092287#.YgvOEy-l1pQ

Technodata Input

We must note, before starting, that we require consistency in input and output units. For example, if capacity is in PJ, the same basis would be needed for the output files `CommIn.csv` and `CommOut.csv`. In addition, across sectors a commodity needs to maintain the same unit. In these examples, we use the unit petajoule (PJ).

Next, we will edit the `CommIn.csv` file in the power sector, which specifies the commodities consumed by solar photovoltaics.

The table below shows the original `CommIn.csv` version in normal text, and the added column and row in bold.

ProcessNam	RegionName	Time	Level	electricity	gas	heat	CO2f	wind	**solar**
Unit	-	Year	-	PJ/PJ	PJ/PJ	PJ/PJ	kt/PJ	PJ/PJ	**PJ/PJ**
gasCCGT	R1	2020	fixed	0	1.67	0	0	0	**0**
windturbine	R1	2020	fixed	0	0	0	0	1	**0**
solarPV	**R1**	**2020**	**fixed**	**0**	**0**	**0**	**0**	**0**	**1**

Figure 4.1: Modified CommIn.csv file for the power sector

We must first add a new row at the bottom of the file, to indicate the new solar photovoltaic technology:

- we call this technology `solarPV`
- place it in region `R1`
- the data in this row is associated to the year 2020
- the input type is fixed
- `solarPV` consumes solar

As the solar commodity has not been previously defined, we must define it by adding a column, which we will call `solar`. We fill out the entries in the solar column, ie. that neither `gasCCGT` nor `windturbine` consume solar.

We repeat this process for the file: `CommOut.csv`. This file specifies the output of the technology. In our case, solar photovoltaics only output electricity. This is unlike `gasCCGT` which also outputs `CO2f`, or carbon dioxide.

ProcessNam	RegionName	Time	Level	electricity	gas	heat	CO2f	wind	**solar**
Unit	-	Year	-	PJ/PJ	PJ/PJ	PJ/PJ	kt/PJ	PJ/PJ	**PJ/PJ**
gasCCGT	R1	2020	fixed	1	0	0	91.67	0	**0**
windturbine	R1	2020	fixed	1	0	0	0	0	**0**
solarPV	**R1**	**2020**	**fixed**	**1**	**0**	**0**	**0**	**0**	**0**

Figure 4.2: Modified CommOut.csv file for the power sector

Similar to the the `CommIn.csv`, we create a new row, and add in the solar commodity. We must ensure that we call our new commodity and technologies the same as the previous file for MUSE to successfully run, i.e. `solar` and `solarPV`. Please note that the commodity names are case-sensitive.

Please note that we use flat forward extension of the values when only one value is defined. For example, in the `CommOut.csv` we only provide data for the year 2020. Therefore for the benchmark years, 2025, 2030, 2035â€¦ we assume the data remains unchanged from 2020.

The next file to modify is the `ExistingCapacity.csv` file. This file details the existing capacity of each technology, per benchmark year. For this example, we will set the existing capacity to be 0.5 for all technologies in the base year and 0 for the remaining years. Please note, that the model interpolates between benchmark years linearly.

ProcessNam	RegionName	Unit	2020	2025	2030	2035	2040	2045	2050
gasCCGT	R1	PJ/y	0.5	0	0	0	0	0	0
windturbine	R1	PJ/y	0.5	0	0	0	0	0	0
solarPV	**R1**	**PJ/y**	**0.5**	**0**	**0**	**0**	**0**	**0**	**0**

Figure 4.3: Modified ExistingCapacity.csv file for the power sector

Finally, the technodata.csv contains parametrisation data for the technology, such as the cost, growth constraints, lifetime of the power plant and fuel used. The technodata file is too long for it all to be displayed here, so we will truncate the full version.

Here, we will only define the parameters:`processName`, `RegionName`, `Time`, `Level`,`cap_par`, `Fuel`, `EndUse`, `Agent2` and `Agent1`

We shall copy the existing parameters from the windturbine technology for the remaining parameters that can be seen in the `technodata.csv` file for brevity. You can see the full file at the zenodo link, below:

https://zenodo.org/record/6092287#.YgvOEy-l1pQ

Again, flat forward extension is used here. Therefore, as in this example we only provide data for the benchmark year 2020, 2025 and the following benchmark years will keep the same characteristics, e.g. costs, for each benchmark year of the simulation.

A	B	C	D	E	F	T	U	V
ProcessNam	RegionName	Time	Level	cap_par	cap_exp	Fuel	EndUse	Agent2
Unit	-	Year	-	MUS$2010/I	-	-	-	Retrofit
gasCCGT	R1	2020	fixed	23.782344	1	gas	electricity	1
windturbine	R1	2020	fixed	36.3077118	1	wind	electricity	1
solarPV	**R1**	**2020**	**fixed**	**30**	**1**	**solar**	**electricity**	**1**

Figure 4.4: Modified Technodata.csv file for the power sector

Notice that we have hidden the cells between F and T. These are the same as the `windturbine` technology, but we've changed the `cap_par` input to 30 and the `Fuel` technology to `solar`.

Global inputs

Next, navigate to the input folder, found at:

```
{muse_installation_location}/src/muse/data/example/d
efault/input
```

We must now edit each of the files found here to add the new solar commodity. The edited files can be viewed in the zenodo link below:

https://zenodo.org/record/6092287#.YgvOEy-l1pQ

The `BaseYearExport.csv` file defines the exogenous exports for commodities. For our example we add a column to indicate that there is no export for solar. However, it is important that a column exists for our new commodity.

It is noted, however, that the `BaseYearImport.csv` as well as the `BaseYearExport.csv` files are optional files to define exogenous imports and exports; all values are set to zero if they are not used.

RegionName	Attribute	Time	electricity	gas	heat	CO2f	wind	**solar**
Unit	-	Year	PJ	PJ	PJ	kt	PJ	**PJ**
R1	Exports	2010	0	0	0	0	0	**0**
R1	Exports	2015	0	0	0	0	0	**0**
R1	Exports	2020	0	0	0	0	0	**0**
R1	Exports	2025	0	0	0	0	0	**0**
R1	Exports	2030	0	0	0	0	0	**0**
R1	Exports	2035	0	0	0	0	0	**0**
R1	Exports	2040	0	0	0	0	0	**0**
R1	Exports	2045	0	0	0	0	0	**0**
R1	Exports	2050	0	0	0	0	0	**0**
R1	Exports	2055	0	0	0	0	0	**0**
R1	Exports	2060	0	0	0	0	0	**0**
R1	Exports	2065	0	0	0	0	0	**0**
R1	Exports	2070	0	0	0	0	0	**0**
R1	Exports	2075	0	0	0	0	0	**0**
R1	Exports	2080	0	0	0	0	0	**0**
R1	Exports	2085	0	0	0	0	0	**0**
R1	Exports	2090	0	0	0	0	0	**0**
R1	Exports	2095	0	0	0	0	0	**0**
R1	Exports	2100	0	0	0	0	0	**0**

Figure 4.5: Modified BaseYearExport.csv file for the power sector

The `BaseYearImport.csv` file defines the imports in the base year. Similarly to `BaseYearExport.csv`, we add a column for solar in the `BaseYearImport.csv` file. Again, we indicate that solar has no imports.

RegionName	Attribute	Time	electricity	gas	heat	CO2f	wind	**solar**
Unit	-	Year	PJ	PJ	PJ	kt	PJ	**PJ**
R1	Imports	2010	0	0	0	0	0	**0**
R1	Imports	2015	0	0	0	0	0	**0**
R1	Imports	2020	0	0	0	0	0	**0**
R1	Imports	2025	0	0	0	0	0	**0**
R1	Imports	2030	0	0	0	0	0	**0**
R1	Imports	2035	0	0	0	0	0	**0**
R1	Imports	2040	0	0	0	0	0	**0**
R1	Imports	2045	0	0	0	0	0	**0**
R1	Imports	2050	0	0	0	0	0	**0**
R1	Imports	2055	0	0	0	0	0	**0**
R1	Imports	2060	0	0	0	0	0	**0**
R1	Imports	2065	0	0	0	0	0	**0**
R1	Imports	2070	0	0	0	0	0	**0**
R1	Imports	2075	0	0	0	0	0	**0**
R1	Imports	2080	0	0	0	0	0	**0**
R1	Imports	2085	0	0	0	0	0	**0**
R1	Imports	2090	0	0	0	0	0	**0**
R1	Imports	2095	0	0	0	0	0	**0**
R1	Imports	2100	0	0	0	0	0	**0**

Figure 4.6: Modified BaseYearImport.csv file for the power sector

The `GlobalCommodities.csv` file is the file which defines the commodities. Here we give the commodities a commodity type, CO2 emissions factor and heat rate. For this file, we will add the `solar` commodity, with zero CO2 emissions factor and a heat rate of 1.

Commodity	CommodityT	CommodityN	CommodityE	HeatRate	Unit
Electricity	Energy	electricity	0	1	PJ
Gas	Energy	gas	56.1	1	PJ
Heat	Energy	heat	0	1	PJ
Wind	Energy	wind	0	1	PJ
CO2fuelcom	Environment	CO2f	0	1	kt
Solar	**Energy**	**solar**	**0**	**1**	**PJ**

Figure 4.7: Modified GlobalCommodities.csv file for the power sector

The `projections.csv` file details the initial market prices for the commodities. The market clearing algorithm will update these throughout the simulation; however, an initial estimate is required to start the simulation. As solar irradiance as a fuel is free, we will indicate this by adding a final column.

Please note that the unit row is not read by MUSE, but used as a reference for the user. The units should be consistent across all input files for MUSE; MUSE does not carry out any unit conversion.

RegionName	Attribute	Time	electricity	gas	heat	CO2f	wind	solar
Unit	-	Year	MUS$2010/I	MUS$2010/I	MUS$2010/I	MUS$2010/I	MUS$2010/I	MUS$2010/kt
R1	CommodityP	2010	14.8148147	6.6759	100	0	0	0
R1	CommodityP	2015	17.8981481	6.914325	100	0.05291385	0	0
R1	CommodityP	2020	19.5	7.15275	100	0.08314119	0	0
R1	CommodityP	2025	21.9351853	8.10645	100	0.1200698	0	0
R1	CommodityP	2030	26.5092592	9.06015	100	0.1569984	0	0
R1	CommodityP	2035	26.5185186	9.2191	100	0.21487757	0	0
R1	CommodityP	2040	23.8518519	9.37805	100	0.27275673	0	0
R1	CommodityP	2045	23.9722222	9.19382934	100	0.35394801	0	0
R1	CommodityP	2050	24.0648147	9.00960867	100	0.43513929	0	0
R1	CommodityP	2055	25.3425925	8.8326256	100	0.54236558	0	0
R1	CommodityP	2060	25.5370369	8.65564253	100	0.64959187	0	0
R1	CommodityP	2065	25.3240742	8.48561271	100	0.78089262	0	0
R1	CommodityP	2070	23.3611111	8.31558288	100	0.91219338	0	0
R1	CommodityP	2075	22.2777778	8.15223313	100	1.07832169	0	0
R1	CommodityP	2080	22.2592592	7.98888337	100	1.24445	0	0
R1	CommodityP	2085	22.1759258	7.83195124	100	1.4253503	0	0
R1	CommodityP	2090	22.0370369	7.6750191	100	1.6062506	0	0
R1	CommodityP	2095	21.9444444	7.52425246	100	1.73877515	0	0
R1	CommodityP	2100	21.3981481	7.37348582	100	1.8712997	0	0

Figure 4.8: Modified projections.csv file for the power sector

Running our customised simulation

Now we are able to run our simulation with the new solar power technology.

To do this we run the same run command as previously in the anaconda command prompt:

```
python -m muse settings.toml
```

If the simulation has run successfully, you should now have a folder in the same location as your `settings.toml` file called `Results`. It must be noted, however, that if you update a value and re-run the model, the results folder will be overwritten.

The next step is to visualise the results using Excel.

We will use the PivotChart, similar to that shown in hands-on 1. The file to be used is the MCACapacity.csv file. For our visualisation we have selected a stacked area chart, but you are free to choose the type you like.

sector	power			
Sum of capacity	**Column Labels**			
Row Labels	**gasCCGT**	**solarPV**	**windturbine**	**Grand Total**
2020	0.5	0.5	0.5	1.5
2025	1.75	1.5	1.5	4.75
2030	1.7778	1.5	1.5	4.7778
2035	1.7778	1.5	1.5	4.7778
2040	1.7778	1.5	1.5	4.7778
2045	1.7778	1.5	1.5	4.7778
2050	2.1825			2.1825
Grand Total	**11.5437**	**8**	**8**	**27.5437**

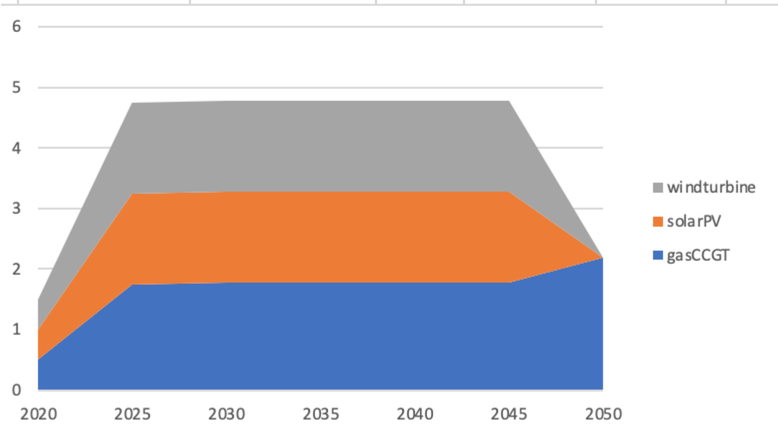

Figure 4.9: Visualisation with new technology.

The power sector now shows us the new `solarPV` technology.