OPTIPLANT tool

User guide

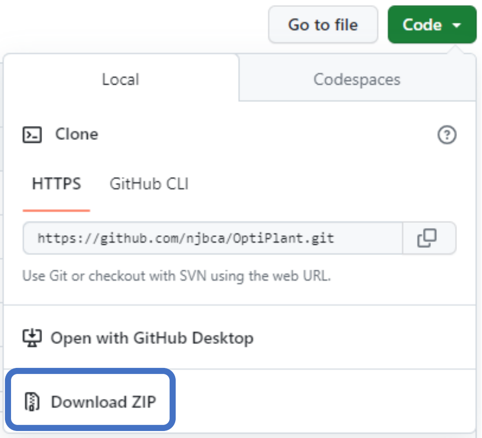
By David Garcia Marin

Table of contents

Using OptiPlant: summary

Using OptiPlant: Summary

1) Download all the OptiPlant ZIP folder from https://github.com/njbca/OptiPlant



-more info here-

2) Modify/tune the parameters found in the **Base > Data** folder:

Techno-ecnomical data ('Inputs')

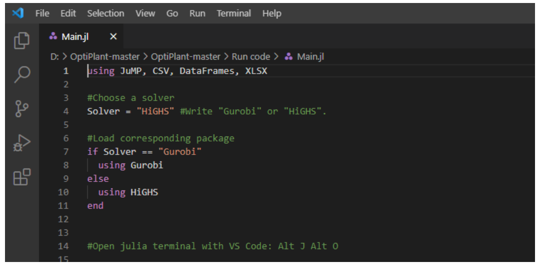
Wind/solar profiles and electricity prices ('Profiles')

-more info here-

3) Install the necessary software: Julia, VSCode and add the necessary packages (*JuMP, HiGHS, XLSX, DataFrames and CSV*)

-more info here-

Open the ***Main.jl*** Julia file found in the **Run Code** folder. Edit the code if necessary. Run the code file



-more info here-

4) Check the obtained outcomes (CSV) in the defined drectory inside the **Base > Results** folder

Import the CSV data to the 'Results' excel file found in the same folder to process and visualize the model outcomes

-more info here-

-(Extended and detailed instructions to install all the software and to run OptiPlant tool are presented in the next slides of this document)-

About OptiPlant

About OptiPlant

The **OptiPlant** model is designed in such a way that the input parameters, the optimization objective, variables or constrains, and the outcoming results can be modified in a fairly easy way. The solving time on a personal computer is usually below 5 minutes using an open-source solver.

For a more detailed description of the OptiPlant model such as the components and structure of the simulated plant, the mathematical description of the optimization model, the sources of the data inputs or other considerations, one can check the following article:

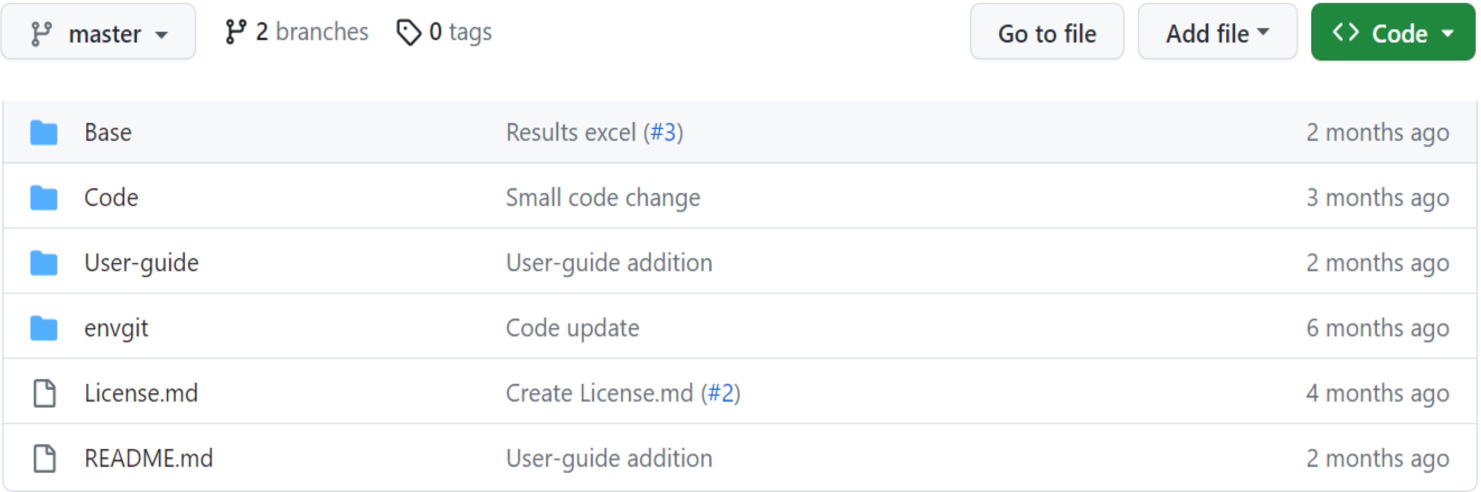
Nicolas Campion et al. “Techno-economic assessment of green ammonia production with different wind and solar potentials”. English. In: Renewable Sustainable Energy Reviews 173 (2023). issn: 1364-0321. doi: 10.1016/j.rser. 2022.113057.

https://www.sciencedirect.com/science/article/pii/S1364032122009388

About OptiPlant

OptiPlant is a linear optimization model that minimize the investment and operation costs of a power-to-X system that can be powered with wind, solar and the grid

All the documentation regarding OptiPlant (including this guide) and the tool itself can be downloaded directly from the website https://github.com/njbca/OptiPlant and clicking on <> Code => Download ZIP



[Fig.1]

As stated on the *README.md* file, the purpose of this document is to provide detailed user guide to correctly run and interpret the outcomes of the OptiPlant tool.

About OptiPlant

OptiPlant is a tool developed by Nicolas Campion from the DTU Department of Technology, Management and Economics that enables the user to model Power-to-X fuel production systems with a high variety of customizable input parameters and to optimize them according to different criteria. The model works under the 'dynamic power supply and system optimization' approach (DPS-Syst-Opt).

The fuel plant is modelled using a linear deterministic programming model which aims to minimize the fuel production cost of a PtX plant by managing the investments and operation of storage, power-supply and fuel production units under certain constraints. It assumes perfect foresight.

The **yearly fuel demand** is the main driver of the model, meaning that the model would minimize the fuel production costs of the PtX plant as long as the yearly fuel demand is fulfilled.

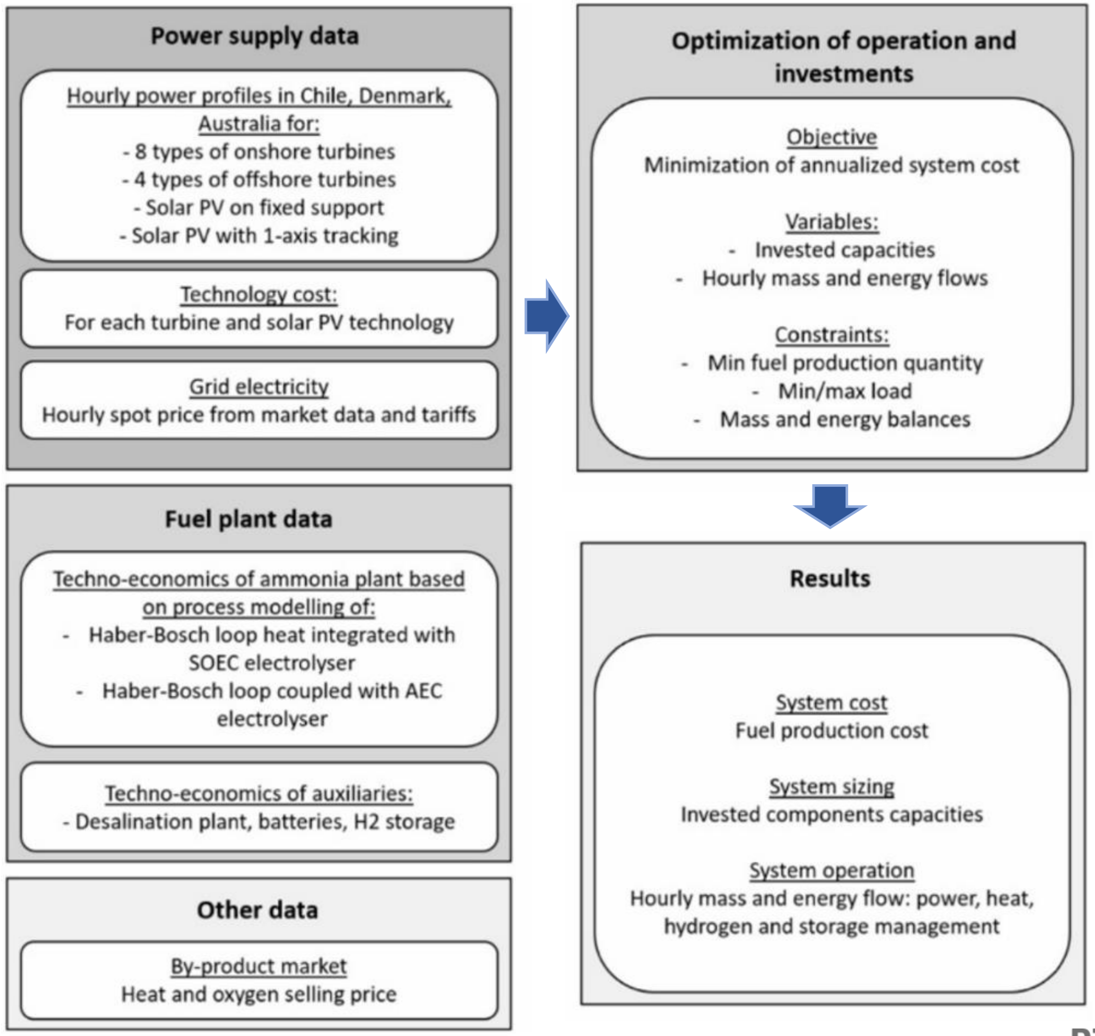
About OptiPlant

The system's optimization model has the following specifications:

- The model **input parameters** are: the techno-economic data of the different units, the hourly grid electricity prices, the hourly renewable power production profiles and the by-product market prices.

- The **goal or objective** of this model is to minimize the annualized system cost of the PtX power plant, using as variables the invested capacities and the hourly mass/energy flows. The system is constrained by a minimum fuel production quantity, the min/max load of the different units and the mass/energy balances between the different units.

- The **outcomes or results** of the model are the fuel production cost, the sizing of the different units of the system and the operation of the system (in terms of mass and energy flows).



[Fig.2]

Software installation

Software installation

In the following slides, an installation guide for all the necessary software to run **OptiPlant** is provided. All the needed software programs are listed below:

- **Julia**: The programming language that we are going to use to formulate the optimization problems. For download and documentation, visit https://julialang.org/.

- **Visual Studio Code**: An editor for writing and executing your Julia code. For download and documentation, go to the link https://code.visualstudio.com/. *(You are welcome to work with any other editor of your choice, for example: Jupyter notebook or Atom).*

- **JuMP**: A package embedded in the Julia programming language. It allows users to write optimization problems. For documentation, see https://jump.dev/JuMP.jl/stable/.

- **HiGHS**: An open-source high performance solver for linear programming problems (LP). For documentation, go to the page https://highs.dev/. Alternatively, the commercial soler **Gurobi** can also be used.

- *The installation guide of other necessary specific packages to read and clean data, visualize and plot results, etc. is also provided in this document.*

Software installation

All the items mentioned in the previous slide work together as follows:

An optimization problem is written in the **Julia** programming language, using the **JuMP** package syntax and **VS-Code** as a text editor. Various data is imported from CSV or excel files using *specific packages*. Then, all the info is passed to the solver **HiGHS (other solvers can also be used)**, which finds an optimal solution to the problem by using a variety of optimization approaches and techniques. Finally, the obtained results can be exported as CSV files or plotted in graphs using other *specific packages*.

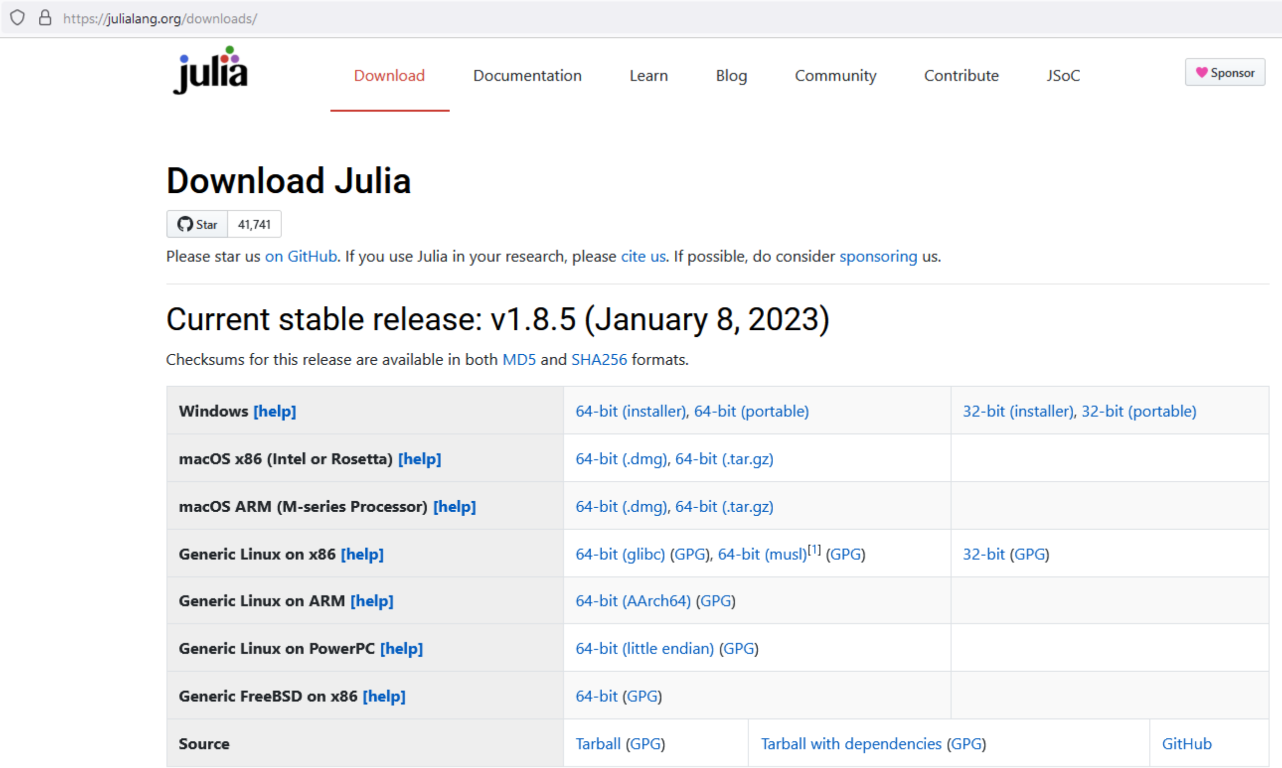
Installation guidance for each of these software can be found in the following slides.

Julia installation

Julia is a high-level, general-purpose, dynamic programming language. Its features are well suited for numerical analysis and computational science

<u>Installation steps:</u>

1) Go to: https://julialang.org/downloads/ and download the Julia version corresponding to your operating system.



https://julialang.org/downloads/

julia

Download | Documentation | Learn | Blog | Community | Contribute | JSoC | Sponsor

Download Julia

Star 41,741

Please star us on GitHub. If you use Julia in your research, please cite us. If possible, do consider sponsoring us.

Current stable release: v1.8.5 (January 8, 2023)

Checksums for this release are available in both MD5 and SHA256 formats.

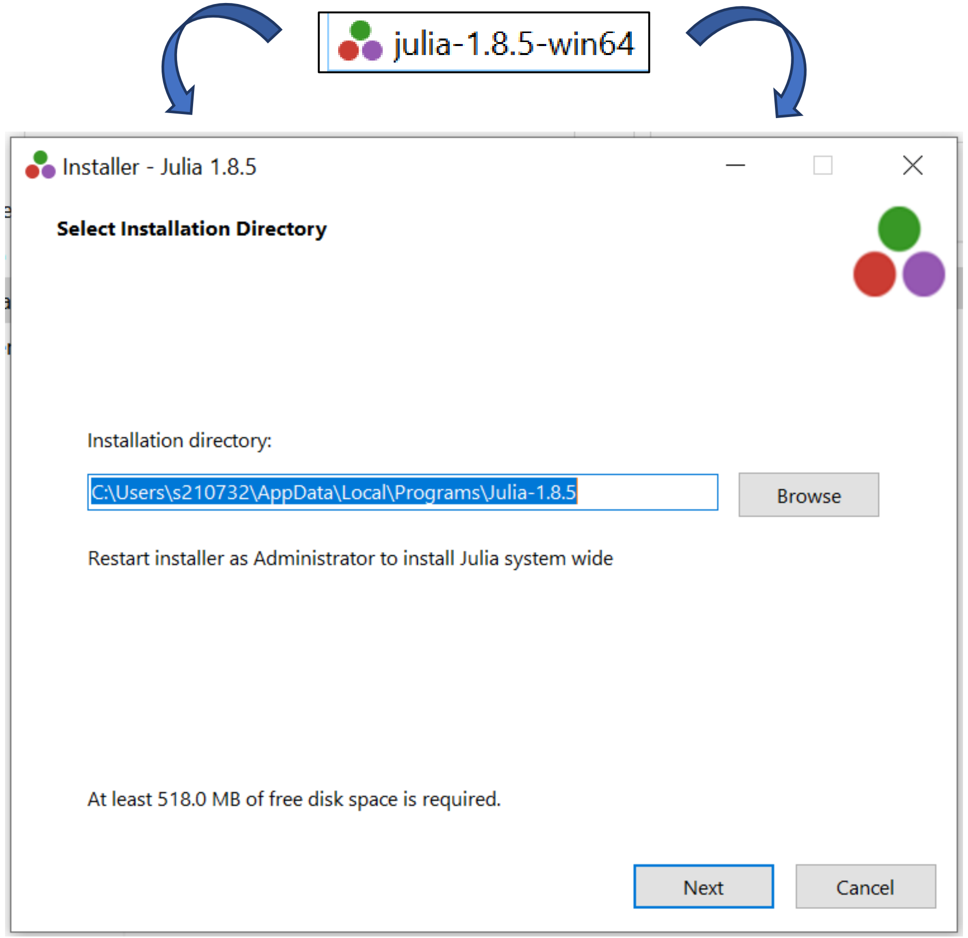
Windows [help]	64-bit (installer), 64-bit (portable)	32-bit (installer), 32-bit (portable)	
macOS x86 (Intel or Rosetta) [help]	64-bit (.dmg), 64-bit (.tar.gz)		
macOS ARM (M-series Processor) [help]	64-bit (.dmg), 64-bit (.tar.gz)		
Generic Linux on x86 [help]	64-bit (glibc) (GPG), 64-bit (musl)[1] (GPG)	32-bit (GPG)	
Generic Linux on ARM [help]	64-bit (AArch64) (GPG)		
Generic Linux on PowerPC [help]	64-bit (little endian) (GPG)		
Generic FreeBSD on x86 [help]	64-bit (GPG)		
Source	Tarball (GPG)	Tarball with dependencies (GPG)	GitHub

[Fig.3]

Julia installation

Installation steps:

2) Run the Julia installer and install the program:



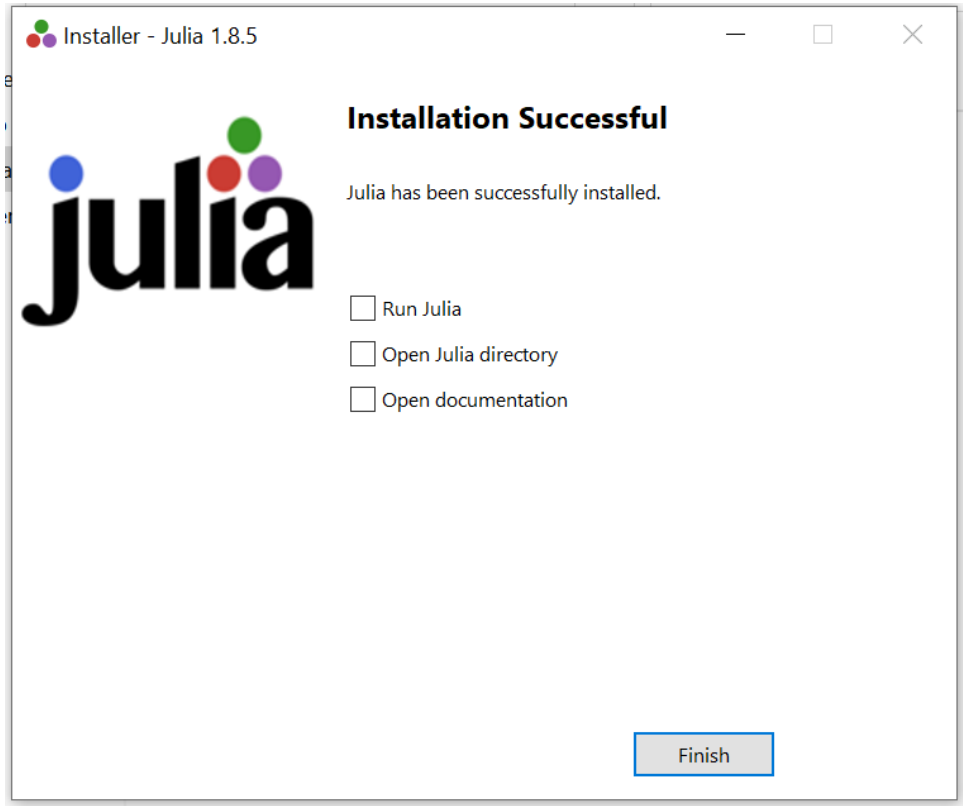
[Fig.4]

❗ *Tick the box 'Add Julia to PATH' **only if** you already have Visual Studio Code already installed on your PC*

Julia installation

Installation steps:

3) If the installation is successful, this message will appear:



You just got Julia on your PC!

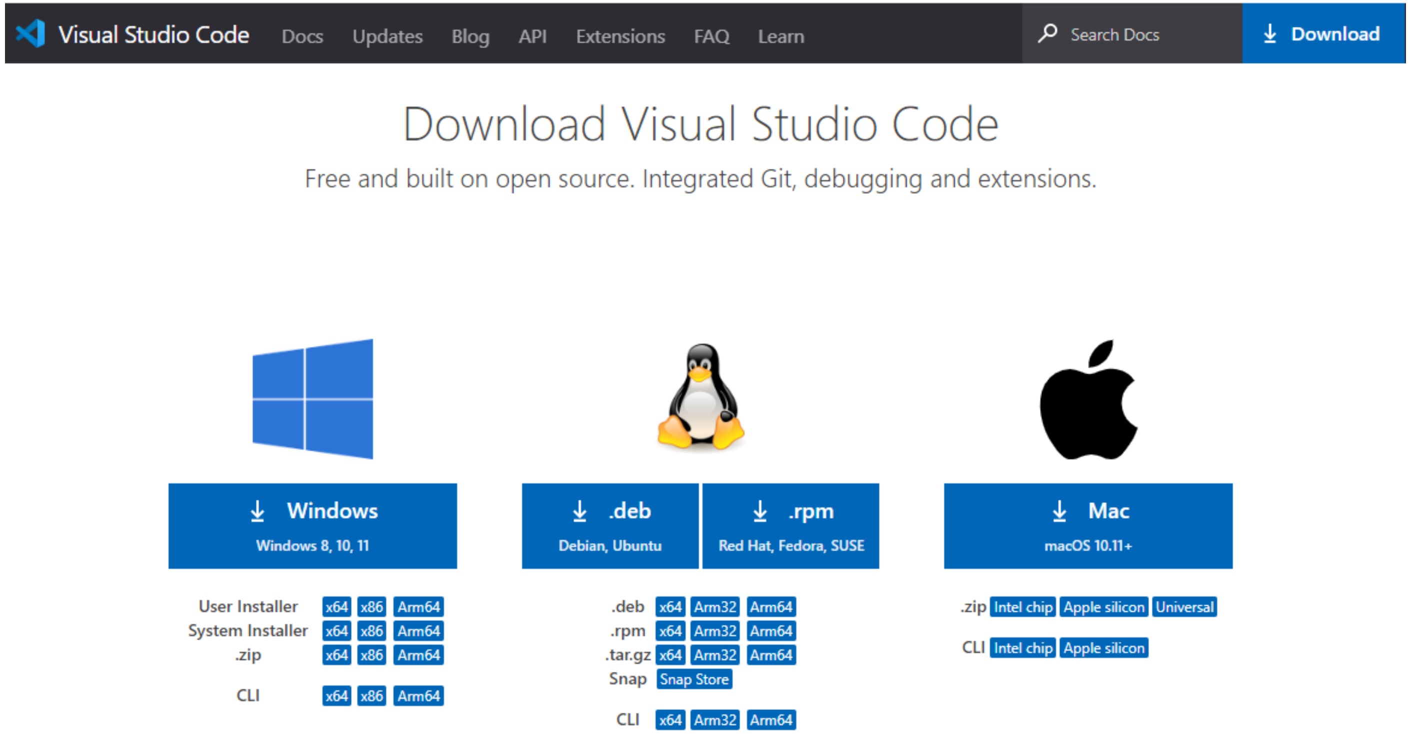
[Fig.5]

Visual Studio Code installation

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux.

Installation steps:

1) Go to the website: https://code.visualstudio.com/Download and download the version corresponding to your operating system.

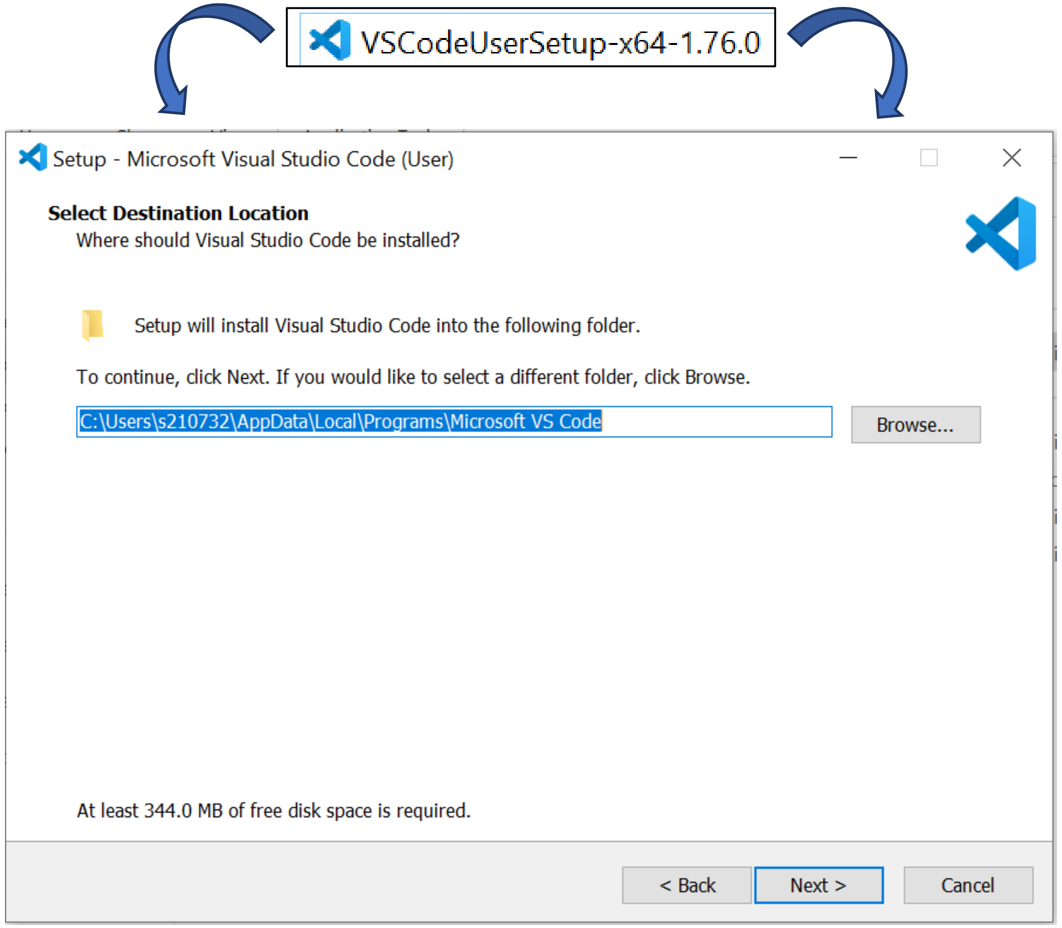


[Fig.6]

Visual Studio Code installation

Installation steps:

2) Run the Visual Studio Code installer and install the program

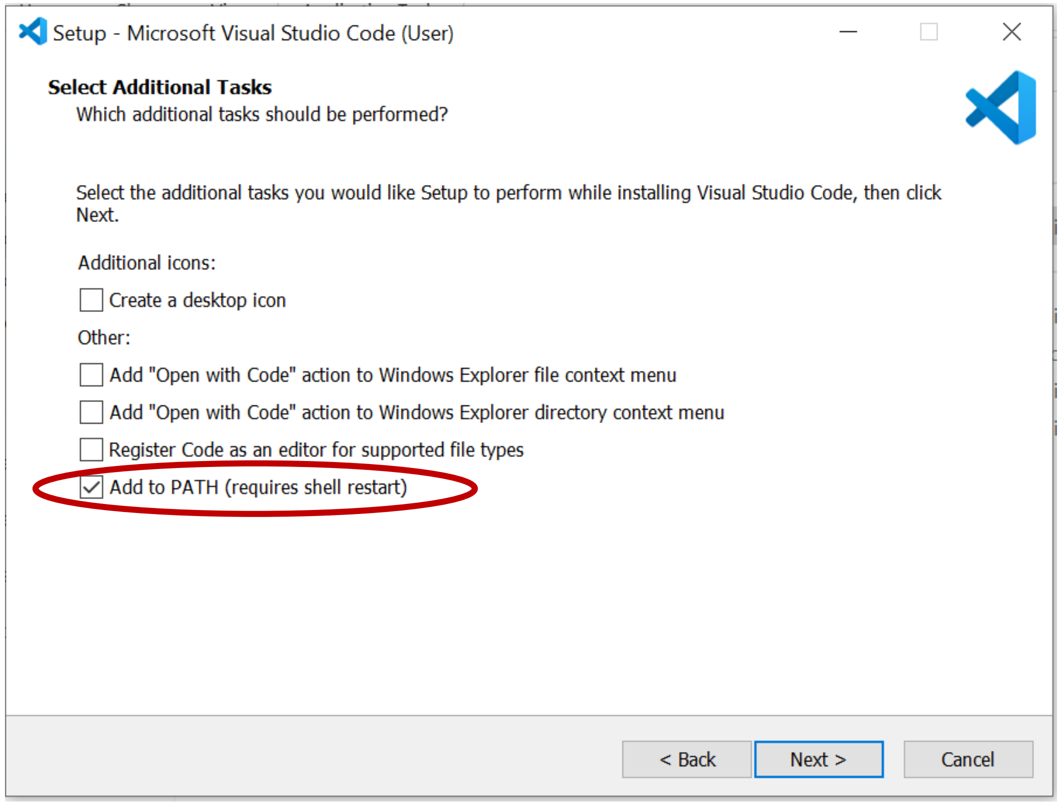


[Fig.7]

Visual Studio Code installation

Installation steps:

❗ IMPORTANT NOTE: During the installation process, remember to tick the option 'Add to PATH (requires shell restart)':

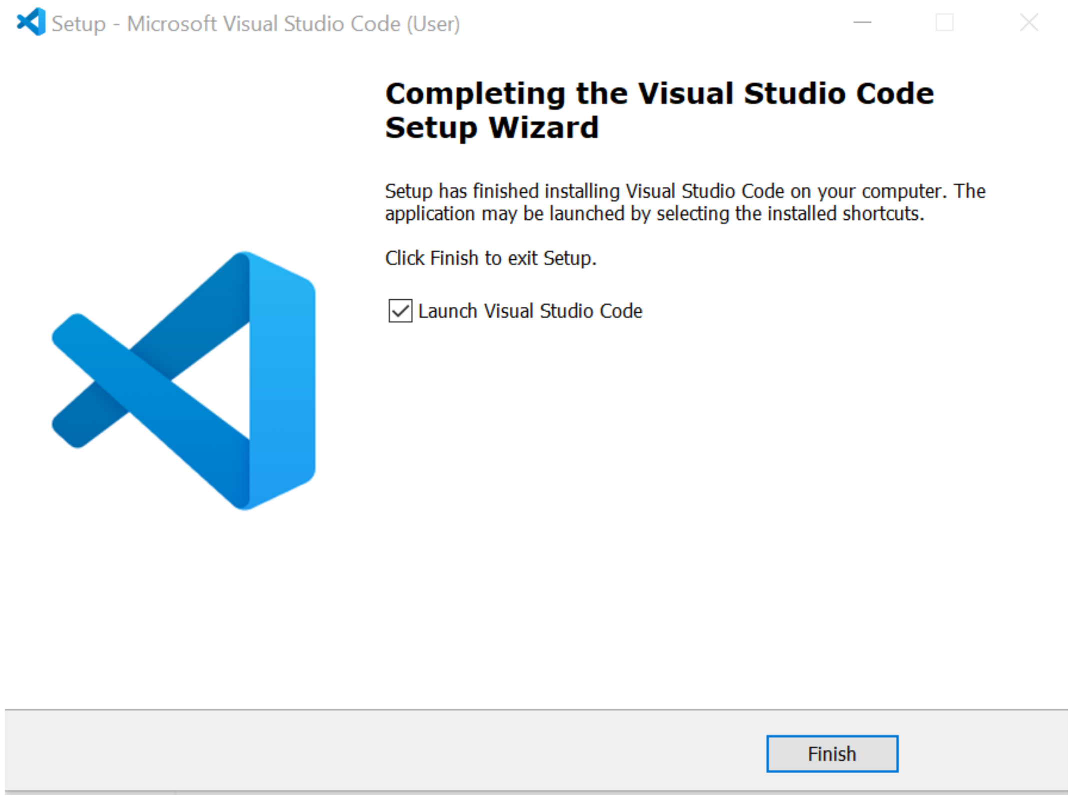


[Fig.8]

Visual Studio Code installation

Installation steps:

3) If the installation is successful, this message will appear:



[Fig.9]

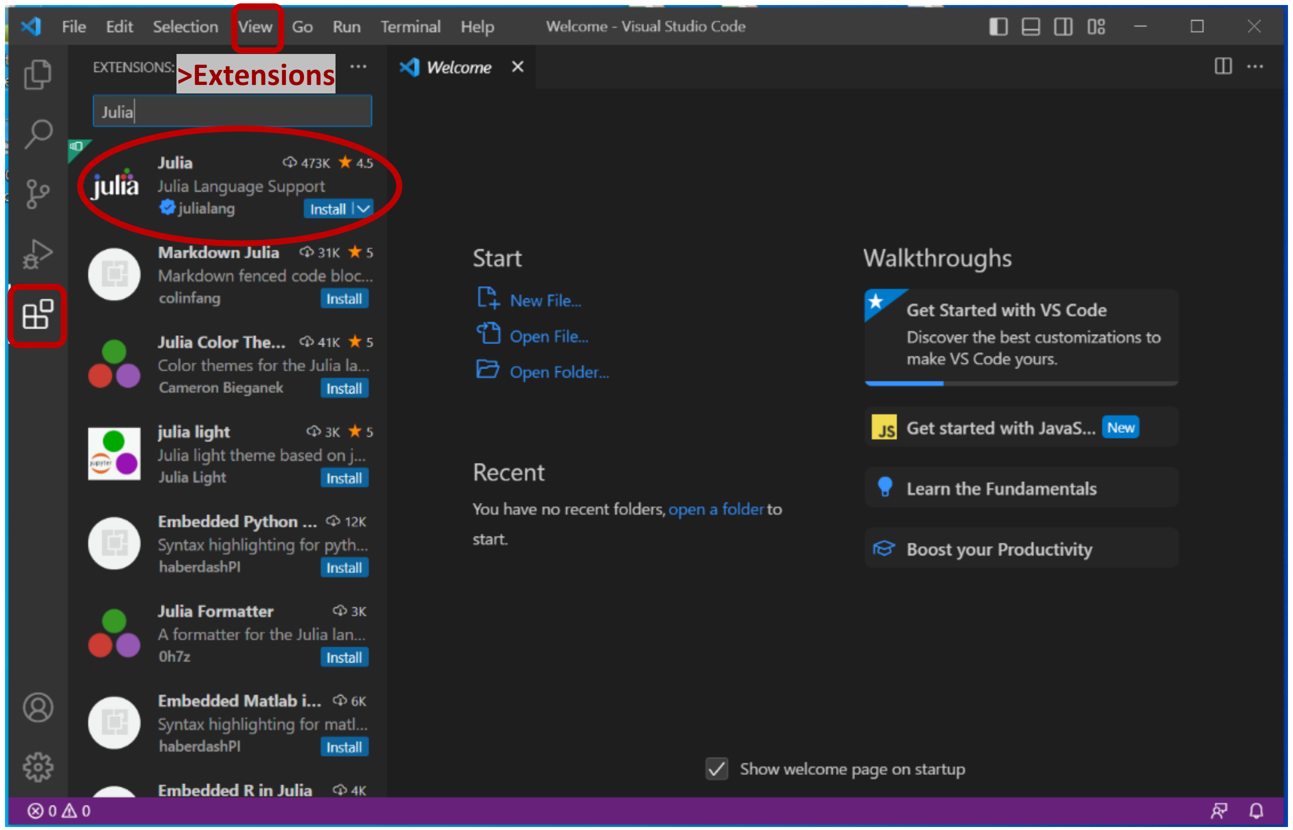
You just got VS Code on your PC! Next step is to add the corresponding extensions and save them in an ‘environment’.

Packages installation (on VS Code)

In order to be able to run the OptiPlant model in Visual Studio Code, several extensions and packages need to be installed.

Julia extension

Provides support for the Julia programming language in the VS Code editor. To install it, open VS Code and go to 'View > Extensions' or click in the fifth icon on the left. Type 'Julia' and install the extension.



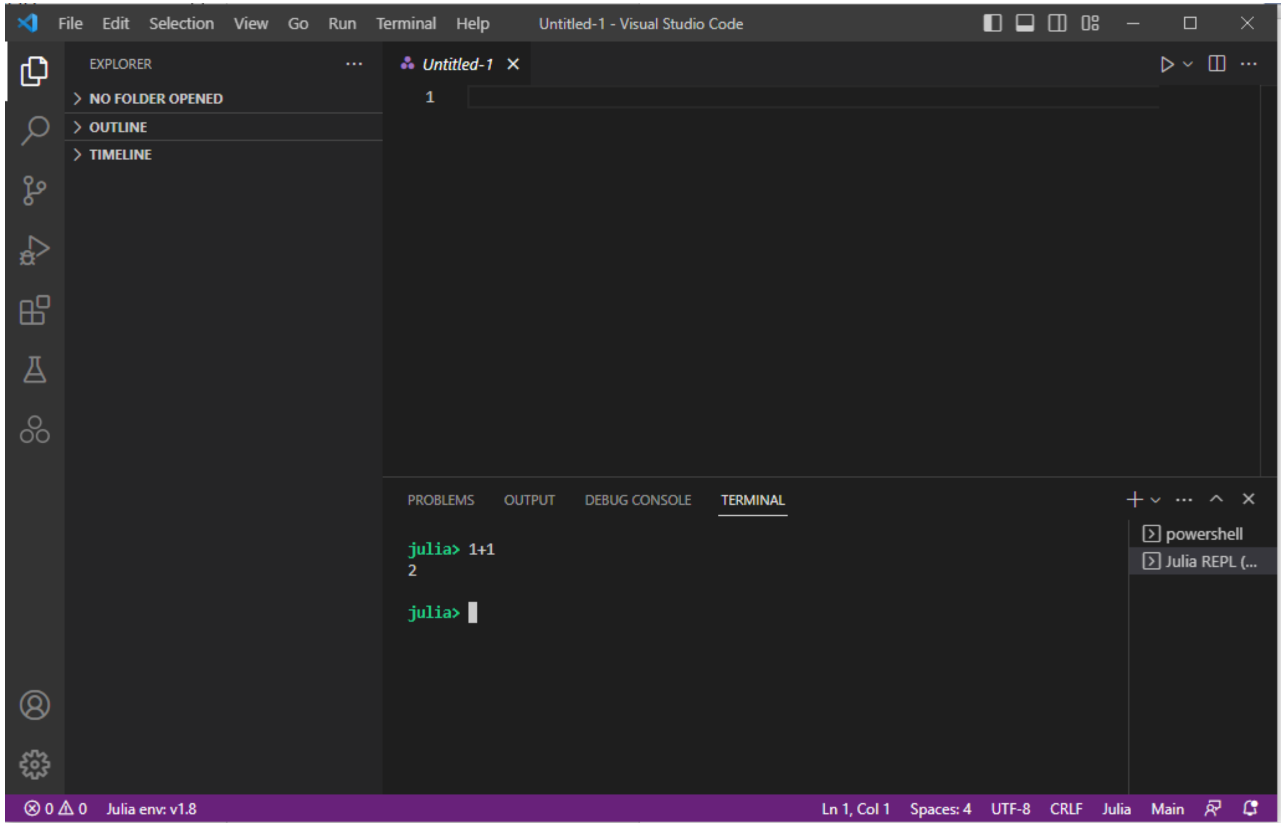
[Fig.10]

Packages installation (on VS Code)

Julia extension

For more information such as a 'start-up' guide for Julia in VSCode, go to: https://code.visualstudio.com/docs/languages/julia

To start Julia in visual code, press **'Ctrl+Shift+P'** (it opens the Command Palette) and type he command: **'Start Julia'.** After, you can start coding with Julia (e.g. 1 + 1 = 2):



This installed extension provides a Julia REPL (console) inside VS Code. Start the terminal at the beginning of every session by writing "Julia: Start REPL" in the command palette.

[Fig.11]

Packages installation (on VS Code)

<u>Other packages (‘Package manager’)</u>

The package manager lets you install, update and remove packages. To enter the ‘**package manager**’ press ‘**]**’ (you will see that the colored text ‘**julia>**’ text changes to **‘(@v1.8) pkg>**’.

**(You can go back by just pressing the backspace button)*

Activate the environment by writing ‘activate env’ on the package manager (you can see it is activated as the name ‘**(@v1.8) pkg**>’ will change to **‘(env) pkg>**’). This creates a new environment/folder called env:

```
(@v1.8) pkg> activate env
  Activating project at `C:\Users\[redacted]\env`

(env) pkg>
```

[Fig.12]

Packages installation (on VS Code)

Other packages ('Package manager')

The packages required to run 'OptiPlant' are: **JuMP** (allows writing optimization problems in Julia), **HiGHS** or **Gurobi** (LP solvers), **DataFrames** (enables working with structured data), **CSV** (reads CSV files) and **XLSX** (reads excel files in .xlsx format).

```
(env) pkg> add JuMP
    Updating registry at `C:\Users\[redacted]\.julia\registries\General.toml`
   Resolving package versions...
   Installed IrrationalConstants — v0.2.2
   Installed DiffResults ———————— v1.1.0
   Installed DiffRules —————————— v1.15.1
   Installed MutableArithmetics —— v1.3.0
   Installed JSON ——————————————— v0.21.4
   Installed SpecialFunctions ——— v2.2.0
```

```
(env) pkg> status
Status `C:\Users\[redacted]\env\Project.toml`
  [336ed68f] CSV v0.10.11
  [a93c6f00] DataFrames v1.5.0
  [87dc4568] HiGHS v1.5.2
  [4076af6c] JuMP v1.11.1
  [fdbf4ff8] XLSX v0.9.0
```

[Fig.13]

The installation of these packages is quite easy: After <u>activating the environment (env)</u>, write the following command in the terminal: '**add *****' (***= name of the package) and press enter. The installation of the packages will start automatically, and they will be saved inside the 'env' folder.

The check that all packages have been correctly installed, write the command '**status**' in the terminal -inside the env- and press enter.

*Some other extra packages that would enable the plotting and visualization of the results are: **Plots**, **StatsPlots** or **PrettyTables**. They can be installed the same way as described above.*

*Packages installation (Gurobi)**

***OPTIONAL**

Although **HiGHS is the recommended solver** to be used with OptiPlant -as it is open-source-, **Gurobi** is a faster -but commercial- alternative. Only one solver is required to run the OptiPlant tool, and both provide the same results.

To install and use **Gurobi**, one needs to get a using license first. To get that, go to https://www.gurobi.com/ in 'Downloads & Licenses' and register yourself as a user. Once you are registered, get the license that better suits you.

❗After generating the license, you will get a ***grbgetkey***, it is important you copy this safely, as it is crucial during the installation of Gurobi

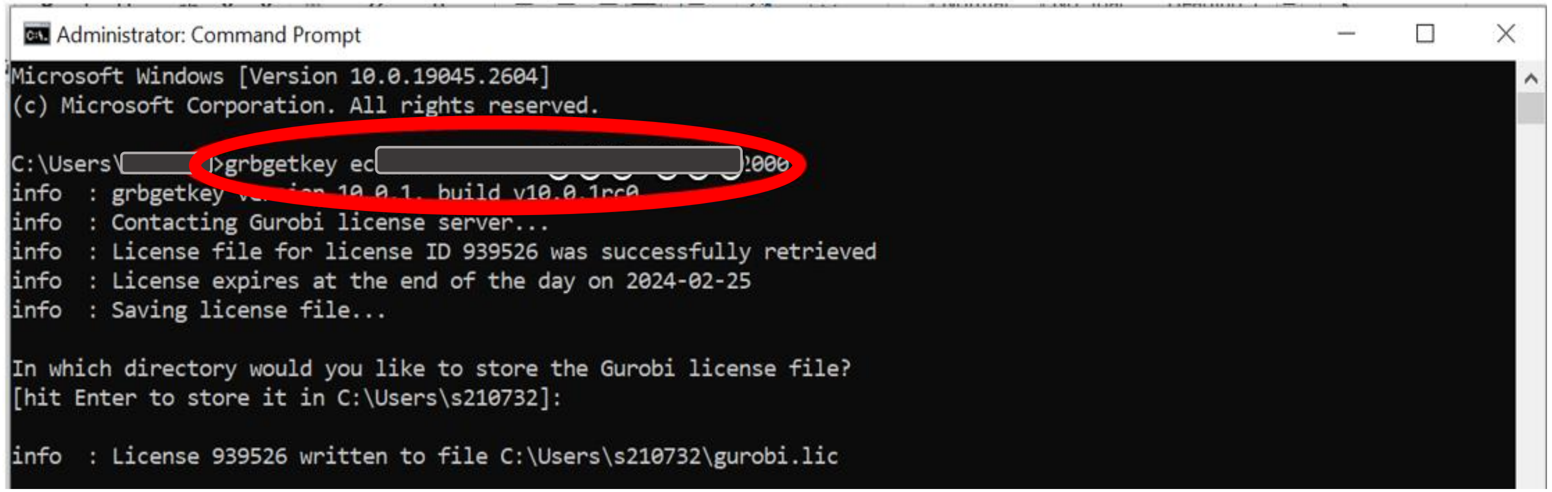
Afterwards, go to https://www.gurobi.com/downloads/gurobi-optimizer-eula/ and install the latest version of Gurobi optimizer.

Packages installation (Gurobi)*

***OPTIONAL**

Open the Gurobi installer and follow the instructions. Once Gurobi is installed restart your system manually (if it is not done automatically after its installation).

Next, open the ‘Command Prompt’ of your system –write “cmd” in the search menu of your PC- and insert the ***grbgetkey*** you have previously saved “*grbgetkey *n*u*m*b*e*r**”, as shown below:



[Fig.14]

Save the license in the default location suggested by the ’cmd’.

Finally, add the Gurobi package in your Julia code by writing : ‘**add Gurobi**’ (as any of the other mentioned packages).

OptiPlant: files overview

OptiPlant tool: File overview

Each of the folders comprising OptiPlant includes the following:

BASE: It includes two subfolders named ‘Data’ and ‘Results’. Includes all the elements that are not ’code-related’.

Data: Includes two subfolders named ‘Inputs’ and ‘Profiles’.

Inputs: This folder contains different excel sheets where one can check and modify the input data for different study-case scenarios such as: units conforming for the PtX plant, their techno-economic information, the operation strategy of the plant,etc. More details on the ‘Inputs excel sheets’ *-here-*.

Profiles: This folder contains the excel sheets where one can check and modify the wind/solar profiles and the electricity prices of different locations during different years. More details on the ‘Profiles excel sheets’ *-here-*.

Results: Has the results/outputs of the simulation. A new folder will be created any time a simulation is run, and its name would correspond to the one written in the ‘Inputs excel sheet’. Includes different subfolders: Data used, Hourly results and Main results. More details on how to process and interpret the results *-here-*.

OptiPlant tool: File overview

RUN CODE: It includes three Julia scripts named ImportData.jl, ImportScenarios.jl, and Main.jl.

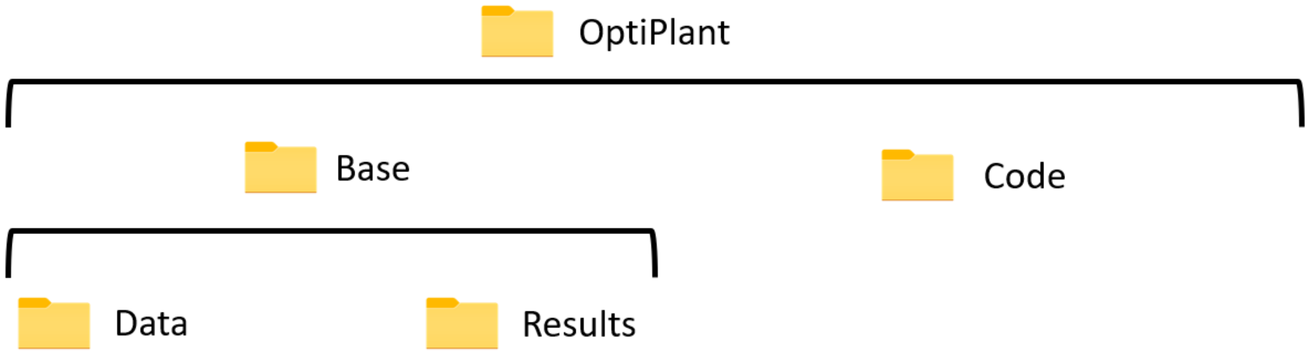
- *ImportData.jl:* Imports into Julia the necessary input data to run the simulation such as: PtX plant units, their techno-economic characteristics, power profiles. . .

- *ImportScenarios.jl*: Imports into Julia the necessary information regarding the scenario in which the plant operates in the study.

- ***Main.jl***: This is the optimization model *per se*. Uses the data imported from ImportData.jl, ImportScenarios.jl and extracts some results/outputs of the optimization model. More details on how to run the Main.jl script in section C.

OptiPlant tool: File overview

Once the **OptiPlant-master** ZIP file has been downloaded *-as shown in slide 3-*, one will find the following folders inside:



[Fig.15]

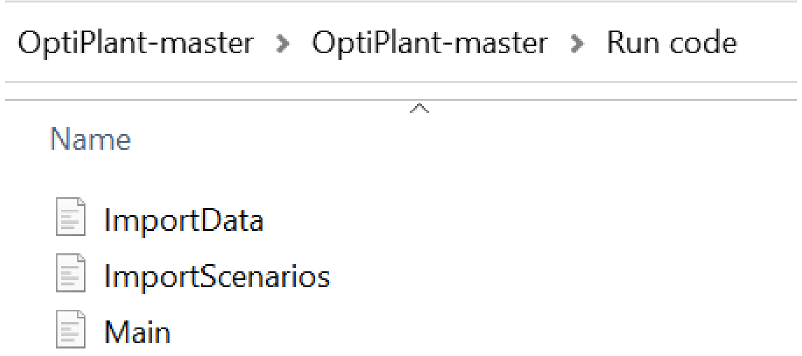
Note: The downloaded OptiPlant-master ZIP may also include other files like the README.md, the installation and user guides, etc. These elements are not part of the OptiPlant tool *per se,* that's why they are not included in the figure above.

OptiPlant files overview

RUN CODE folder

RUN CODE folder

The ‘**CODE**’ folder is one of the two main folders in OptiPlant tool -together with the ’**BASE**’ folder-. This folder is simpler than the ’**BASE**’ one as it only includes 3 Julia code files inside:



[Fig.35]

The main purpose of each of the Julia scripts has been previously described (sl.9). In most cases, one will only need to modify and run the ‘***Main.jl***’ file, as the other files act just as a bridge to import the date form the ‘Input’ excel files.

RUN CODE folder

To run the OptiPlant model under standard operation, one should only check/edit two parameters in the ‘Main.jl’ code:

- The solver being used *(line 4)*: Either “HiGHS” or “Gurobi” (you can customize the code to use your own solver)

```
3    #Choose a solver
4    Solver = "HiGHS" #Write "Gurobi" or "HiGHS".
```

[Fig.36]

- The directories in which the OptiPlant tool is stored in the PC and the folders from which the input data should be taken from *(lines 22-25)*:

```
16   #---------------------------Problem set up-----------------------------------
17   #Project name
18   Project = "Base"
19   # Folder name for all csv file
20   all_csv_files = "All_results"
21   # Folder paths for data acquisition and writing
22   Main_folder = "C:/Users/njbca/Documents/Models/OptiPlant-World - Copy";
23   Profiles_folder = joinpath(Main_folder,Project,"Data","Profiles") ;
24   Inputs_folder = joinpath(Main_folder,Project,"Data","Inputs") ;
25   Inputs_file = "Data_ammonia_paper"
26
27   # Scenario set (same name as exceel sheet)
28   Scenarios_set = "ScenariosToRun" ; include("ImportScenarios.jl")
```

*Folder name inside the ‘Results’ where the outputs will be saved

*OptiPlant directory

*Input data-excel file name

*Input data-excel sheet name

[Fig.37]

OptiPlant files overview

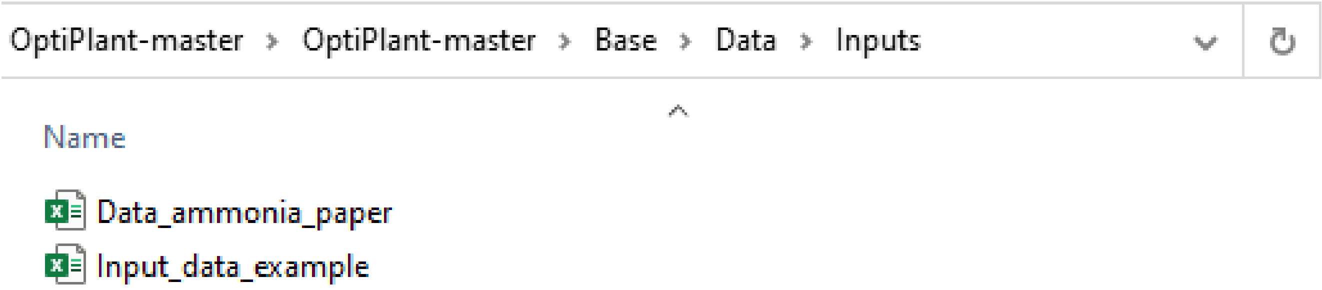
BASE folder: Data

BASE folder: Data

One of the main folders of the OptiPlant tool is the ‘**Data**’ folder. As it can be seen in *[Fig.3]*, it is contained in the mother folder ‘**BASE**’. At the same time, it includes the subfolders ’<u>Inputs</u>’ and ’<u>Profiles</u>’

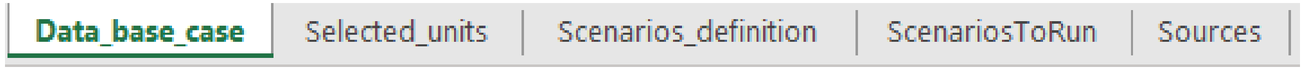
‘Inputs’ subfolder:

The ‘Inputs’ folder of the OptiPlant tool can be found at **BASE > Data**. Inside the folder, it looks like something like this:



[Fig.16]

If we enter one of these files, for instance *Input_data_example*, we find an excel document that has the following sheets:



[Fig.17]

BASE folder: Data

The purpose for each of the different sheets in any **'Data'** excel file *[Fig.5]* is described below:

Data_base_case: This sheet includes a list of the different units that can constitute the PtX plant and their characteristics: production rates, heat and electrical flows, load ranges, ramp up/down times, CapEx and OpEx, etc…

	A	B	C	D	E	H	I	J	K	L	M	N	O	P
1	Red: Don't change the name without changing the Julia code					Yearly demand (kg fuel	Produced from	El balance	Heat balance	H2 balance	Max Capacity	uction rate (kg output	Heat generated (kWh/output)	Load min (% of max capacity)
2					Parameters-->	Yearly demand (kg fuel	Produced from	El balance	Heat balance	H2 balance	Max Capacity	Fuel production rate	Heat generated (kWh/output	Load min (% of max capacity)
3		Subsets	Subsets_2	Type of units	Year-->	All	All	All	All	All	All	2025	2025	2025
5					Line/Column index	3	4	5	6	7	8	9	10	11
6	Non-electrical	Product	NoSubset	CO2 capture DAC	1	0	Product/Reactant1	0	0	0	40000	1,37	0	0%
7		Product	NoSubset	CO2 capture PS	2	0	Product/Reactant1	0	0	0	40000	1,37	0	0%
8		Product/Reactant1	Min_demand_MainFuel	MeOH plant CCU with AEC	3	397989950	Reactant2	0	1	0	20000	5,26	0,68	40%
9		Product	Min_demand_MainFuel	NH3 plant + ASU with AEC	4	425806452	Reactant6	0	1	0	20000	5,56	0	40%
10		Product	Min_demand_MainFuel	NH3 plant + ASU with SOEC	5	425806452	Reactant6	0	1	0	20000	5,56	0	40%
11		Product	Min_demand_MainFuel	H2 client	6	66000000	Reactant7	0	1	0	20000	1	0	0%
12		Reactant8	NoSubset	Water supply (desalination plant)	7	0	-	0	0	0	400000	0	0	0%
13		Product/Reactant3	NoSubset	Electrolysers AEC	8	0	Reactant8	0	1	1	20000	0,088888889	7,07	0%
14		Product/Reactant3	NoSubset	Electrolysers SOEC heat integrated	9	0	Reactant8	0	1	1	20000	0,088888889	0	0%
15		Reactant2	NoSubset	H2 pipeline to MeOH CCU plant	10	0	-	0	0	-1	20000	0	0	0%
16		Reactant6	NoSubset	H2 pipeline to NH3 plant	11	0	-	0	0	-1	20000	0	0	0%
17		Reactant7	NoSubset	H2 pipeline to client	12	0	-	0	0	-1	20000	0	0	0%
18		Heat_in	NoSubset	Heat from district heating	13	0	-	0	1	0	2000000	0	0	0%
19		Heat_out	Heat_sell	Heat sent to district heating	14	0	-	0	-1	0	2000000	0	1	0%
20		Product	O2_sell	Sale of oxygen	15	0	Product/Reactant3	0	0	0	180000	8	0	0%
21		Stor_in	NoSubset	H2 storage compressor	16	0	-	0	0	-1	20000	0	0	0%
22		Stor_out	NoSubset	H2 storage valve	17	0	-	0	0	1	20000	0	0	0%
23		Tank	NoSubset	H2 storage (buried pipes)	18	0	-	0	0	0	20000000	0	0	9%
24	Electrical	RPU_Solar_fixed	NoSubset	Solar fixed	19	0	-	1	0	0	2000000	0	0	0%
25		RPU_Solar_track	NoSubset	Solar tracking	20	0	-	1	0	0	2000000	0	0	0%
26		RPU_ON_SP198-HH100	NoSubset	ON_SP198-HH100	21	0	-	1	0	0	2000000	0	0	0%
27		RPU_ON_SP198-HH150	NoSubset	ON_SP198-HH150	22	0	-	1	0	0	2000000	0	0	0%
28		RPU_ON_SP237-HH100	NoSubset	ON_SP237-HH100	23	0	-	1	0	0	2000000	0	0	0%
29		RPU_ON_SP237-HH150	NoSubset	ON_SP237-HH150	24	0	-	1	0	0	2000000	0	0	0%
30		RPU_ON_SP277-HH100	NoSubset	ON_SP277-HH100	25	0	-	1	0	0	2000000	0	0	0%
31		RPU_ON_SP277-HH150	NoSubset	ON_SP277-HH150	26	0	-	1	0	0	2000000	0	0	0%
32		RPU_ON_SP321-HH100	NoSubset	ON_SP321-HH100	27	0	-	1	0	0	2000000	0	0	0%
33		RPU_ON_SP321-HH150	NoSubset	ON_SP321-HH150	28	0	-	1	0	0	2000000	0	0	0%
34		RPU_OFF_SP379-HH100	NoSubset	OFF_SP379-HH100	29	0	-	1	0	0	2000000	0	0	0%
35		RPU_OFF_SP379-HH150	NoSubset	OFF_SP379-HH150	30	0	-	1	0	0	2000000	0	0	0%
36		RPU_OFF_SP450-HH100	NoSubset	OFF_SP450-HH100	31	0	-	1	0	0	2000000	0	0	0%
37		RPU_OFF_SP450-HH150	NoSubset	OFF_SP450-HH150	32	0	-	1	0	0	2000000	0	0	0%
38		PU_Grid_in	Grid_buy	Electricity from the grid	33	0	-	1	0	0	2000000	0	0	0%
39		Grid_out	NoSubset	Curtailment	34	0	-	-1	0	0	24000	0	0	0%
40		Stor_in	NoSubset	Charge batteries	35	0	-	-1	0	0	10000000	0	0	0%
41		Stor_out	NoSubset	Discharge batteries	36	0	-	1	0	0	60000000	0	0	0%
42		Tank	NoSubset	Batteries	37	0	-	0	0	0	20000000	0	0	10%

…

[Fig.18]*

BASE folder: Data

***Note on *[Fig.18]*:** A lot of information is displayed in the Data_base_case excel sheet and can be overwhelming at first. However, it is quite simple to read once you start getting used to it, every unit parameter has their units and source stated. One would not usually change anything on this sheet.

The red box in -*Fig.18*- shows the default yearly demands of each fuel. As previously mentioned, these values are the main drivers of the model, meaning that the model would minimize the fuel production costs of the PtX plant as long as the yearly fuel demand is fulfilled.

! It is important to note that the different units are divided in ’non-electrical’ and ’electrical’.

BASE folder: Data

Selected_units: This sheet contains a list of the different units and technologies that can constitute the PtX plant and the ones that are used for each fuel production process -i.e. NH3, H2, MeOH, etc.- For each case, a 1 implies that the unit is considered in the PtX plant and a 0 implies that it is not.

	A	B	C	D	E	F	G	H	I	J	K
1		Fuel produced	NH3	NH3	MeOH	MeOH	MeOH	MeOH	H2	H2	H2
2		Fuel energy content LHV (MJ/kg fue	18,6	18,6	19,9	19,9	19,9	19,9	120	120	120
3		Electrolyser	AEC	SOEC	AEC	SOEC	AEC	SOEC	AEC	SOEC	Mix
4		Carbon capture	None	None	DAC	DAC	PS	PS	None	None	None
5		Configuration	NH3_AEC_None	NH3_SOEC_None	MeOH_AEC_DAC	MeOH_SOEC_DAC	MeOH_AEC_PS	MeOH_SOEC_PS	H2_AEC_None	H2_SOEC_None	H2_Mix_None
6	1	CO2 capture DAC	0	0	1	1	0	0	0	0	0
7	2	CO2 capture PS	0	0	0	0	1	1	0	0	0
8	3	MeOH plant CCU with AEC	0	0	1	1	1	1	0	0	0
9	4	NH3 plant + ASU with AEC	1	0	0	0	0	0	0	0	0
10	5	NH3 plant + ASU with SOEC	0	1	0	0	0	0	0	0	0
11	6	H2 client	0	0	0	0	0	0	1	1	1
12	7	Water supply (desalination plant)	1	1	1	1	1	1	1	1	1
13	8	Electrolysers AEC	1	0	1	0	1	0	1	0	0
14	9	Electrolysers SOEC heat integrated	0	1	0	0	0	0	0	0	0
15	10	H2 pipeline to MeOH CCU plant	0	0	1	1	1	1	0	0	0
16	11	H2 pipeline to NH3 plant	1	1	0	0	0	0	0	0	0
17	12	H2 pipeline to client	0	0	0	0	0	0	1	1	1
18	13	Heat from district heating	1	1	1	1	1	1	1	1	1
19	14	Heat sent to district heating	1	1	1	1	1	1	1	1	1
20	15	Sale of oxygen	1	1	1	1	1	1	1	1	1
21	16	H2 storage compressor	1	1	1	1	1	1	1	1	1
22	17	H2 storage valve	1	1	1	1	1	1	1	1	1
23	18	H2 storage (buried pipes)	1	1	1	1	1	1	1	1	1
24	19	Solar fixed	1	1	1	1	1	1	1	1	1
25	20	Solar tracking	1	1	1	1	1	1	1	1	1
26	21	ON_SP198-HH100	1	1	1	1	1	1	1	1	1
27	22	ON_SP198-HH150	1	1	1	1	1	1	1	1	1
28	23	ON_SP237-HH100	1	1	1	1	1	1	1	1	1
29	24	ON_SP237-HH150	1	1	1	1	1	1	1	1	1
30	25	ON_SP277-HH100	1	1	1	1	1	1	1	1	1
31	26	ON_SP277-HH150	1	1	1	1	1	1	1	1	1
32	27	ON_SP321-HH100	1	1	1	1	1	1	1	1	1
33	28	ON_SP321-HH150	1	1	1	1	1	1	1	1	1
34	29	OFF_SP379-HH100	1	1	1	1	1	1	1	1	1
35	30	OFF_SP379-HH150	1	1	1	1	1	1	1	1	1
36	31	OFF_SP450-HH100	1	1	1	1	1	1	1	1	1
37	32	OFF_SP450-HH150	1	1	1	1	1	1	1	1	1
38	33	Electricity from the grid	1	1	1	1	1	1	1	1	1
39	34	Curtailment	1	1	1	1	1	1	1	1	1
40	35	Charge batteries	1	1	1	1	1	1	1	1	1
41	36	Discharge batteries	1	1	1	1	1	1	1	1	1
42	37	Batteries	1	1	1	1	1	1	1	1	1

...

[Fig.19] *

BASE folder: Data

***Note on *[Fig.19]*:** One can change the 1/0 values according to their preferences. However, it is important to be aware that the default values are the ’standard case’ ones.

❗ Note down which values you change or work on a copy-file in order to be able to go back to the ’standard case’ settings.

BASE folder: Data

Scenarios_definition: This sheet is used to define different scenarios considering factors such as the operation strategy of the plant, different operating constrains/conditions of the units,etc. It acts between a 'switch' between the Data_base_case, the Selected_units and the output of the model.

	A	B	C	D	E	F	G	H	I	J	K
1	Changes also include the changes made in the reference scenario. If nothing is specified in the reference scenario column, changes are made compared to the input data sheet (i.e Data_base_case)										
2	Change are made compared to the reference scenario. "Chains" of reference scenarios doesn't work.										
3											
4	Reference scenario	Scenario name definitior	Type of units for change	Parameter changed	Year new value	New value	Year old value	Old value			
5		Semi-islanded	Electricity from the grid	Used (1 or 0)	All	1	All	1			
6	Semi-islanded	Semi-islandedflex	MeOH plant CCU with AE(	Load min (% of max capacity	All	0	2025	0,4			
7		Semi-islandedflex	NH3 plant + ASU with AEC	Load min (% of max capacity	All	0	2025	0,4			
8		Islanded	Electricity from the grid	Used (1 or 0)	All	0	All	1			
9	Islanded	Is_nonflex	MeOH plant CCU with AE(	Load min (% of max capacity	All	1	2025	0,4			
10		Is_nonflex	NH3 plant + ASU with AEC	Load min (% of max capacity	All	1	2025	0,4			
11		SI_nonflex	MeOH plant CCU with AE(	Load min (% of max capacity	All	1	2025	0,4			
12		SI_nonflex	NH3 plant + ASU with AEC	Load min (% of max capacity	All	1	2025	0,4			
13	Islanded	Is_flex	MeOH plant CCU with AE(	Load min (% of max capacity	All	0	2025	0,4			
14		Is_flex	NH3 plant + ASU with AEC	Load min (% of max capacity	All	0	2025	0,4			

...

[Fig.20] *

BASE folder: Data

***Note on *[Fig.20]*:** The logic of these cells is situated in between the inputs (1/0) made in the Selected_units sheet and the final output. So all the changes made in this sheet are made compared to the input data sheet (i.e Data_base_case and Selected_units).

This sheet is especially useful when one wants to do **sensitivity analysis.**

BASE folder: Data

ScenariosToRun: These sheet is used to list the different scenarios to be run through the optimization model. The conditions and characteristics of each of the listed scenarios make reference to the other sheets in the same excel document. One can set the scenario parameters such as: operating strategy, location wind/solar data, year data, produced fuel, electrolyzer technology, etc. The output results of the model are going to be stored as CSV files in **Results > Results folder name** (the folder is automatically created)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		Options available																
2		Name of the scenario in the output csv file	Scenarios_defini	Esbjerg	MeOH	PS	2025	Folder name where the excel profile is	Name of the excel file profile	AEC	ıy data sheet nan	Any name						
3				Ceduna	NH3	DAC				SOEC								
4				Arica		None												
5																		Results to write
6																		
7	Scenario number	Scenario name	Scenario	Location	Fuel	CO2 capture	Year data	Profile folder name	Profile name	Electrolyser	Input data sheet	Result folder name	Max capacity	Ramping	No negative elec price	Fixed heat sale	Fixed oxygen sale	Flows
8	1	Behind-the-meter	**Islanded**	Esbjerg	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
9	2	Behind-the-meter	**Islanded**	Ceduna	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
10	3	Behind-the-meter	**Islanded**	Arica	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
11	4	Base_case	**Semi-islanded**	Esbjerg	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
12	5	Base_case	**Semi-islanded**	Ceduna	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
13	6	Base_case	**Semi-islanded**	Arica	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
14	7	Base_case	**Semi-islanded**	Esbjerg	**NH3**	**None**	**2025**	**All_locations**	**2019**	**SOEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
15	8	Base_case	**Semi-islanded**	Ceduna	**NH3**	**None**	**2025**	**All_locations**	**2019**	**SOEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
16	9	Base_case	**Semi-islanded**	Arica	**NH3**	**None**	**2025**	**All_locations**	**2019**	**SOEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
17	10	Base_case	**Semi-islanded**	Esbjerg	**MeOH**	**PS**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
18	11	Base_case	**Semi-islanded**	Ceduna	**MeOH**	**PS**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
19	12	Base_case	**Semi-islanded**	Arica	**MeOH**	**PS**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_base_case	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
20	13	Bhm-nonflexible	**Is_nonflex**	Esbjerg	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
21	14	Bhm-nonflexible	**Is_nonflex**	Ceduna	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
22	15	Bhm-nonflexible	**Is_nonflex**	Arica	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
23	16	Semi-islanded_nonflex	**SI_nonflex**	Esbjerg	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
24	17	Semi-islanded_nonflex	**SI_nonflex**	Ceduna	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
25	18	Semi-islanded_nonflex	**SI_nonflex**	Arica	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
26	19	Bhm-superflexible	**Is_flex**	Esbjerg	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
27	20	Bhm-superflexible	**Is_flex**	Ceduna	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
28	21	Bhm-superflexible	**Is_flex**	Arica	**NH3**	**None**	**2025**	**All_locations**	**2019**	**AEC**	Data_base_case	Results_sensitivitie	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE

...

[Fig.21] *

BASE folder: Data

***Note on [Fig.21]**: Be especially cautious on writing the name for the different cell's inputs correctly in this sheet, as these 'names' need to call the other sheets in the same excel document.

It is also possible to create a new sheet for each study case where you define the scenarios to run through the model. If you do that, though, be aware that the name of this excel sheet will be called in the code Main.jl (line 28), so be sure it is correctly called/changed in this code line too.

BASE folder: Data

Sources: These sheets include the references and sources of the data used in the sheet Data_base_case. In principle, one would not need to edit the content of these sheets when running optimization of scenarios.

Good practice is that if a user changes/adds some input data displayed in the Data_base_case sheet, he/she also updates the source where this information was taken from in this sheet.

	A	B	C	D	E	F	G
1	Reference tag in the comments and mendeley	Title	Main author	Year	Document type	F	Link (doi is preferred)
2	MMZCS2023	Communication with Mærsk Mc-Kinney Møller Center for Zero Carbon Shipping	Mærsk center	2023	Database		
3	BhmWP12023	Assessment of resources available in Bornholm	WP1	2023	Report		
4	Adams2019	Onboard Type IV Compressed Hydrogen Storage System - Cost and Performance Status (19008)	Adams	2019	Record 19008 NREL		https://www.hydrogen.energy.gov/program_r
5	Papadias2021	Bulk storage of hydrogen	Papadias	2021	Article		https://www.sciencedirect.com/science/article
6	Armijo2020	Flexible production of green hydrogen and ammonia from variable solar and wind energy:Case study of Chile and Argentina	Armijo	2020	Article		https://doi.org/10.1016/j.ijhydene.2019.11.028
7	IEA2019	IEA The future of hydrogen 2019	IEA	2019	IEA Report		https://webstore.iea.org/the-future-of-hydrog
8	Campion2023	Techno-economic assessment of green ammonia production with different wind and solar potentials	Campion	2023	Article		https://doi.org/10.1016/j.rser.2022.113057
9	Induspart2023	Communication with industrial partners	Induspart	2023	Database		
10	NREL2020	Annual technlogy baseline for hydrogen	NREL	2020	Webpage		https://atb.nrel.gov/transportation/2020/hydr
11	Ikäheimo2018	Power-to-ammonia in future North European 100 % renewable power and heat system	Ikäheimo	2018	Paper		https://www.sciencedirect.com/science/article
12	DEAstor2020	Technology data for energy storage	DEA	2020	Report/Catalogue		https://ens.dk/en/our-services/projections-an
13	Campion2021	1arE-fuel: LCOE and optimal electricity supply strategies for P2X plant	Campion	2021	Report		https://backend.orbit.dtu.dk/ws/portalfiles/pc

...

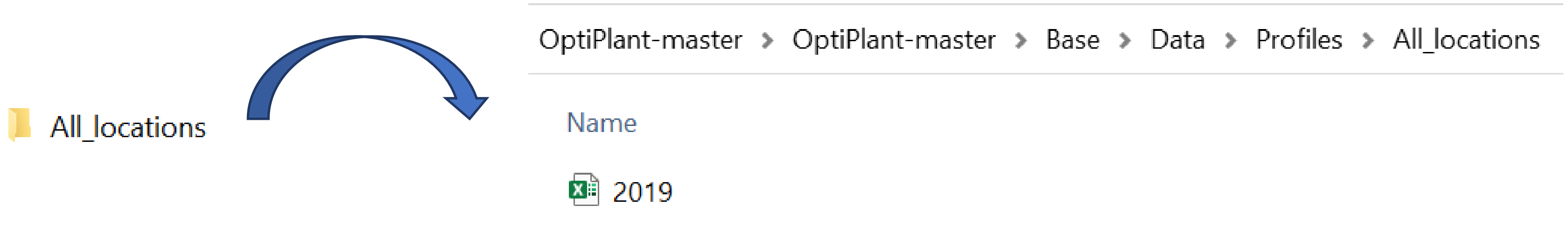
[Fig.22]

BASE folder: Data

As previously mentioned, the '**Data**' folder is one of the most important elements of the OptiPlant tool. As it can be seen in *[Fig.3]*, it is contained in the mother folder '**BASE**'. It contains the subfolders '<u>Inputs</u>' and '<u>Profiles</u>'

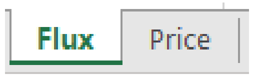
'Profiles' subfolder:

The 'Profiles' folder of the OptiPlant tool can be found at **BASE > Data**. Inside this folder, it looks like something like this:



[Fig.23]

If we enter the *2019* file, we find an excel document that has only two sheets:



[Fig.24]

BASE folder: Data

The purpose for each of the different sheets in any **'Data>Profiles'** excel file *[Fig.12]* is described below:

Flux: This sheet contains the hourly solar and wind profiles for the year '2019' using different solar and wind technologies in different locations.

	A	B	C	D	E	F	G	H
1		**Solar fixed**	**Solar tracking**	**SP198-HH100**	**SP198-HH150**	**SP237-HH100**	**SP237-HH150**	**SP277-HH100**
2	**Locations**	Esbjerg	Esbjerg	Esbjerg	Esbjerg	Esbjerg	Esbjerg	Esbjerg
3	**Subsets**	RPU_Solar_fixed	RPU_Solar_track	RPU_ON_SP198-HH100	RPU_ON_SP198-HH150	RPU_ON_SP237-HH100	RPU_ON_SP237-HH150	RPU_ON_SP277-HH100
4	**Index**	1	2	3	4	5	6	7
5		0	0	0.948222675	0.94950456	0.949869324	0.949648231	0.949991927
6		0	0	0.94843455	0.949741218	0.949762282	0.949671095	0.95
7		0	0	0.948977738	0.94987903	0.949653408	0.94968632	0.95
8		0	0	0.949029693	0.949825081	0.949648633	0.94968036	0.95
9		0	0	0.94907414	0.949784875	0.949644548	0.949675918	0.95
10		0	0	0.949958807	0.95	0.949870529	0.95	0.949905566
11		0	0	0.94998084	0.95	0.949939781	0.95	0.949956077
12		0	0	0.949776904	0.949976121	0.949675037	0.949924948	0.949906222
13		0.013	0.017	0.949673593	0.949940522	0.949664507	0.949813059	0.949957034
14		0.075	0.082	0.949680356	0.949926569	0.949665158	0.949769204	0.949956305
15		0.101	0.105	0.949803499	0.94997502	0.949677975	0.949921486	0.949877659
16		0.074	0.079	0.949967631	0.95	0.949898262	0.95	0.949925794
17		0.084	0.088	0.949689098	0.95	0.949666	0.95	0.949955363
18		0.047	0.05	0.95	0.95	0.95	0.95	0.95
19		0.01	0.014	0.95	0.95	0.95	0.95	0.95
20		0	0	0.95	0.95	0.95	0.95	0.95
21		0	0	0.95	0.95	0.95	0.95	0.95
22		0	0	0.95	0.95	0.95	0.95	0.95
23		0	0	0.95	0.95	0.95	0.95	0.95
24		0	0	0.949806416	0.95	0.949678298	0.95	0.949874526
25		0	0	0.949750737	0.94999576	0.949672146	0.949986673	0.949934325
26		0	0	0.949372402	0.949919418	0.949635506	0.949746728	0.949989483
27		0	0	0.948417781	0.949445297	0.949768707	0.949642524	0.95
28		0	0	0.948407626	0.949100899	0.949907888	0.949642088	0.949957773

...

[Fig.25] *

BASE folder: Data

Note on** ***[Fig.13]: Take into account that the numbers in the 'Flux' excel sheet represent the <u>normalized output power</u> for each generator.

The solar/wind power profiles included in the Flux excel sheet are extracted from the **CorRES tool** (for wind profiles), and from **renewable.ninja** website (for solar profiles).

BASE folder: Data

Price: This sheet contains the hourly electricity grid buy price for the year '2019' at different locations.

	A	B	C	D	E
1					
2		Grid buy price €/kWh	Grid buy price €/kWh	Grid buy price €/kWh	
3	Locations	Esbjerg	Ceduna	Arica	
4	Subsets	Grid_buy	Grid_buy	Grid_buy	
5	Index	1	2	3	
6		0.02832	0.04800877	0.054261	
7		0.01007	0.04733007	0.054261	
8		-0.00408	0.04433145	0.054261	
9		-0.00991	0.03155955	0.054261	
10		-0.00741	0.03044278	0.054261	
11		-0.01255	0.0299862	0.054261	
12		-0.01725	0.02838817	0.054261	
13		-0.01507	0.03102893	0.050699	
14		-0.00493	0.03209634	0.046389	
15		-0.00633	0.03570579	0.045936	
16		-0.00493	0.04196834	0.045936	
17		0.00045	0.05205629	0.045936	
18		0.00012	0.05346922	0.045936	
19		-0.00002	0.0572576	0.045908	
20		0	0.0553449	0.018223	
21		-0.00003	0.06775277	0	
22		0.00197	0.05301881	0	

...

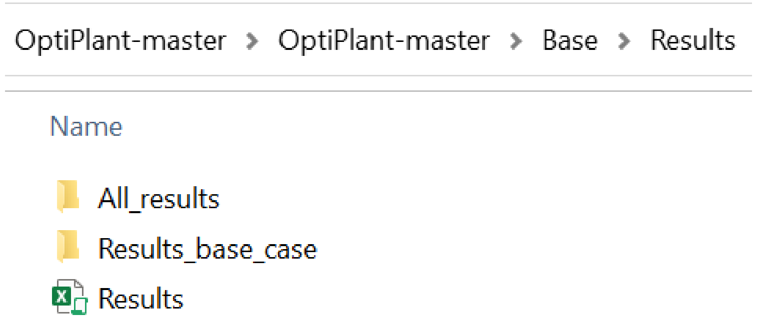
[Fig.26]

OptiPlant files overview

BASE folder: Results

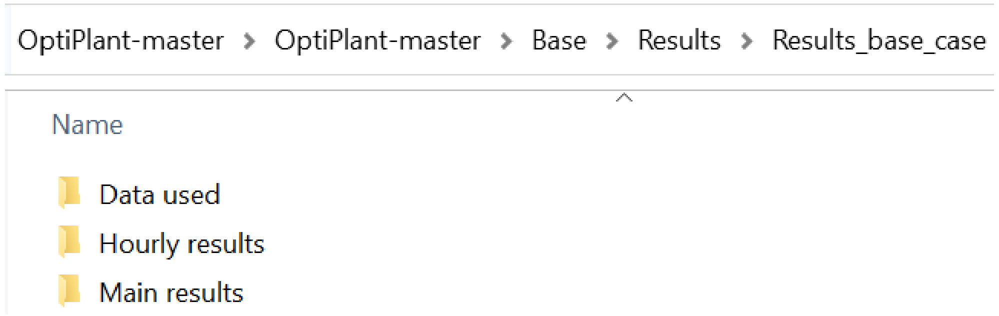
BASE folder: Results

One of the main folders of the OptiPlant tool is the '**Results**' folder. It is contained in the mother folder '**BASE**' and it has the following files and subfolders inside:



[Fig.27]

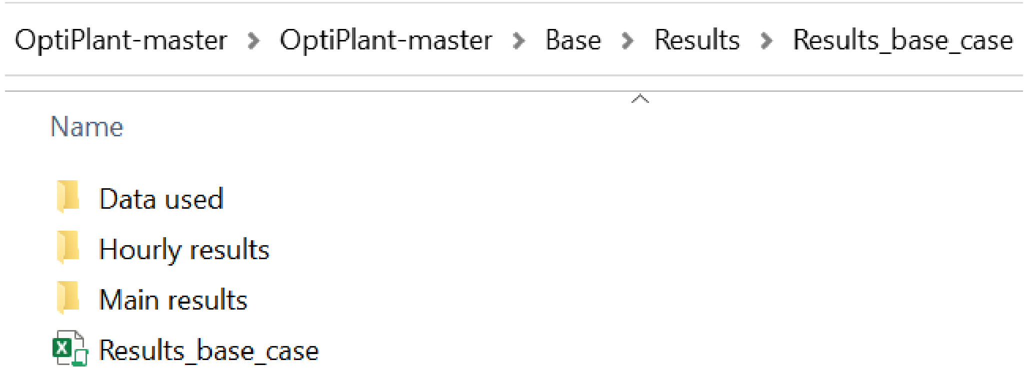
Each of the folders corresponds to one simulation run, and as stated before, its name is settled inside the 'ScenariosToRun' excel sheet from the '<u>Inputs</u>' file. Each of these folders -e.g. 'Results_base_case' - includes three subfolders named 'Data used', 'Hourly results', and 'Main results', that include the inputs and results of the simulation in CSV files, respectively:



[Fig.28]

BASE folder: Results

To process and read the CSV results included in the folders shown above, it is recommended to create a copy of the 'Results' excel file that can be found in the 'Results' folder -see [Fig.15]- and save it in the corresponding 'Results ***' subfolder, like shown below:



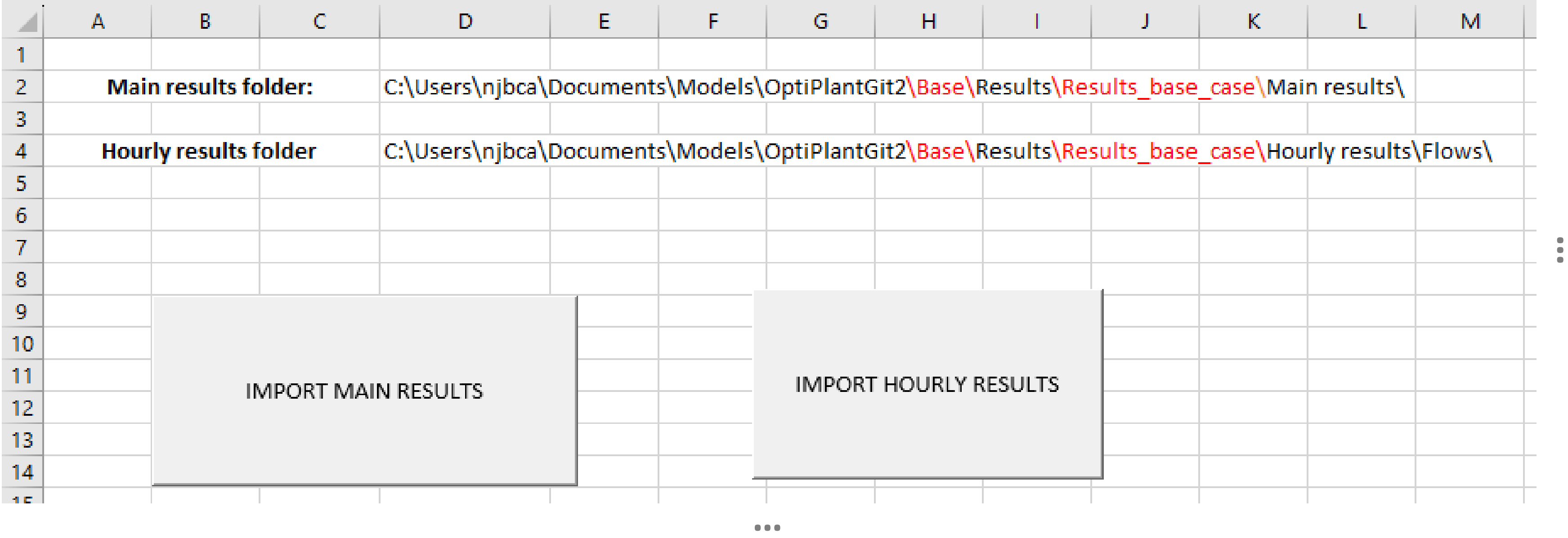
[Fig.29]

When opening any of the 'Results' excel file (or one of its copies) one would find a document with the following sheets:

[Fig.30]

BASE folder: Results

First off, to check the results of the corresponding simulations, open the corresponding '**Results**' excel file and import the data by going to the 'Import' excel sheet, writing the right CSV files directory and clicking on the macros:



[Fig.31]

Note: Make sure the directories for the 'Main results folder ' and the 'Hourly results folder ' are written correctly and end with '\'

BASE folder: Results

Once the results are imported, one would be able to see the outcomes and results for the different studied scenarios individually in the different excel sheets. Each scenario will be named as it was in the '<u>Inputs</u>' excel file.

For instance, in the **'All_scenarios'** sheet, one can see all the output values for each unit and scenario:

	A	B	C	D	E	F	G	H	I	J	K	L
1	Scenario	Type of unit	Year	Location	Profile	Fuel	Electrolyser	CO2 cap	Installed capacity (MW (elec	Total investment(MEuros	Annualised investment(MEuros	Fixed O&M(MEuros)
2	Behind-the-meter	NH3 plant + ASU with AEC	2025	Esbjerg	2019	NH3	AEC	None	61.52946253	1111.096573	98.32713036	44.44386292
3	Behind-the-meter	Water supply (desalination pla	2025	Esbjerg	2019	NH3	AEC	None	172.0882025	25.44795595	2.973073113	0.763438678
4	Behind-the-meter	Electrolysers AEC	2025	Esbjerg	2019	NH3	AEC	None	15.29672911	841.3201009	74.45310627	84.13201009
5	Behind-the-meter	H2 pipeline to NH3 plant	2025	Esbjerg	2019	NH3	AEC	None	11.07530325	0	0	0
6	Behind-the-meter	Heat from district heating	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
7	Behind-the-meter	Heat sent to district heating	2025	Esbjerg	2019	NH3	AEC	None	108.1478748	0	0	0
8	Behind-the-meter	Sale of oxygen	2025	Esbjerg	2019	NH3	AEC	None	122.3738329	0	0	0
9	Behind-the-meter	H2 storage compressor	2025	Esbjerg	2019	NH3	AEC	None	4.221425852	0	0	0
10	Behind-the-meter	H2 storage valve	2025	Esbjerg	2019	NH3	AEC	None	11.07530325	0	0	0
11	Behind-the-meter	H2 storage (buried pipes)	2025	Esbjerg	2019	NH3	AEC	None	591.9032131	147.9758033	12.0959651	4.439274098
12	Behind-the-meter	Solar fixed	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
13	Behind-the-meter	Solar tracking	2025	Esbjerg	2019	NH3	AEC	None	1021.252567	660.2702178	56.65334018	11.39498602
14	Behind-the-meter	ON_SP198-HH100	2025	Esbjerg	2019	NH3	AEC	None	566.8592264	996.924411	91.16683945	8.275611857
15	Behind-the-meter	ON_SP198-HH150	2025	Esbjerg	2019	NH3	AEC	None	470.9322911	1030.7735	94.26227418	6.875168774
16	Behind-the-meter	ON_SP237-HH100	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
17	Behind-the-meter	ON_SP237-HH150	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
18	Behind-the-meter	ON_SP277-HH100	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
19	Behind-the-meter	ON_SP277-HH150	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
20	Behind-the-meter	ON_SP321-HH100	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
21	Behind-the-meter	ON_SP321-HH150	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
22	Behind-the-meter	OFF_SP379-HH100	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
23	Behind-the-meter	OFF_SP379-HH150	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
24	Behind-the-meter	OFF_SP450-HH100	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
25	Behind-the-meter	OFF_SP450-HH150	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
26	Behind-the-meter	Electricity from the grid	2025	Esbjerg	2019	NH3	AEC	None	0	0	0	0
27	Behind-the-meter	Curtailment	2025	Esbjerg	2019	NH3	AEC	None	964.2952999	0	0	0
28	Behind-the-meter	Charge batteries	2025	Esbjerg	2019	NH3	AEC	None	90.21712897	0	0	0
29	Behind-the-meter	Discharge batteries	2025	Esbjerg	2019	NH3	AEC	None	100.2412544	0	0	0
30	Behind-the-meter	Batteries	2025	Esbjerg	2019	NH3	AEC	None	100.2412544	55.13268993	6.441127076	0.826990349

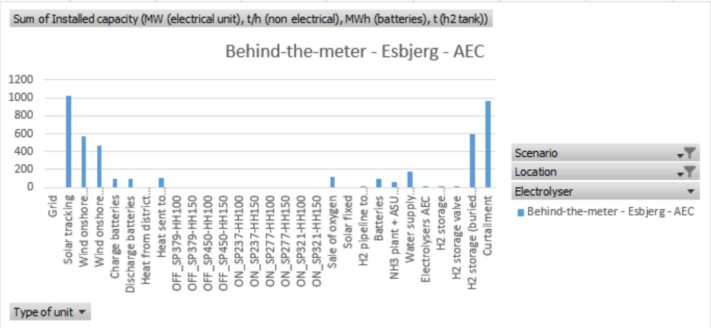
...

[Fig.32]

BASE folder: Results

In the other excel sheets such as **‘Elec production’, ‘Electricity consumption’, ‘Production cost’, ‘Cost breakdown’ and ‘Installed capacities’**, one can see a break down of the output values of each unit and scenario and plot simply the results by using ‘Pivot Tables’. For instance:

Sum of Installed capacity (MW (electrical unit), t/h (non electrical), MWh (batteries), t (h2 tank))	Column Labels	
	Behind-the-meter	Grand Total
	Esbjerg	
Row Labels	AEC	
Grid	0	0
Solar tracking	1021	1021
Wind onshore (SP198-HH100)	567	567
Wind onshore (SP198-HH150)	471	471
Charge batteries	90	90
Discharge batteries	100	100
Heat from district heating	0	0
Heat sent to district heating	108	108
OFF_SP379-HH100	0	0
OFF_SP379-HH150	0	0
OFF_SP450-HH100	0	0
OFF_SP450-HH150	0	0
ON_SP237-HH100	0	0
ON_SP237-HH150	0	0
ON_SP277-HH100	0	0
ON_SP277-HH150	0	0
ON_SP321-HH100	0	0
ON_SP321-HH150	0	0
Sale of oxygen	122	122
Solar fixed	0	0
H2 pipeline to NH3 plant	11	11
Batteries	100	100
NH3 plant + ASU with AEC	62	62
Water supply (desalination plant)	172	172
Electrolysers AEC	15	15
H2 storage compressor	4	4
H2 storage valve	11	11
H2 storage (buried pipes)	592	592
Curtailment	964	964
Grand Total	4411.750369	4411.750369



...

[Fig.33]

❗ Remember to refresh the pivot table every time you import new results

BASE folder: Results

Furthermore, one would see that some new excel sheets appear when importing the ‘hourly results’. Each of the new excel sheets corresponds to one of the run scenarios, and the hourly flows for different parameters is displayed. As an example:

	A	B	C	D	E	F	G	H	I	J
1	Informations	Time	NH3-AEC	H2O sea	H2_from_AEC	H2_pipeline_to_NH3_plant	Heat_import	Heat_export	O2	H2_to_pipe
2	Scenario: Behind-the-meter	1	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
3	Profile: 2019	2	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
4	Location: Esbjerg	3	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
5	Fuel: NH3	4	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
6	Electrolyser: AEC	5	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
7	CO2 capture: None	6	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
8		7	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
9		8	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
10		9	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
11		10	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
12		11	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
13		12	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
14		13	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
15		14	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
16		15	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
17		16	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
18		17	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
19		18	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
20		19	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
21		20	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
22		21	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
23		22	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
24		23	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
25		24	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
26		25	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
27		26	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
28		27	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
29		28	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
30		29	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
31		30	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
32		31	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852
33		32	61529.46253	172088.2025	15296.72911	11075.30325	0	108147.8748	122373.8329	4221.425852

...

[Fig.34]

BASE folder: Results

Note on the 'Results' units: It is important to mention that the **UNITS** for the outcomes and results (if not specified) are in tonnes per hour (t/h) for the 'non-electrical' units and in megawatts (MW) for the 'electrical' ones. The units corresponding to the results shown in the hourly flows are x/1000 (i.e. kg per hour and kilowatt, respectively). The units for mass storages (i.e. hydrogen tank) are in tonnes (t) and the units for electricity storage (i.e. batteries) are in megawatthours (MWh).

Final note

Final note

Hopefully, this user guide was clear and useful for the reader.

In case the user encounters errors during the installation of the software, the author suggests to check the official installation guides for each program. On the other hand, for some issues related to the OptiPlant tool use, the author suggests to check the next section of this document called **‘troubleshooting**’, where one can find some of the most common errors that may appear when using this tool (and how to solve them). When encountered with other errors not mentioned in the ‘troubleshooting’ section, use internet forums or other tools (such as AI) to try to tackle them.

In last resort you may contact one of the author of the model.

Have fun using OptiPlant! ☺

Troubleshooting

Troubleshooting

The purpose of this section is to list and offer a solution for some of the most typical errors one can encounter when running the OptiPlant tool for the first time. This part of the document is though to be edited and upgraded by the different users using the tool. The three main ERRORS tackled in this section are listed below:

1) ERROR: Package X not found

2) ERROR: File not found ’not such file or directory’

3) ERROR: Format error when displaying the simulation results in excel

Troubleshooting

The purpose of this section is to list and offer a solution for some of the most typical errors one can encounter when running the OptiPlant tool for the first time. This part of the document is though to be edited and upgraded by the different users using the tool.

1) ERROR: Package X not found

```
ERROR: ArgumentError: Package JuMP not found in current path.
- Run `import Pkg; Pkg.add("JuMP")` to install the JuMP package.
```

[Fig.38]

If this message appears, you probably haven't activated the environment before running a code. The environment includes all the packages installed necessary to run the code successfully. <u>To solve it, activate the environment:</u>

Enter the 'package manager' by pressing ']' and write 'activate env' on the package manager. If the error still appears, check that the package you need to use is correctly installed by writing 'status' on the package manager -after activating the environment-. If it does NOT appear there, it means the package is not installed. You can install it by writing: 'add ***' (***= name of the package) in the 'env' option. Another reason this error may appear is that you are not calling the package X at the beginning of the code. To solve it, call the necessary packages. It is done like this:

```
using JuMP, Gurobi, CSV, DataFrames, XLSX, ExcelReaders
```

[Fig.39]

Troubleshooting

2) ERROR: File not found 'not such file or directory'

```
ERROR: LoadError: PyError ($(Expr(:escape, :(ccall(#= C:\Users\s210732\.julia\packages\PyCall\twYvK\src\pyfncall.jl:43 =# @pysym(:PyObject_Call),
FileNotFoundError(2, 'No such file or directory')
  File "C:\Users\s210732\.julia\conda\3\x86_64\lib\site-packages\xlrd\__init__.py", line 166, in open_workbook
    file_format = inspect_format(filename, file_contents)
  File "C:\Users\s210732\.julia\conda\3\x86_64\lib\site-packages\xlrd\__init__.py", line 60, in inspect_format
    with open(path, "rb") as f:
```

[Fig.40]

If this message appears, you probably haven't routed the different Julia scripts and excel sheets correctly to the optimization code '**Main.jl**'. To solve it, make sure that you are routing all the files correctly. For instance:

```
#-----------------------------Problem set up------------------------------------
#Project name
Project = "Base"
# Folder name for all csv file
all_csv_files = "All_results"
# Folder paths for data acquisition and writing
Main_folder = "C:/Users/njbca/Documents/Models/OptiPlant-World - Copy" ;
Profiles_folder = joinpath(Main_folder,Project,"Data","Profiles") ;
Inputs_folder = joinpath(Main_folder,Project,"Data","Inputs") ;
Inputs_file = "Data_ammonia_paper"

# Scenario set (same name as exceel sheet)
Scenarios_set =  "ScenariosToRun" ; include("ImportScenarios.jl")
```

[Fig.41]

❗ Pay extra attention to the routes when having numerous folders and subfolders!

Troubleshooting

3) ERROR: Format error when displaying the simulation results in excel

Once the model has run successfully and some results have been generated, one can encounter some problems when reading the CSV ’results’ file. For instance, when importing the ’main results’ CSV, one can get ’weird/unrealistic’ results if the importing method is not defined correctly:

Scenario	Type of unit	Criterion applicati	Installed capacity(MW or t/h)	Total investment(MEuros)
Ammonia-AEC	NH3 plant + ASU - AEC	0	8.484.093.126.040.480	5.652.238.586.058.180
Ammonia-AEC	Waste water plant	0	19.530.929.787.645.200	23.631.089.127.453.200
Ammonia-AEC	Electrolysers AEC	0	17.360.826.477.906.800	6.916.553.268.798.090
Ammonia-AEC	H2 pipeline to NH3 plant	0	15.271.367.626.872.800	0
Ammonia-AEC	Heat from district heating	0	0	0
Ammonia-AEC	Heat sent to district heating	0	985.400.510.885.994	0
Ammonia-AEC	Sale of oxygen	0	13.472.001.346.855.700	0
Ammonia-AEC	H2 pipes compressor	0	9.476.005.393.052.820	0

...

[Fig.42]

DTU

Troubleshooting

... 3) ERROR: Format error when displaying the simulation results in excel

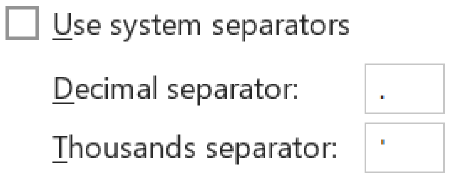
It can be clearly seen that the obtained results for the installed capacities are too large. There might be an error on the CSV reading process. To check that the error is in the file lecture and not actually in the results, we open any of the CSV files on 'Main results' with the notebook. One should see the following:

```
Scenario,Type of unit,Year data,Location,Profile,Fuel,Electrolyser,CO2 capture,CO2 tax level upstream (Eur/kgCO2)
Ammonia-AEC,NH3 plant + ASU - AEC,2030,Bornholm,2019,NH3,AEC,None,0.0,0.0,C1.0_E0.0,-1.0,None,0.0,84.840931260404
Ammonia-AEC,Waste water plant,2030,Bornholm,2019,NH3,AEC,None,0.0,0.0,C1.0_E0.0,-1.0,None,0.0,195.30929787645232,
```

[Fig.43]

The numbers here are realistic and a priori seem correct. Observe that the CSV file separates cells with commas and decimals with dots. To make our excel file read the file and use the separators correctly we do the following:

1. Inside excel, go to File > Options > Advanced and look for this lines here:



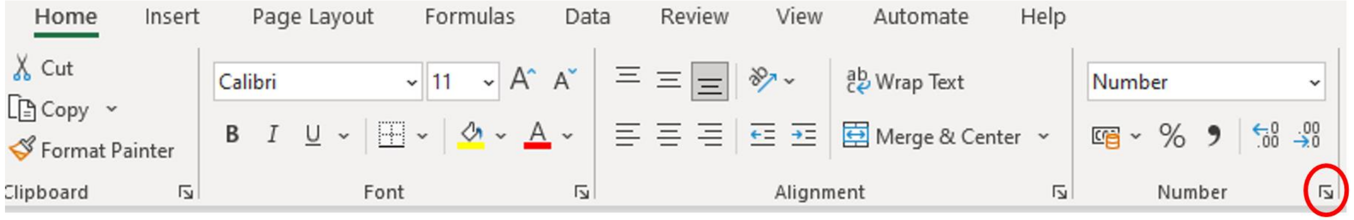
[Fig.44]

The decimal separator should be a dot (.). If possible, set the thousands separator to 'none' or just add any symbol which is not a dot (.), for instance an apostrophe (').

Troubleshooting

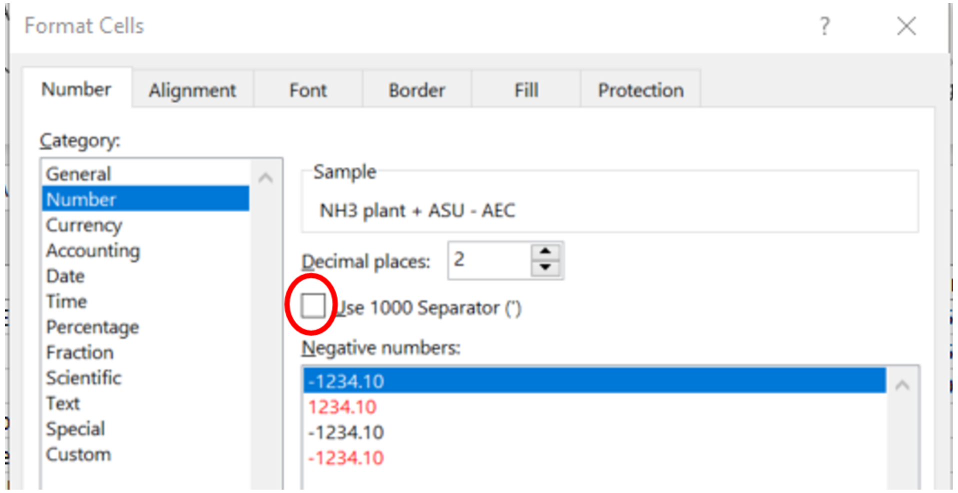
… 3) ERROR: Format error when displaying the simulation results in excel

2. Go to HOME and click in the ‘Number’ tab -bottom corner right of the image-:



[Fig.45]

3. Untick the ‘Use 1000 separator’ box:



[Fig.46]

Troubleshooting

… 3) ERROR: Format error when displaying the simulation results in excel

After completing the previous steps, reset Excel and import the data again. It should look like this now:

Scenario	Type of unit	Criterion applicati	Installed capacity(MW	Total investment(MEuros)
Ammonia-AEC	NH3 plant + ASU - AEC	0	84.84093126	565.2238586
Ammonia-AEC	Waste water plant	0	195.3092979	23.63108913
Ammonia-AEC	Electrolysers AEC	0	17.36082648	691.6553269
Ammonia-AEC	H2 pipeline to NH3 plant	0	15.27136763	0
Ammonia-AEC	Heat from district heating	0	0	0
Ammonia-AEC	Heat sent to district heating	0	98.54005109	0
Ammonia-AEC	Sale of oxygen	0	134.7200135	0
Ammonia-AEC	H2 pipes compressor	0	947.6005393	0

…

[Fig.47]