Sizing Microgrids package – User´s manual

Sebastián Felipe Castellanos Buitrago

sebastian.castellanos@udea.edu.co

Walter Mauricio Villa Acevedo

walter.villa@udea.edu.co

Santiago Marulanda Escobar

santiago.marulandae@udea.edu.co

Pablo Andrés Maya Duque

pablo.maya@udea.edu.co

SENECA

” Alianza Sostenibilidad Energética para Colombia”

Universidad de Antioquia

Medellín – Colombia

Table of contents

1. Introduction

This package uses optimization and simulation models to size the isolated microgrid. Multiple generation units of different technologies (Solar, Wind, Diesel, Batteries.) are considered. The amount of energy generated by each unit in a defined time horizon is calculated. Demand forecasts and environmental variables are used to optimize the sizing microgrid at the lowest possible cost and guaranteeing the reliability.

The package can solve deterministic, stochastic, multiyear deterministic and stochastic multiyear model. The Stochastic Model considers uncertainty in some inputs variables.

The deterministic model is solved with three algorithms: One-Stage Optimization, Two-Stage Optimization with Iterated Local Search (ILS) and Two-Stage Simulation that combines ILS and dispatch strategies. The other three models are solved with the two-stage simulation model. The optimization model is for small data, to solve larger instances there is the two-stage model with optimization or simulation.

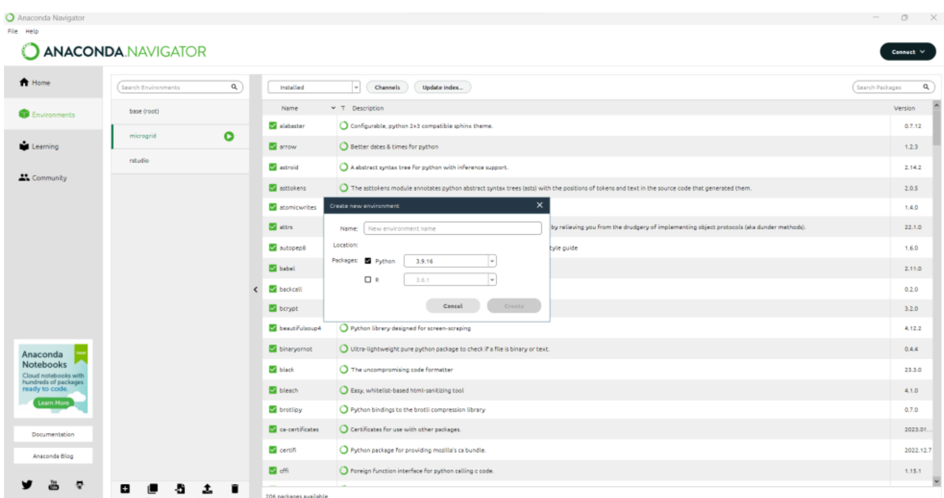
2. Installation

The package is compatible with Python version 3 or higher only. If the optimization model going to be used, the user is expected to have installed Pyomo and Gurobi before running the package. Go to http://www.pyomo.org for more information.

To proceed with the installation process, you must clone the Github package (if you do not have git, you must install it) later, through setuptools, is made the installation, it should be noted that it is installed as develop, so the user can modify the code according to their needs.

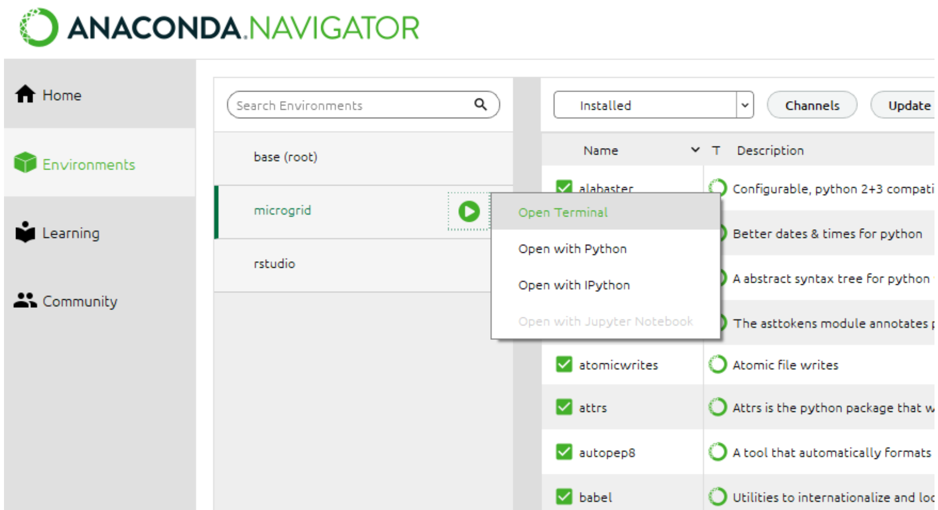
Example of installation through an environment in the anaconda software:

First, an environment is created.

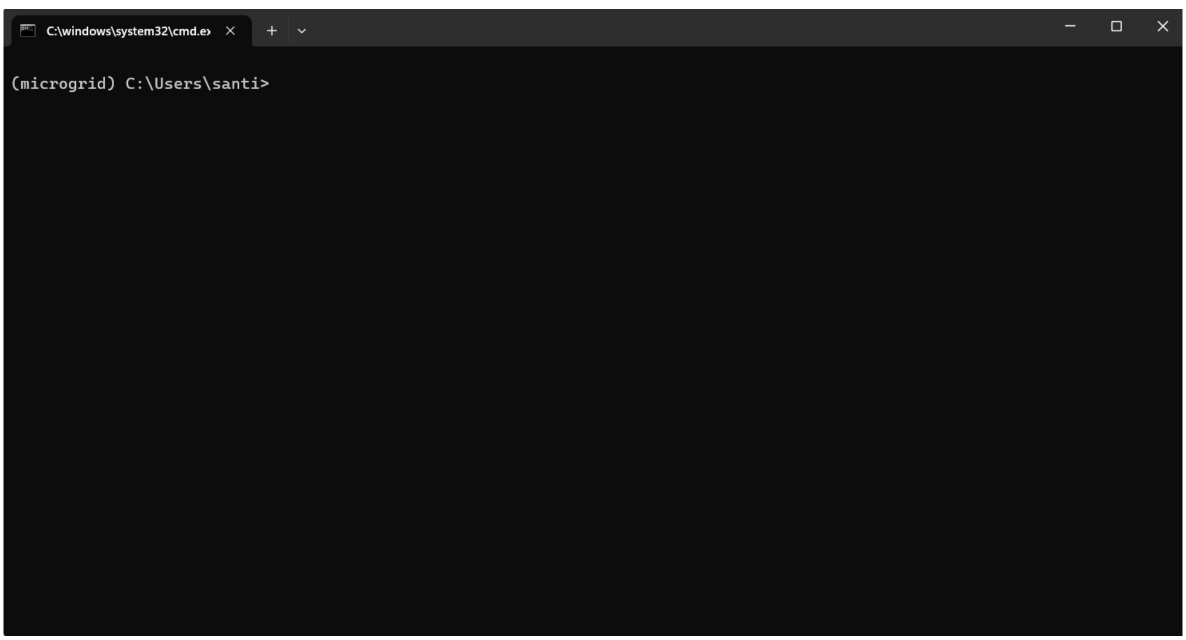


in this case the environment was named "microgrid".

From anaconda, in the green button found in the environment created, open the terminal in the option “open terminal.”



When you open the terminal, you will see the following window. The text in parenthesis corresponds to the name of the environment created, followed by the name of the user, which varies according to the username that executes the application.



Make sure you have git installed in anaconda. If this is not the case, perform the installation using the following command:

Conda install git

Clone repository and enter the microgrids_sizing folder, finally run the setup file to create the setup file to create the package with the following command:

git clone https://github.com/SENECA-UDEA/microgrids_sizing.git

Followed by:

cd microgrids_sizing

Next by:

python setup.py develop

By executing the above commands, we can see how the package and other packages required for its execution are installed.

```
(microgrid) C:\Users\santi>git clone https://github.com/SENECA-UDEA/microgrids_sizing.git
Cloning into 'microgrids_sizing'...
remote: Enumerating objects: 2341, done.
remote: Counting objects: 100% (2341/2341), done.
remote: Compressing objects: 100% (728/728), done.
remote: Total 2341 (delta 1756), reused 2158 (delta 1596), pack-reused 0 eceiving objects:  97% (2271/2341), 1.51 MiB | 2.98 MiB/s
Receiving objects: 100% (2341/2341), 4.00 MiB | 4.06 MiB/s, done.
Resolving deltas: 100% (1756/1756), done.

(microgrid) C:\Users\santi>cd microgrids_sizing

(microgrid) C:\Users\santi\microgrids_sizing>python setup.py develop
running develop
C:\Users\santi\anaconda3\envs\microgrid\lib\site-packages\setuptools\command\easy_install.py:144: EasyInstallDeprecationWarning: easy_install command is dep
recated. Use build and pip and other standards-based tools.
  warnings.warn(
C:\Users\santi\anaconda3\envs\microgrid\lib\site-packages\setuptools\command\install.py:34: SetuptoolsDeprecationWarning: setup.py install is deprecated. Us
e build and pip and other standards-based tools.
  warnings.warn(
running egg_info
creating sizingmicrogrids.egg-info
writing sizingmicrogrids.egg-info\PKG-INFO
writing dependency_links to sizingmicrogrids.egg-info\dependency_links.txt
writing requirements to sizingmicrogrids.egg-info\requires.txt
writing top-level names to sizingmicrogrids.egg-info\top_level.txt
writing manifest file 'sizingmicrogrids.egg-info\SOURCES.txt'
reading manifest file 'sizingmicrogrids.egg-info\SOURCES.txt'
writing manifest file 'sizingmicrogrids.egg-info\SOURCES.txt'
running build_ext
Creating c:\users\santi\anaconda3\envs\microgrid\lib\site-packages\sizingmicrogrids.egg-link (link to .)
sizingmicrogrids 1 is already the active version in easy-install.pth

Installed c:\users\santi\microgrids_sizing
Processing dependencies for sizingmicrogrids==1
Searching for xlsxwriter==3.1.0
Best match: xlsxwriter 3.1.0
Processing xlsxwriter-3.1.0-py3.9.egg
xlsxwriter 3.1.0 is already the active version in easy-install.pth
Installing vba_extract.py script to C:\Users\santi\anaconda3\envs\microgrid\Scripts
```

Finally, by executing the "conda list" command we can see the packages installed in the environment.

```
#
# Name                    Version                   Build  Channel
ca-certificates           2023.01.10           haa95532_0
certifi                   2022.12.7                pypi_0    pypi
charset-normalizer        3.1.0                    pypi_0    pypi
click                     8.1.3                    pypi_0    pypi
idna                      3.4                      pypi_0    pypi
matplotlib                3.7.1                    pypi_0    pypi
numpy                     1.19.5                   pypi_0    pypi
openssl                   1.1.1t               h2bbff1b_0
packaging                 23.1                     pypi_0    pypi
pandas                    2.0.1                    pypi_0    pypi
pip                       23.0.1           py39haa95532_0
plotly                    5.14.1                   pypi_0    pypi
ply                       3.11                     pypi_0    pypi
pyomo                     6.5.0                    pypi_0    pypi
python                    3.9.16               h6244533_2
requests                  2.29.0                   pypi_0    pypi
scipy                     1.10.1                   pypi_0    pypi
setuptools                66.0.0           py39haa95532_0
sizingmicrogrids          1                         dev_0    <develop>
sqlite                    3.41.2               h2bbff1b_0
tenacity                  8.2.2                    pypi_0    pypi
tzdata                    2023c                h04d1e81_0
urllib3                   1.26.15                  pypi_0    pypi
vc                        14.2                 h21ff451_1
vs2015_runtime            14.27.29016          h5e58377_2
wheel                     0.38.4           py39haa95532_0
xlsxwriter                3.1.0                    pypi_0    pypi
```

3. Execution of package

The model can be solved directly in the terminal or in other tools like Spyder or Pycharm.

From the terminal exists six different ways that must be indicated in the instruction or command as will be explained later, they are the following types of models:

1. st = stochastic model with ILS-Simulation algorithm
2. dt = deterministic model with ILS-Simulation algorithm
3. my = multiyear model with ILS-Simulation algorithm
4. sm = stochastic multiyear model with ILS-Simulation algorithm
5. op = deterministic model with Optimization one-stage algorithm
6. ot = Deterministic model with Optimization-ILS Two stage algorithm

For the optimization model you need to install a linear solver such as "Gurobi": www.gurobi.com/ (e.g Gurobi, Cplex, SCJP, etc.)

The package has default values, in this case the "Providencia" data, that is a city in Colombia.

3.1. Execution of package through console

To do a fast run the package through the console, after following the instructions in point 2, you must execute the following command:

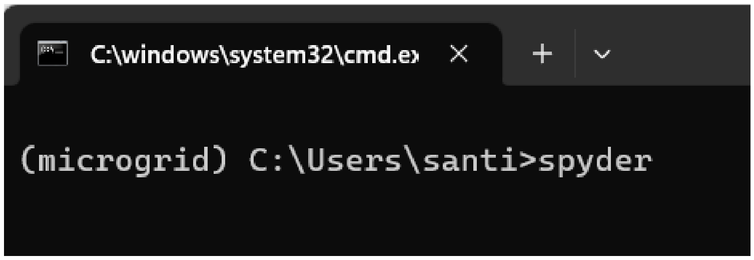
```
python -m sizingmicrogrids -df
"microgrids_sizing/data/Providencia/demand_Providencia.csv" -sw
"microgrids_sizing/data/Providencia/forecast_Providencia.csv"  -id
"microgrids_sizing/data/Providencia/instance_data_Providencia.json" -tm "st"
```

Before it, you have to go back to the initial path, in this case cd.., or modify the description of the default path.

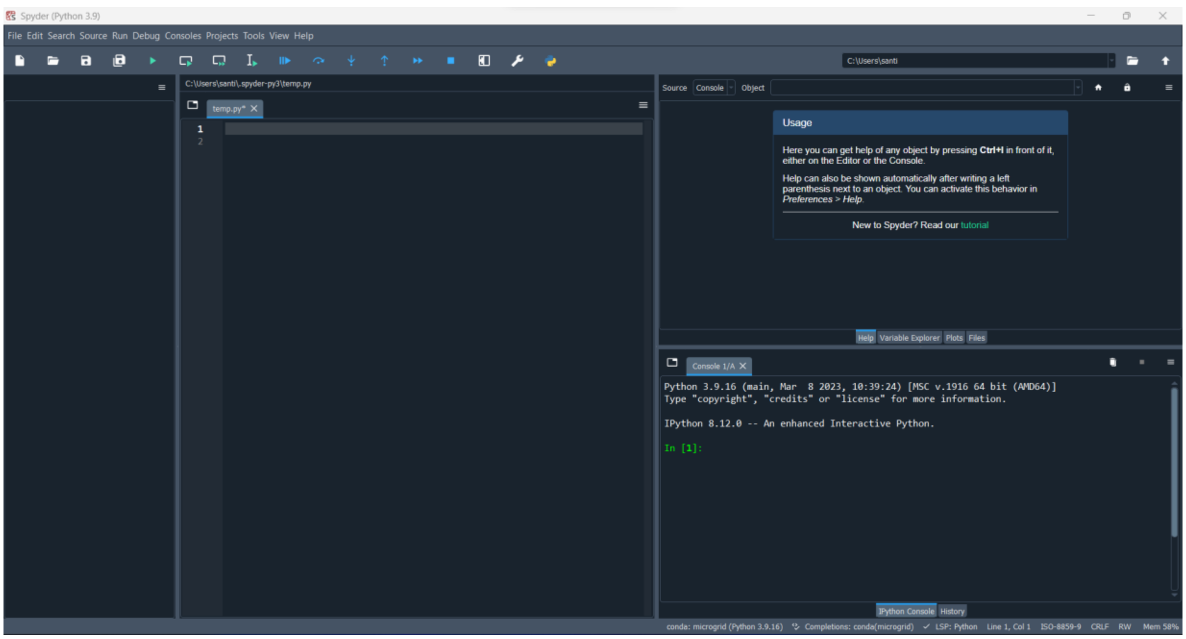
The input data paths must be specified here, which will be discussed in more detail in point four. Additionally, at the end of the command you must indicate the type of model to be executed as explained above, that for this case it is the stochastic model.

3.2. Execution of package through Spyder

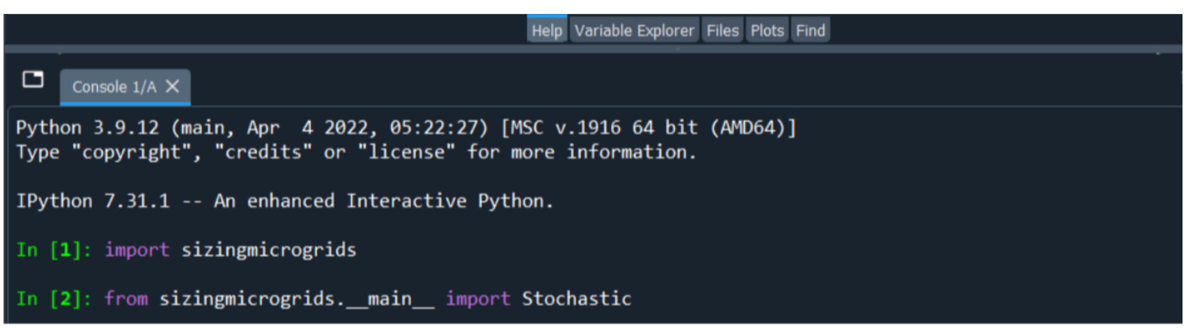
First, open Spyder from the environment terminal, execute the following command:



Immediately Spyder will be executed, make sure that the path where Spyder is executed corresponds to the location where the package is installed.



Import the package, import the function, and choose the path to the data that cannot be entered by default in the Spyder console. Data of "Providencia" is used in this case (Default data are detailed in point 4).



```
In [3]: a = "C:/Users/sebas/Documents/Giogrids_sizing/data/Providencia/demand_Providencia.csv"

In [4]: b = "C:/Users/sebas/Documents/Giogrids_sizing/data/Providencia/forecast_Providencia.csv"

In [5]: c = "C:/Users/sebas/Documents/Giogrids_sizing/data/Providencia/instance_data_Providencia.json"
```

Finally, run the function, in this case stochastic model.

```
In [10]: percent_df, energy_df, renew_df, total_df, brand_df = Stochastic (a,b,c)
C:\Users\sebas\anaconda3\lib\site-packages\scipy\stats\_distn_infrastructure.py:2606: RuntimeWarning:

invalid value encountered in double_scalars

C:\Users\sebas\anaconda3\lib\site-packages\scipy\stats\_continuous_distns.py:639: RuntimeWarning:

invalid value encountered in sqrt

C:\Users\sebas\anaconda3\lib\site-packages\scipy\optimize\minpack.py:175: RuntimeWarning:

The iteration is not making good progress, as measured by the
  improvement from the last ten iterations.
```

As the console execution, in Spyder you must indicate in a,b and c the path of the input data as shown in the example for simplicity in the run, in addition to specifying the type of model imported that you want to execute.

4. Entries

In the package imported through Git, there is some predefined data.

In the "microgrids_sizing" folder, which can be found in the user's folder, there is a folder called "Data". In "Data" you can find the necessary data for the execution of the model corresponding to four geographical locations in Colombia, which are:

- Leticia Town
- Providencia Island
- Puerto Nariño Town
- San Andrés Island
- In addition to another folder called auxiliary, which will be explained below.

When entering the folder for each of these locations different to auxiliar, located in the folder “Data”, four files can be found, which can be modified by the user of the package to find the solution according to the actual location of the decision maker, which are:

- Electrical Demand or Electric power consumption (Excel)
- Forecast or natural resource forecast, solar radiation and wind speed (Excel)
- Instance_data (json)
- Parameters (json)

4.1. Auxiliar

Here you can find three files corresponding to the following parameters and their meaning.

Fiscal incentive

Parameter	Meaning
credit	Credit rate to calculate tax incentive for renewable generation
depreciation	Annual depreciation rate
corporate_tax	Corporate rate to calculate tax incentive for renewable generation
T1	Maximum number of years to apply the investment tax credit
T2	percentage of investment cost depreciation over (T2) year

Multiyear (for the model stochastic or deterministic with multiyear formulation)

Parameter	Meaning
sol_deg	Annual degradation tax for Solar panels
wind_deg	Annual degradation tax for Wind generators
diesel_deg	Annual degradation tax for Diesel generators
bat_deg	Annual degradation tax for batteries
demand_tax	Annual growth rate of demand
fuel_tax	Annual growth rate of fuel price
default_data	Binary, 1 if the user has the multiyear data and 0 if the user has only one year data and the model calculates the subsequent years.

Parameters_cost

life_cicle	Life cicle of the generators Diesel and Batteries
param_r_solar	Percentage that represents the cost of recovery with respect to the cost of installation (for Solar panels)
param_s_solar	Percentage that represents the salvament cost with respect to the cost of installation (for Solar panels)
param_f_solar	Percentage that represents the fixed operation and maintenance cost, with respect to the cost of installation (for Solar panels)
param_v_solar	Percentage that represents the variable operation and maintenance cost, with respect to the cost of installation (for Solar panels)

param_r_wind	Percentage that represents the cost of recovery with respect to the cost of installation (for Wind generators)
param_s_wind	Percentage that represents the salvament cost with respect to the cost of installation (for Wind generators)
param_f_wind	Percentage that represents the fixed operation and maintenance cost, with respect to the cost of installation (for Wind generators)
param_v_wind	Percentage that represents the variable operation and maintenance cost, with respect to the cost of installation (for Wind generators)
param_r_diesel	Percentage that represents the cost of recovery with respect to the cost of installation (for Diesel generators)
param_s_diesel	Percentage that represents the salvament cost with respect to the cost of installation (for Diesel generators)
param_f_diesel	Percentage that represents the fixed operation and maintenance cost, with respect to the cost of installation (for Diesel generators)
param_r_bat	Percentage that represents the cost of recovery with respect to the cost of installation (for batteries)
param_s_bat	Percentage that represents the salvament cost with respect to the cost of installation (for batteries)
param_f_bat	Percentage that represents the fixed operation and maintenance cost, with respect to the cost of installation (for batteries)
param_v_bat	Percentage that represents the variable operation and maintenance cost, with respect to the cost of installation (for batteries)
NSE_COST	Cost of not served energy (varies from L1 to L4, according to the percentage of NSE (First column))

4.2. Parameters

This file contains the data of the generators. The following are the characteristics of each one of them, together with their meaning.

Generators (General)

Parameter	Meaning
Id_gen	id of the generator
self.tec	Technology associated to the generator
br	brand of the generator
area	Generator area (m^2)
cost_up	Investment cost
cost_r	Replacement cost
Cost_s	Salvament cost
cost_fopm	Fixed Operation & Maintenance cost

Solar Generators

Parameter	Meaning
cost_vopm	Variable Operation & Maintenance cost
n	Number of generators
T_noct	Nominal Operating cell Temperature
G_noct	Irradiance operating Standard Condition
Ppv_stc	Rated power
fpv	Derating factor
Kt	Temperature coefficient

Wind Generators

Parameter	Meaning
cost_vopm	Variable Operation & Maintenance cost
s_in	Turbine Minimum Generating Speed (Input Speed)
s_rate	Rated speed of the wind turbine
s_out	Turbine Maximum Generation Speed (Output Speed)
p_y	Rated power of the wind turbine
n_eq	n to calculate the generation curve, usually 1,2 or 3
h	height

Diesel generators

Parameter	Meaning
DG_min	Minimum generation to active the Diesel
DG_max	Rated power, maximum generation
f0	fuel consumption curve coefficient
f1	fuel consumption curve coefficient

Batteries

Parameter	Meaning
efc	Charge efficiency
efd	Discharge efficiency
eb_zero	Energy that the battery has stored at time 0
soc_max	Maximum capacity that the battery can storage
dod_max	Maximum depth of discharge
alpha	self-discharge coefficient
soc_min	Minimum level of energy that must be in the battery
cost_vopm	variable cost

4.3. Forecast

This CSV file consists of nine columns with the following data and the meaning of these data, obtained from the required location.

Parameter	Meaning
t	Time
DNI	Diffused Normal Irradiance
t_amb	Ambient temperature
Wt	Wind speed
Qt	Caudal (for hydraulic energy)
GHI	Global Horizontal Irradiance
day	Day
SF	Shadow factor or cloud opacity
DHI	Diffused Horizontal Irradiance

4.4. Demand

This CSV file consists of two columns with the required data. In this case, the first column indicates the time “t”, and the second column shows the energy demand at time “t”. Both data come from the decision maker's site of interest.

4.5. Instance_data

Parameter	Meaning
Amax	Available area
i_f	Nominal interest rate
Inf	Inflation rate
nse	Maximum NSE allowed
demand_covered	Percent of demand to be covered from demand csv
years	Horizon time from the Project in Years
year_of_data	Year of the data
lpsp	Parameter that determines how often the demand to be covered must be met, 1 is for all hours, and as it increases it is more flexible
splus_cost	Cost of surplus energy
inverter_cost	Inverter cost (if apply)
h2	Height 2 wind generators
coef_hel	Hellman coefficient
W	Distance to the mount in solar installation
case	For solar installation: case 1: direct mount case 2: stand-off case 3: rack mount.
G_stc	Standar solar radiation
fuel_cost	Fuel cost
Alpha_shortlist	Percentage of best generators that will be chosen for the initial solution
N_iterations	Number of iterations in the Iterated local search
tilted_angle	tilted angle of Module (Degrees) – for solar
module_azimuth	Module azimuth – For solar prediction
time_zone	Time zone of the site of interest
longitude	Longitude of site of interest
latitude	Latitude of site of interest
alpha_albedo	Alpha albedo for solar prediction
shading factor	Shading factor for solar prediction

n-scenarios	Number of scenarios to be tested in the stochastical model
stochastic_fuel_cost	Percentage of best generators that will be chosen for the initial solution
percent_robustness	Robustness percent criteria of acceptation in the stochastic model.
sample_escenarios	If the number of scenarios is very, large, it determines how many scenarios will be tested by each best solution found in the stochastic model.

5. Model execution parameters

To run the model from the terminal, the following parameters must or can be included:

Parameter	**Explanation**	**Default**
--df	Path of demand forecast data .csv file	**None**
--sw	Path of weather forecast data .csv file	**None**
--id	Path of instance data .json file	**None**
--tm	type of model { st = stochastic, dt = deterministic, my = multiyear, sm = stochastic multiyear, op = optimization one stage, ot = optimization two stage}	st
--gu	Path of generation and batteries units parameters .json file	Providencia path
--ft	Path of fiscal-tax incentive .json file	Providencia path
--cd	Path of parameters cost .json file	Providencia path
--rs	Seed value for the random object	None = Random
--fp	Base name path for .xlsx output file	Current path
--md	Path of multiyear parameters .json file' (for multiyear formulation)	Providencia path
--sn	Solver name to be used to solve the model (for optimization model)	gurobi
--gp	Solver GAP to be used to solve the model (for optimization model)	0.01

If you only want to import some formulation, the parameters are the following:

			MODEL					
Name	**Explanation**	**Default**	Deterministic Dispatch strategy **“Deterministic”**	Stochastic Dispatch strategy **“Stochastic”**	Deterministic Dispatch strategy – multiyear **“Multiyear”**	Stochastic Dispatch strategy – multiyear **“StocMultiyear”**	Optimization one-stage **“Optimization”**	Optimization two stage with ILS **“IlsOptimization”**
Demand	Demand file path	**None**	X	X	X	X	X	X
Forecast	Forecast file path	**None**	X	X	X	X	X	X
Instance_ filepath	Instance data file path	**None**	X	X	X	X	X	X
Generation units	Generation and batteries json path	Providencia	X	X	X	X	X	X
Tax incentive	Path of fiscal-tax incentive	Providencia	X	X	X	X	X	X
Parameters Cost	Path of parameters cost	Providencia	X	X	X	X	X	X
Folder path	Path to save the output data	Current path	X	X	X	X	X	X
Rand_seed	Seed value for the random object	Random	X	X	X	X		X
gap	Solver gap	0.01					X	X
Solver name	Solver name	gurobi					X	X
my_data	Multiyear data filepath	Providencia			X	X		

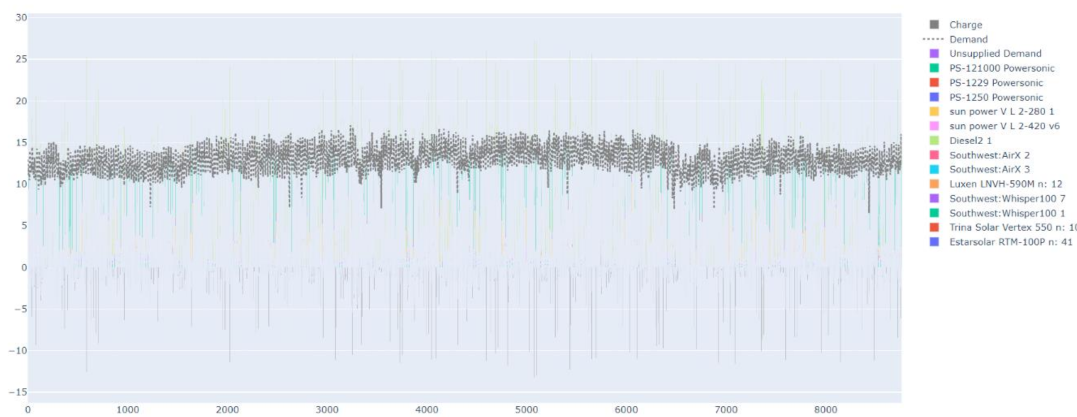
6. Outputs

First, an output with number of hours not served and the performance of the generators is generated. This output will look like this depending on the package is executed via terminal or Spyder. (In Spyder, the elements will be saved.)

```
      sun power V L 2-280 1_cost  PS-1250 Powersonic_cost
0                       0.000000                 0.000000  \
1                       0.000000                 0.000000
2                       0.028256                 0.000318
3                       0.000000                 0.000000
4                       0.000000                 0.000000
...                          ...                      ...
8755                    0.000000                 0.000000
8756                    0.000000                 0.000000
8757                    0.000000                 0.000000
8758                    0.000000                 0.000000
8759                    0.000000                 0.000000

      PS-1229 Powersonic_cost  PS-121000 Powersonic_cost
0                    0.000000                   0.083920
1                    0.000000                   0.000000
2                    0.000286                   0.017626
3                    0.000000                   0.000000
4                    0.000000                   0.000000
...                       ...                        ...
8755                 0.000000                   0.000000
8756                 0.000000                   0.000000
8757                 0.000000                   0.000000
8758                 0.000000                   0.077228
8759                 0.000000                   0.000000

[8760 rows x 40 columns]
best solution number of not served hours: 0
The output results are located in C:\Users\sebas
```

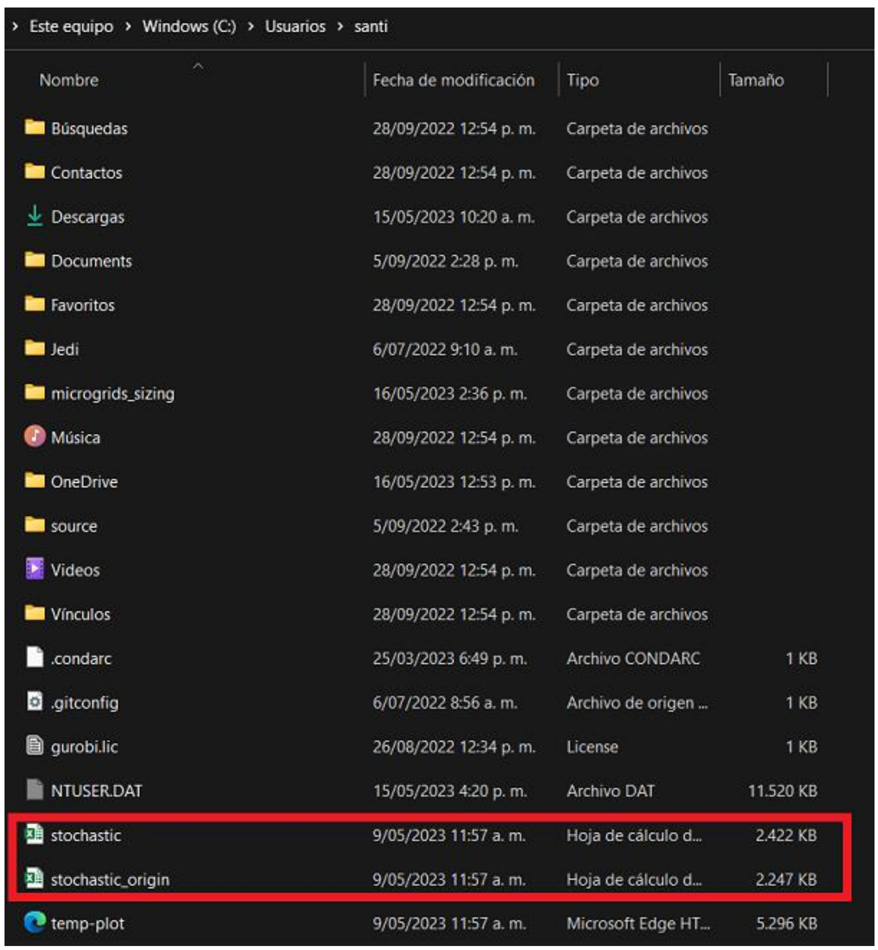
```
In [12]: percent_df
Out[12]:
      PS-12550 Powersonic_%  PS-12550 Powersonic_%charge  LG Chem Resu_%  \
0                       0.0                     0.000104             0.0
1                       0.0                     0.000108             0.0
2                       0.0                     0.000110             0.0
3                       0.0                     0.000111             0.0
4                       0.0                     0.000111             0.0
...                     ...                          ...             ...
8755                    0.0                     0.000085             0.0
8756                    0.0                     0.000089             0.0
8757                    0.0                     0.000094             0.0
8758                    0.0                     0.000101             0.0
8759                    0.0                     0.000105             0.0

      LG Chem Resu_%charge  Ps-1290 Powersonic_%  Ps-1290 Powersonic_%charge  \
0                 0.000022                   0.0                    0.000017
1                 0.000023                   0.0                    0.000017
2                 0.000023                   0.0                    0.000017
3                 0.000024                   0.0                    0.000018
4                 0.000024                   0.0                    0.000018
...                    ...                   ...                         ...
8755              0.000018                   0.0                    0.000013
8756              0.000019                   0.0                    0.000014
8757              0.000020                   0.0                    0.000015
8758              0.000021                   0.0                    0.000016
8759              0.000022                   0.0                    0.000017

      Power Master GPMD6000LN_%  Southwest:Whisper500 1_%  \
0                      0.408730                  0.152400
```

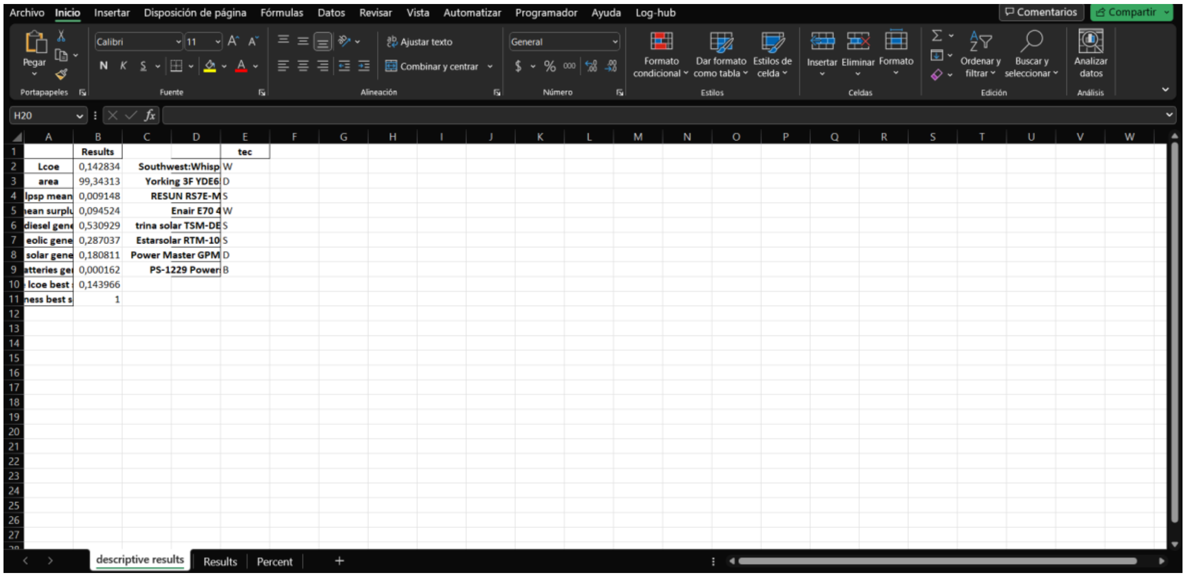
After the execution of the model, it will create an interactive graph where you can see the energy produced with each of the generators, the demand and the unsupplied demand. an example graph is shown below.



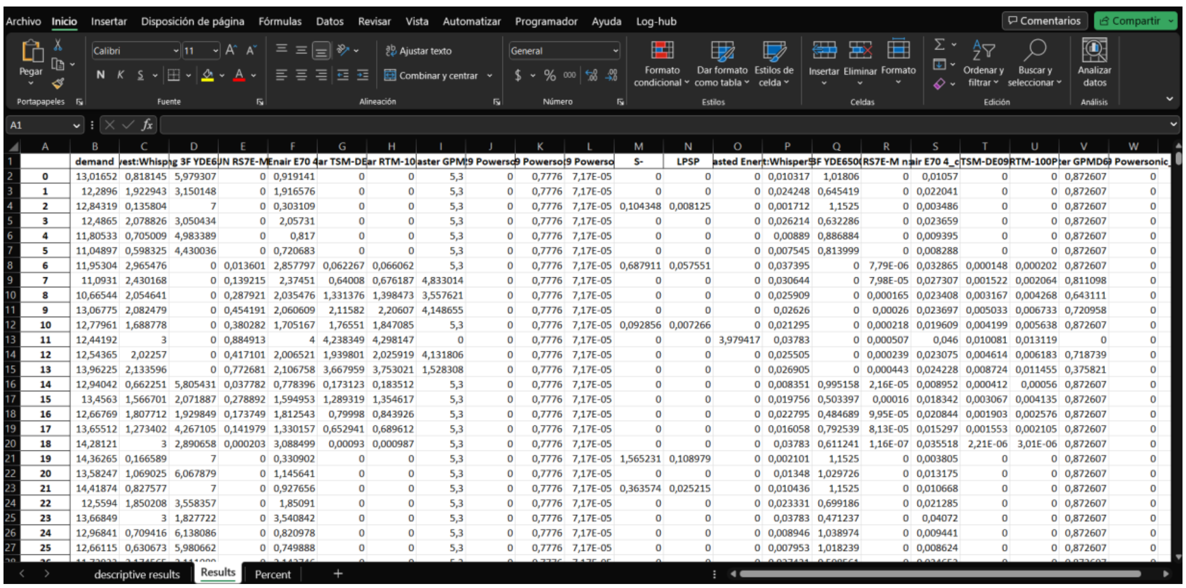
In this graph you can zoom in, or even delete the generators that are not of interest.

Additionally, in the users folder an excel file is created with the name of the type of model executed, where you can find the results of the model solution, such as LCOE, area, LPSP, surplus, power generated by each kind of the technology and the performance of each of the generators.





A	B	C / D	E
	Results		tec
Lcoe	0,142834	Southwest:Whisp	W
area	99,34313	Yorking 3F YDE6	D
lpsp mean	0,009148	RESUN RS7E-M	S
ean surplu	0,094524	Enair E70 4	W
diesel gene	0,530929	trina solar TSM-DE	S
eolic gene	0,287037	Estarsolar RTM-10	S
solar gene	0,180811	Power Master GPM	D
atteries ge	0,000162	PS-1229 Power	B
lcoe best	0,143966		
ness best s	1		



A	demand	est:Whisp	g 3F YDE6	N RS7E-M	Enair E70 4	ar TSM-DE	ar RTM-10	aster GPM	9 Powersc	9 Powerso	9 Powerso
0	13,01652	0,818145	5,979307	0	0,919141	0	0	5,3	0	0,7776	7,17E-05
1	12,2896	1,922943	3,150148	0	1,916576	0	0	5,3	0	0,7776	7,17E-05
2	12,84319	0,135804	7	0	0,303109	0	0	5,3	0	0,7776	7,17E-05
3	12,4865	2,078826	3,050434	0	2,05731	0	0	5,3	0	0,7776	7,17E-05
4	11,80533	0,705009	4,983389	0	0,817	0	0	5,3	0	0,7776	7,17E-05
5	11,04897	0,598325	4,430036	0	0,720683	0	0	5,3	0	0,7776	7,17E-05
6	11,95304	2,965476	0	0,013601	2,857797	0,062267	0,066062	5,3	0	0,7776	7,17E-05
7	11,0931	2,430168	0	0,139215	2,37451	0,64008	0,676187	4,833014	0	0,7776	7,17E-05
8	10,66544	2,054641	0	0,287921	2,035476	1,331376	1,398473	3,557621	0	0,7776	7,17E-05
9	13,06775	2,082479	0	0,454191	2,060609	2,11582	2,20607	4,148655	0	0,7776	7,17E-05
10	12,77961	1,688778	0	0,380282	1,705167	1,76551	1,847085	5,3	0	0,7776	7,17E-05
11	12,44192	3	0	0,884913	4	4,238349	4,298147	0	0	0,7776	7,17E-05
12	12,54365	2,02257	0	0,417101	2,006521	1,939801	2,025919	4,131806	0	0,7776	7,17E-05
13	13,96225	2,133596	0	0,772681	2,106758	3,667959	3,753021	1,528308	0	0,7776	7,17E-05
14	12,94042	0,662251	5,805431	0,037782	0,778396	0,173123	0,183512	5,3	0	0,7776	7,17E-05
15	13,4563	1,566701	2,071887	0,278892	1,594953	1,289319	1,354617	5,3	0	0,7776	7,17E-05
16	12,66769	1,807712	1,929849	0,173749	1,812543	0,79998	0,843926	5,3	0	0,7776	7,17E-05
17	13,65512	1,273402	4,267105	0,141979	1,330157	0,652941	0,689612	5,3	0	0,7776	7,17E-05
18	14,28121	3	2,890658	0,000203	3,088499	0,00093	0,000987	5,3	0	0,7776	7,17E-05
19	14,36265	0,166589	7	0	0,330902	0	0	5,3	0	0,7776	7,17E-05
20	13,58247	1,069025	6,067879	0	1,145641	0	0	5,3	0	0,7776	7,17E-05
21	14,41874	0,827577	7	0	0,927656	0	0	5,3	0	0,7776	7,17E-05
22	12,5594	1,850208	3,558357	0	1,85091	0	0	5,3	0	0,7776	7,17E-05
23	13,66849	3	1,827722	0	3,540842	0	0	5,3	0	0,7776	7,17E-05
24	12,96841	0,709416	6,138086	0	0,820978	0	0	5,3	0	0,7776	7,17E-05
25	12,66115	0,630673	5,980662	0	0,749888	0	0	5,3	0	0,7776	7,17E-05

A	S-	LPSP	asted Ener	t:Whisper	3F YDE650	RS7E-M n	air E70 4_c	TSM-DE09	RTM-100P	er GPMD6	Powersonic_
0	0	0	0	0,010317	1,01806	0	0,01057	0	0	0,872607	0
1	0	0	0	0,024248	0,645419	0	0,022041	0	0	0,872607	0
2	0,104348	0,008125	0	0,001712	1,1525	0	0,003486	0	0	0,872607	0
3	0	0	0	0,026214	0,632286	0	0,023659	0	0	0,872607	0
4	0	0	0	0,00889	0,886884	0	0,009395	0	0	0,872607	0
5	0	0	0	0,007545	0,813999	0	0,008288	0	0	0,872607	0
6	0,687911	0,057551	0	0,037395	0	7,79E-06	0,032865	0,000148	0,000202	0,872607	0
7	0	0	0	0,030644	0	7,98E-05	0,027307	0,001522	0,002064	0,811098	0
8	0	0	0	0,025909	0	0,000165	0,023408	0,003167	0,004268	0,643111	0
9	0	0	0	0,02626	0	0,00026	0,023697	0,005033	0,006733	0,720958	0
10	0,092856	0,007266	0	0,021295	0	0,000218	0,019609	0,004199	0,005638	0,872607	0
11	0	0	3,979417	0,03783	0	0,000507	0,046	0,010081	0,013119	0	0
12	0	0	0	0,025505	0	0,000239	0,023075	0,004614	0,006183	0,718739	0
13	0	0	0	0,026905	0	0,000443	0,024228	0,008724	0,011455	0,375821	0
14	0	0	0	0,008351	0,995158	2,16E-05	0,008952	0,000412	0,00056	0,872607	0
15	0	0	0	0,019756	0,503397	0,00016	0,018342	0,003067	0,004135	0,872607	0
16	0	0	0	0,022795	0,484689	9,95E-05	0,020844	0,001903	0,002576	0,872607	0
17	0	0	0	0,016058	0,792539	8,13E-05	0,015297	0,001553	0,002105	0,872607	0
18	0	0	0	0,03783	0,611241	1,16E-07	0,035518	2,21E-06	3,01E-06	0,872607	0
19	1,565231	0,108979	0	0,002101	1,1525	0	0,003805	0	0	0,872607	0
20	0	0	0	0,01348	1,029726	0	0,013175	0	0	0,872607	0
21	0,363574	0,025215	0	0,010436	1,1525	0	0,010668	0	0	0,872607	0
22	0	0	0	0,023331	0,699186	0	0,021285	0	0	0,872607	0
23	0	0	0	0,03783	0,471237	0	0,04072	0	0	0,872607	0
24	0	0	0	0,008946	1,038974	0	0,009441	0	0	0,872607	0
25	0	0	0	0,007953	1,018239	0	0,008624	0	0	0,872607	0