

sizingmicrogrids 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

Mateo Giraldo Zuluaga mateo.giraldoz@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	
2. Installation	3
3. Execution of package	6
3.1. Execution of package through console	6
3.2. Execution of package through spyder	7
4. Entries	8
4.1. Auxiliar	9
4.2. Parameters	11
4.3. Forecast	12
4.4. Demand	13
4.5. Instance_data	13
5. Model execution parameters	14
6. Outputs	

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 optimization model is for small data, to solve larger instances there is the two-stage model with ILS.

To improve the performance of computing time, there is the simulation model with ILS and dispatch strategies, and it can be deterministic or uncertainty.

To improve the quality of solutions, the multi-year model with simulation and ILS can be used instead of one-year as the previous ones.

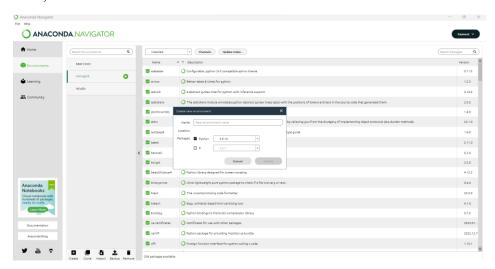
2. Installation

The package is compatible with <u>Python version 3 or higher only</u>. 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.

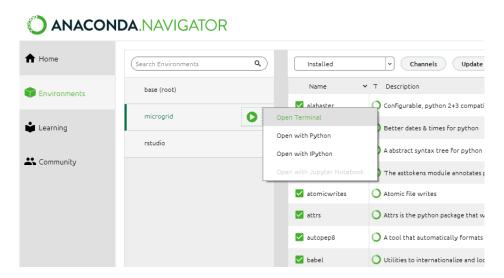
Example of installation through an environment in the anaconda software:

First, an environment is created.

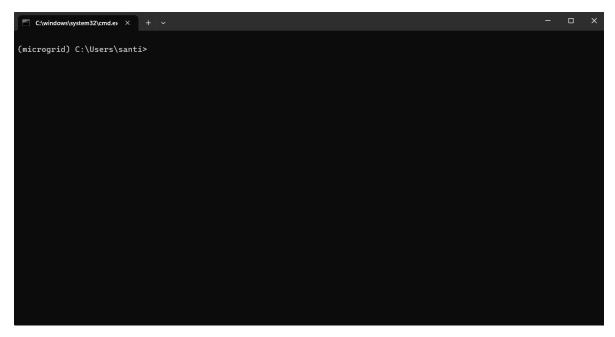


in this case the environment was named "microgid".

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.

```
| Caincoprid C:\Users\santi>pit clone https://github.com/SENECA-UDEA/microgrids_sizing.git
| Caincaprid C:\Users\santi>pit clone https://github.com/SENECA-UDEA/microgrids_sizing.git
| Candang into 'sicrogrids_sizing'...
| remote: Enumerating objects: 190% (2341/2341), done.
| remote: Compressing objects: 190% (2734/2341), done.
| remote: Compressing objects: 180% (7234/728), done.
| remote: Compressing objects: 180% (7234/728), done.
| remote: Total 2341 (delta 1756), remote 2186 (delta 1596), pack-reused 0 eceiving objects: 97% (2271/2341), 1.51 MiB | 2.98 MiB/S Receiving objects: 180% (2341/2341), 4.00 MiB | 4.06 MiB/s, done.
| Recolving deltas: 100% (1756/1756), done.
| Recolving deltas: 100% (1756/1756) done.
| Recolving deltas: 100% (1756/17
```

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

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
- 2. dt = deterministic
- 3. my = multiyear
- 4. sm = stochastic multiyear
- 5. op = optimization one stage
- 6. ot = optimization two stage

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

The model 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.

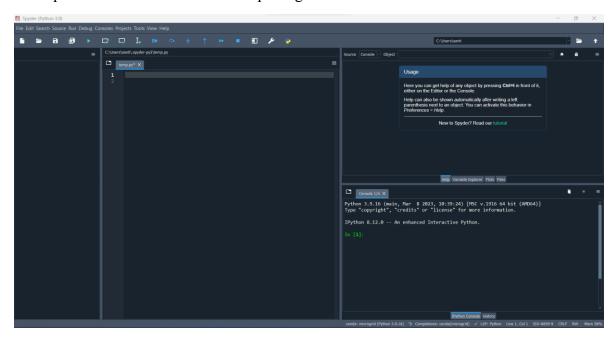
The input data paths must be specified here, which will be discussed in more detail in point four. Additionally, at the end of the command you must indicate the type of model to be executed as explained above, that for this case it is the stochastic model.

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).

```
Help Variable Explorer Files Plots Find

Console 1/A ×

Python 3.9.12 (main, Apr 4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: import sizingmicrogrids

In [2]: from sizingmicrogrids.__main__ import Stochastic
```

```
In [3]: a = "C:/Users/sebas/Documents/Girogrids_sizing/data/Providencia/demand_Providencia.csv"
In [4]: b = "C:/Users/sebas/Documents/Girogrids_sizing/data/Providencia/forecast_Providencia.csv"
In [5]: c = "C:/Users/sebas/Documents/Girogrids_sizing/data/Providencia/instance_data_Providencia.json"
```

finally, run the function, in this case stochastic.

```
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
- Providencia
- Puerto Nariño
- San Andrés
- 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:

- Demand (Excel)
- Forecast (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 generators
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 generators)
param_s_solar	Percentage that represents the salvament cost
	with respect to the cost of installation (for
	Solar generators)
param_f_solar	Percentage that represents the fixed operation
	and maintenance cost, with respect to the cost
	of installation (for Solar generators)
param_v_solar	Percentage that represents the variable
	operation and maintenance cost, with respect to
	the cost of installation (for Solar generators)

param_r_wind	Percentage that represents the cost of recovery
param_r_wind	with respect to the cost of installation (for
	Wind generators)
param_s_wind	Percentage that represents the salvament cost
pmm_s_mm	with respect to the cost of installation (for
	Wind generators)
param_f_wind	Percentage that represents the fixed operation
pmmi_i_mi	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
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 Tmperature
G_noct	Irradiance operating Normal 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
fl	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
tlpsp	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
caso	For solar installation:
	caso 1: direct mount
	caso 2: stand-off
	caso 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
tilted and	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 criterio of aceptation 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		
	st = stochastic,			
	dt = deterministic,			
	my = multiyear,			
	sm = stochastic multiyear,			
	op = optimization one stage,			
	ot = optimization two stage}			
gu	Path of generation and batteries units parameters	Providencia path		
	.json file			
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	Providencia path		
	multiyear formulation)			
sn	Solver name to be used to solve the model (for	gurobi		
	optimization model)			
gp	Solver GAP to be used to solve the model (for	0.01		
	optimization model)			

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			

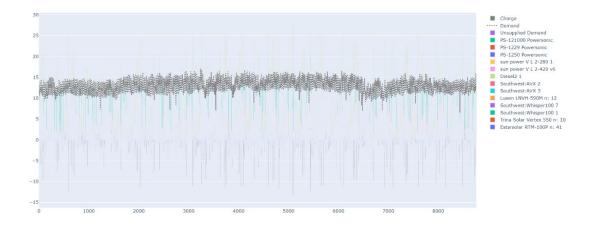
6. Outputs

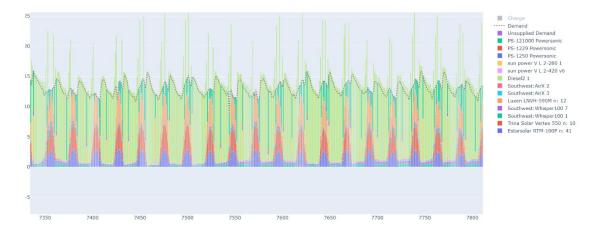
first, an output with number of hours not served and the performance of the generators is generated. This output will look like this depending on the package is executed via terminal or Spyder. (In Spyder, the elements will be saved.)

```
Sun power V L 2-280 1_cost | Cost | C
```

```
In [12]: percent_df
      PS-12550 Powersonic_% PS-12550 Powersonic_%charge
                                                          LG Chem Resu %
                        0.0
                                                0.000104
                                                                     0.0
                        0.0
                                                0.000108
                                                                     0.0
                                                0.000110
                        0.0
                                                                     0.0
                                                0.000111
                        0.0
                                                                     0.0
                                                0.000111
                        0.0
                                                                     0.0
                                                0.000085
                                                                     0.0
                                                                     0.0
8758
                        0.0
                                                0.000101
                                                                     0.0
      LG Chem Resu_%charge Ps-1290 Powersonic_% Ps-1290 Powersonic_%charge
                  0.000022
                  0.000023
                                             0.0
                                                                    0.000017
                  0.000023
                                             0.0
                                                                    0.000017
                  0.000024
                                                                    0.000018
                                             0.0
                  0.000024
                                             0.0
                                                                    0.000018
8755
                                                                    0.000013
                  0.000018
                  0.000019
                                                                    0.000014
8756
                                             0.0
                                                                    0.000015
                  0.000020
                                             0.0
                                             0.0
8758
                  0.000021
                                                                    0.000016
                  0.000022
8759
                                             0.0
                                                                    0.000017
      Power Master GPMD6000LN_% Southwest:Whisper500 1_% \
                       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.

