

# Off-Grid Systems Modelling with MicroGridsPy

## Hands-on 4

GitHub repository for source code:

[SESAM-Polimi/MicroGridsPy-SESAM: MicroGridsPy - SESAM-PoliMi \(github.com\)](https://github.com/SESAM-Polimi/MicroGridsPy-SESAM)

MicroGridsPy is an open-source project, currently under active development, check out the detailed online documentation for usage guidance and updates:

<https://microgridspy-documentation.readthedocs.io/en/latest/>

---

## Learning outcomes

---

By the end of this exercise, you will learn how to:

- 1) Explore the MicroGridsPy User Interface functionalities
- 2) Understand how to initialize and input Renewables Time Series Data in MicroGridsPy
- 3) Understand how Renewables Time Series Data affect results

---

### Important requirement

Ensure that MicroGridsPy is correctly installed and functioning within the dedicated conda environment. It will be also required a stable internet connection. Additionally, it is suggested (though not strictly required) to have Gurobi installed for improved performance of the model optimization. Refer to the installation procedure and guidance provided in Hands-on 1 for detailed instructions.

---

# Introduction

---

## Launch the Application: Recap

Let's briefly recap how to launch the user interface after having correctly creating the MicroGridsPy environment using conda and downloading the MicroGridsPy folder.

- Launch Spyder Using Anaconda. To launch Spyder, an integrated development environment (IDE) for Python, users can proceed this in two ways:
  1. Using Anaconda Prompt: Open the Anaconda Prompt, activate the mgpy environment (if not already activated), type the command *spyder* and press enter to open Spyder.
  2. Using Anaconda Navigator: Launch Anaconda Navigator, activate the mgpy environment from the "Environments" tab. Then, in Anaconda Navigator's "Home" tab, find "Spyder" in the list of available applications and click "Launch" to open Spyder.
- Launch the Interface within Spyder: Once Spyder is open, follow these steps to launch the MicroGridsPy interface:
  1. Check if the default layout is suitable; if not, set it from the "View" button.
  2. Locate the MicroGridsPy working folder using the Spyder interface's "File" menu.
  3. Open the project folder by double-clicking on it.
  4. Navigate to the "Code/User Interface" folder within the project folder.
  5. Look for the `app_main.py` file and double-click on it to open it.
  6. Run `app_main.py` by pressing F5 or using the "Run" button in Spyder's toolbar.

Upon completing these steps, the MicroGridsPy interface will launch within Spyder resulting in this situation:

## Hands-on objective

In this hands-on session, we will delve into the Renewables Time Series data page of MicroGridsPy interface to master its usage through two key activities: simulating endogenously the time series data thanks to NASA POWER project and providing exogenous renewables data. Our goal is to understand how to manipulate and analyze renewables time series data within the software effectively.

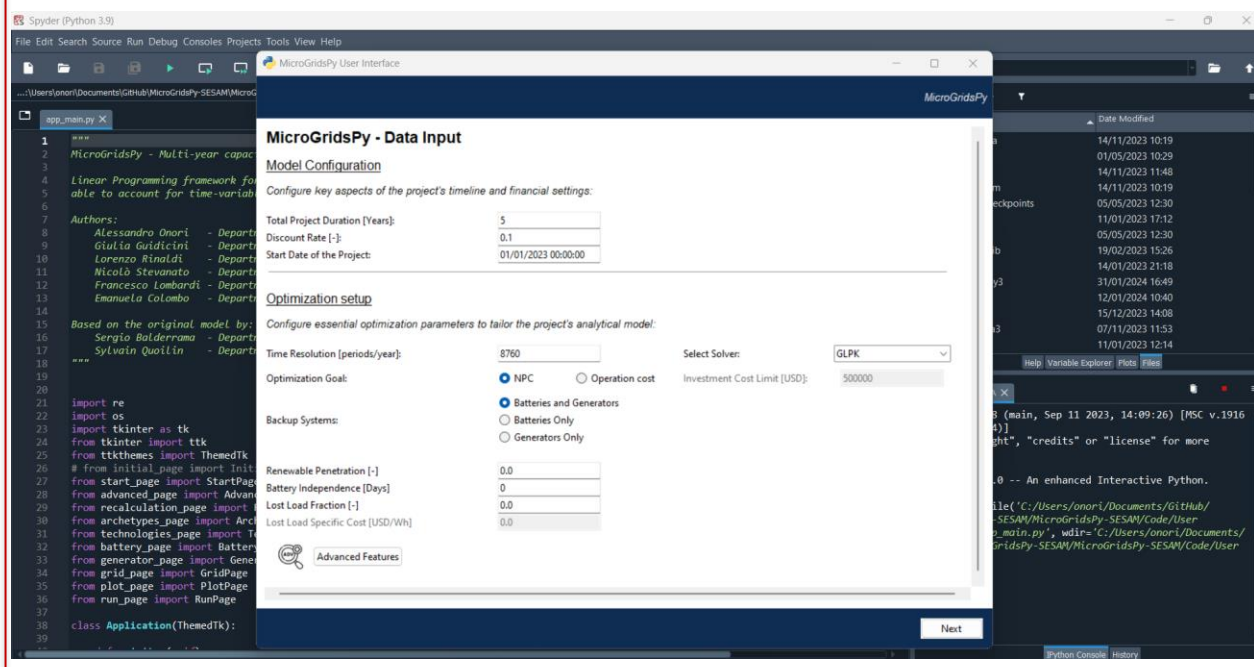
## Default configuration

To recap, the default configuration we'll work with includes a 20-year model horizon, hourly operation, a 10% discount rate, and a focus on minimizing net present costs. The model is allowed to consider both batteries as storage system and generators as backup system and there are no additional constraints implemented. The exemplary case study focuses on Sub-Saharan village in Zambia with an annual growth rate of 5%, leveraging NASA Power Project data for resource assessment (specifically solar and wind). Technology characterization includes photovoltaic modules, wind turbines, battery banks, and diesel generator with exemplary but not realistic parameters.

### Note

If you have Gurobi installed with an activated license, you can choose to leave Gurobi as the default solver for your MicroGridsPy simulations. This solver is known for its capabilities and speed, which can significantly expedite the optimization process.

However, if you don't have a Gurobi license, you can opt to use the open-source free-to-use solver option, "GLPK," but making a specific adjustment. Please select "GLPK" from the solver dropdown menu and set the time horizon to 5 years. While this configuration may result in slight variations in the results (though not significantly), it substantially reduces the computational effort and the time required for solving. You can expect the solving time to be approximately 45-60 minutes (depending on your system computational performance), striking a favorable balance between result accuracy and computational efficiency.



# Endogenous RES Calculation

The RES (Renewable Energy Sources) estimation features within MicroGridsPy provide a convenient and comprehensive way to estimate renewable energy potential using NASA Power Project database. By simply checking the "activate" check button, all relevant parameters will be enabled, allowing users to tailor their RES estimation. The layout offers a scrollable area for a comprehensive list of parameters, including latitude, longitude, time zone, and turbine information. Custom fonts and tooltips enhance the user experience, making it a user-friendly interface for setting up required parameters. In the default configuration, the system is set to evaluate a specific location in Zambia, taking advantage of its solar-rich environment. The configuration includes a 1 kW solar PV module with standard values for parameters such as tilt angle, ground reflectivity (ro\_ground), and operating temperature. Additionally, the horizontal axis Alstom.Eco.80 wind turbine, rated at 1670 kW, is integrated into the assessment.

**Location Coordinates**  
Input latitude and longitude in DMS system (degrees, minutes and seconds)

**Solar PV parameters**  
Specifications for **solar panel module**, including **nominal power**, **tilt angle**, **azimuth**, and **efficiency-related coefficients** to calculate **solar generation potential** accurately

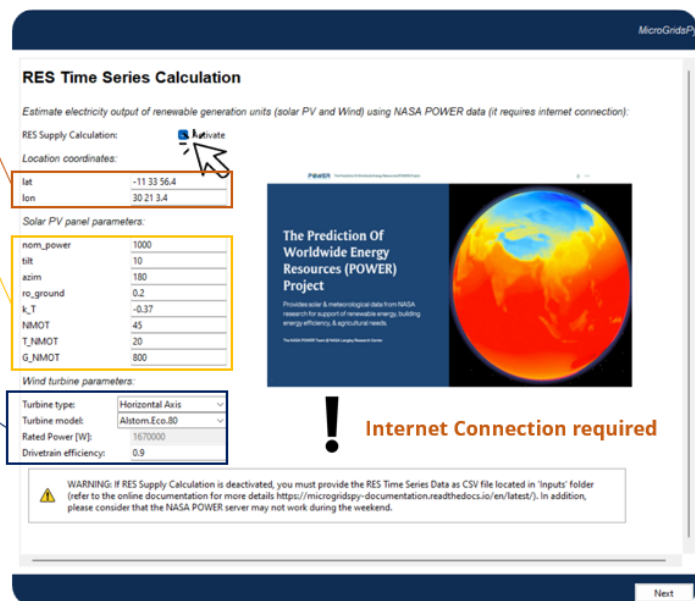
**Wind turbine parameters**  
Select the **turbine type and model**, see the associated **rated power** and input **drivetrain efficiency** to estimate **wind energy generation**



Models available:

- **Horizontal Axis:** Alstom.Eco.80 and NPS100c-21
- **Vertical Axis:** Hi-VAWT.DS1500 and Hi-VAWT.DS700

 Code/Inputs/WT\_Power\_Curve.csv



**MicroGridsPy**

### RES Time Series Calculation

Estimate electricity output of renewable generation units (solar PV and Wind) using NASA POWER data (it requires internet connection):

RES Supply Calculation: ☒ **Activate**

Location coordinates:

lat: -11 33 56.4  
lon: 30 21 3.4

Solar PV panel parameters:

nom_power	1000
tilt	10
azim	180
ro_ground	0.2
k_t	-0.37
NMOT	45
T_NMOT	20
G_NMOT	800

Wind turbine parameters:

Turbine type: Horizontal Axis  
Turbine model: Alstom.Eco.80  
Rated Power (W): 1670000  
Drivetrain efficiency: 0.9

**! Internet Connection required**

WARNING: If RES Supply Calculation is deactivated, you must provide the RES Time Series Data as CSV file located in 'Inputs' folder (refer to the online documentation for more details <https://microgridspy-documentation.readthedocs.io/en/latest/>). In addition, please consider that the NASA POWER server may not work during the weekend.

Next

## Exercise 1

In this exercise, you will be adapting MicroGridsPy's model for a village located at the following coordinates:

Lat: -43° 59' 4.254" , Lon: -65° 21' 16.527"

The objective is to estimate the Renewable Energy Sources (RES) Time Series using data from NASA's POWER project and show the influence of location-specific renewable energy data on model outcomes. Use the Demand.csv file located in "Material" folder as input and select the Horizontal Axis NPS100c-21 wind turbine model.

Once set the described (exemplative) scenario, run the model while keeping other parameters that are NOT affected by this new scenario (not necessary all) unchanged. Refer to the previous hands-on guides for detailed steps on running the model.

After the optimization process, take note of key results and the relevant plots generated by MicroGridsPy. The aim is also to investigate the results obtained, trying to understand the reasons why peculiar results were obtained, inquiring in particular about the characteristics of the location provided.

## Exogenous RES Time Series

---

In MicroGridsPy, renewables time series data can be assessed through two approaches. Endogenous estimation through the NASA POWER project data, automating the generation of load profiles within the software. On the other hand, exogenous RES time series allows users to input their own specific data, or using data downloaded from an online database, offering the flexibility to model energy needs based on external or pre-existing datasets.

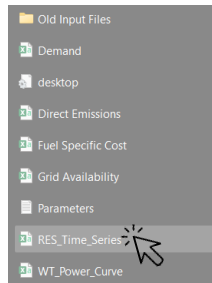
To provide exogenous RES time series data in MicroGridsPy:

- Begin by ensuring that the 'RES Supply Calculation' check box for RES estimation in the user interface is NOT activated.
- Locate the RES\_Time\_Series.csv file within the Code/Inputs directory of your MicroGridsPy folder.
- Edit this CSV file, adhering to the structure shown in the picture below: numbered columns for types of renewable sources, and hourly electricity output in watts for each row corresponding to the hours in a year. After editing the file with your specific data, save the changes.

### RES Time Series Calculation

Estimate electricity output of renewable generation units

RES Supply Calculation: ☐ Activate



 Code/Inputs/RES\_Time\_Series .csv

Numbered columns equal to the number of renewables

	1 2 ...
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
....	

Hourly electricity output in W for each renewable sources unit

Hours in a year

**EXAMPLE**

	1	
1	0	432812,5
2	0	461477,8
3	0	430550,8
4	0	318517,4
5	0	156753,5
6	0	346411,4
7	21,26435	612342,8
8	193,8794	589845,4
9	382,0014	455267,6
10	561,6518	302666,6
11	708,8004	186153
12	805,386	101622,2
13	840,2869	52911,46
14	810,5032	28713,44
15	718,4495	268
16	574,5458	55

**CSV**

- Return to the MicroGridsPy interface and proceed by clicking 'Next' to continue with the simulation using your provided demand data.

## Exercise 2

To integrate exogenous Renewable Energy Sources (RES) time series data from renewables.ninja into MicroGridsPy, follow these steps:

- Navigate to the Renewables.ninja website ([Renewables.ninja](https://renewables.ninja)) and enter the specific location coordinates: lat -2.0810658244483826, lon 41.10252219722386 (Faza island, Kenya)
- Set the standard parameters for photovoltaic (PV) solar panels and wind turbines. Use the default parameters except for tilt angle of PV which should be set at 10°.
- Click the 'Run' button on Renewables.ninja to simulate the energy output based on historical weather and solar irradiation data for the selected location.
- Download the resulting CSV data file once the simulation is complete.
- Open the downloaded CSV file and adjust the time column to align local time with UTC time by shifting the appropriate rows from the top to the bottom of the spreadsheet.
- Copy the hourly electricity output data for both solar and wind from the adjusted CSV file into the RES\_Time\_Series.csv file in the Code/Inputs directory of MicroGridsPy.
- Ensure that the structure of the RES\_Time\_Series.csv file matches the required format, with columns for each type of renewable source and rows for hourly electricity output in Watt.
- Save the updated RES\_Time\_Series.csv file, ensuring you do not change the file name.



**Location**

Renewables.ninja

Search by location

Lat: -6.3073 Lon: 35.1367

Solar photovoltaic power (PV)

Wind power

Solar photovoltaic power (PV)

Dataset: MERRA-2 (global)

Select a year of data: 2019

Capacity (kW): 1

System loss (fraction): 0.1

Tracking: None

Tilt (°): 35

Azimuth (°): 180

☒ Include raw data

**PV panel parameters**

Run


The outputs displayed show the **variation in solar energy production over the year** and the **monthly capacity factor** indicating the efficiency of the solar PV system, with an overall average provided.

☒ Include raw data

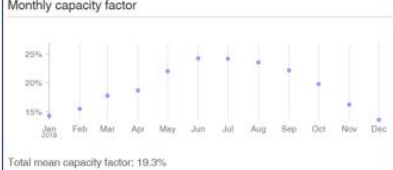
Click here to include additional outputs with the simulation itself. Raw weather data is included in the downloadable output file but not visualized in the web application.

Press to simulate the **expected energy output of the PV system based on historical weather and solar irradiation data specific to the chosen location.**

**Daily mean**



**Monthly capacity factor**



Total mean capacity factor: 19.3%

Save hourly output as CSV

License: Creative Commons Attribution-NonCommercial  
Citation: Planerger and Staffell (2016)



Press to **download the hourly output of the PV panel in a comma-separated values (CSV) file**. If raw data included, the csv file would provide time, local time, electricity, irradiance and temperature data

time	local_time	electricity
01/01/2019 00:00	01/01/2019 03:00	0
01/01/2019 01:00	01/01/2019 04:00	0
01/01/2019 02:00	01/01/2019 05:00	0
01/01/2019 03:00	01/01/2019 06:00	0.003
01/01/2019 04:00	01/01/2019 07:00	0.054
01/01/2019 05:00	01/01/2019 08:00	0.191
01/01/2019 06:00	01/01/2019 09:00	0.351
01/01/2019 07:00	01/01/2019 10:00	0.473
01/01/2019 08:00	01/01/2019 11:00	0.547
01/01/2019 09:00	01/01/2019 12:00	0.575
01/01/2019 10:00	01/01/2019 13:00	0.558
01/01/2019 11:00	01/01/2019 14:00	0.496
01/01/2019 12:00	01/01/2019 15:00	0.390
01/01/2019 13:00	01/01/2019 16:00	0.245
01/01/2019 14:00	01/01/2019 17:00	0.089
01/01/2019 15:00	01/01/2019 18:00	0.015
01/01/2019 16:00	01/01/2019 19:00	0
01/01/2019 17:00	01/01/2019 20:00	0
01/01/2019 18:00	01/01/2019 21:00	0
01/01/2019 19:00	01/01/2019 22:00	0
01/01/2019 20:00	01/01/2019 23:00	0
01/01/2019 21:00	02/01/2019 00:00	0
01/01/2019 22:00	02/01/2019 01:00	0
01/01/2019 23:00	02/01/2019 02:00	0
02/01/2019 00:00	02/01/2019 03:00	0
02/01/2019 01:00	02/01/2019 04:00	0

local_time	electricity
01/01/2020 00:00	0
01/01/2020 01:00	0
01/01/2020 02:00	0
01/01/2020 03:00	0
01/01/2020 04:00	0
01/01/2020 05:00	0
01/01/2020 06:00	0.003
01/01/2020 07:00	0.054
01/01/2020 08:00	0.191
01/01/2020 09:00	0.351
01/01/2020 10:00	0.473
01/01/2020 11:00	0.547
01/01/2020 12:00	0.575
01/01/2020 13:00	0.558
01/01/2020 14:00	0.496
01/01/2020 15:00	0.390
01/01/2020 16:00	0.245
01/01/2020 17:00	0.089
01/01/2020 18:00	0.015
01/01/2020 19:00	0
01/01/2020 20:00	0
01/01/2020 21:00	0
01/01/2020 22:00	0
01/01/2020 23:00	0
02/01/2020 00:00	0
02/01/2020 01:00	0

Input data for MicroGridsPy

Sources for images: [Renewables.ninja](https://renewables.ninja/) website

Return to MicroGridsPy and proceed with the simulation, using the 'Next' button, now incorporating your exogenous RES data.

## Note

Please be aware that the filenames within the Code folder of MicroGridsPy, especially input files like 'RES\_Time\_Series.csv', must remain unchanged. The software's code references these files by name, and altering them can disrupt the ability to correctly access and process the necessary data. This is a crucial aspect of the model's structure for accurate and efficient functioning.

Once set the described (exemplative) scenario, run the model while keeping other parameters that are NOT affected by this new scenario (not necessary all) unchanged. Refer to the previous hands-on guides for detailed steps on running the model.

After the optimization process, take note of key results and the relevant plots generated by MicroGridsPy.

## Managing csv files

When working with CSV (Comma-Separated Values) files downloaded from sources like Renewables.ninja, you may need to properly format or manipulate the data in a spreadsheet application like Microsoft Excel, or using a simple text editor. Here are the steps and tips for doing so:

### Separating Columns in Excel

1. Open the CSV File in Excel:
  - Open Excel
  - Go to File > Open
  - Navigate to where your CSV file is saved and select it
  - Excel might automatically detect and separate the columns. If not, proceed to the next steps
2. Use Text to Columns Feature:
  - Click on the column that contains your combined data
  - Go to the Data tab in the ribbon
  - Click on Text to Columns
  - Choose Delimited > Click Next
  - Check Comma as your delimiter. Other options like Tab, Semicolon, etc., can be unchecked
  - Click Next, then Finish

### Modifying CSV File for Decimal Formats

Problem: Different regions use different decimal separators (e.g., some use a comma ,, others use a period .). This can cause issues when importing data into certain programs.

Solution with Text Editor:

1. Open the CSV file in a text editor that supports "find and replace" (e.g., Notepad++, Sublime Text).
2. Use the "find and replace" feature:
  - Find "," and replace it with "." if you need to change the decimal separator from a comma to a period. Alternatively, find "." and replace it with "," for the opposite
3. Save the file after making the changes.



### **Tip for Renewable.ninja Specific CSV Files**

If the file RES\_Time\_series.csv requires commas for float numbers, ensure that your regional settings in Excel or your text editor match this requirement. For instance, if your Excel is set to use a period as the decimal separator, it might incorrectly interpret the data. The "find and replace" method in a text editor can be particularly handy in such situations to ensure the file adheres to the required format.

Remember, when using "find and replace," be cautious as it can unintentionally change data if not used precisely (for example, changing commas in text fields). Always make a backup of your original file before performing such operations.

Well done on completing the hands-on 4!