Computer practical assignment: Utrecht smart district

Part A: Building-integrated photovoltaics (BIPV) - Irradiance and PV performance modelling

Course: Energy in the Built Environment (GEO4-2522) 2021/2022

Date/Version: Sep. 1st, 2021

Contributors/Editors

Course coordinator: dr. ir. Ioannis Lampropoulos (i.lampropoulos@uu.nl)

Developer of Part A (Building-integrated photovoltaics): Energy in the Built Environment (GEO4-2522) course team

Tutorial instructors: Nico Brinkel M.Sc. (n.b.g.brinkel@uu.nl) and Thomas de Bruin (t.a.debruin@uu.nl)

Abstract

The tutorials will address the employment of programming and optimisation tools for solving a computer practical assignment. Through this practical assignment, the students are introduced to the concept of distributed generation in the built environment and demand side management for residential customers. The students will perform the computer practical assignment in groups by using Python programming. The assignment consists of two parts:

Part A: Building-integrated photovoltaics (BIPV) - Irradiance and PV performance modelling

Part B: Optimal Home Energy Management (HEM) - Multi-objective optimisation of home energy management

After successfully completing the assignment (both Part A & B), the students are expected to be able to apply Python programming and optimisation tools to solve a practical problem addressing performance modelling of distributed energy resources and energy management in buildings, and to report their results and findings in a clear and understandable manner. In this manual, there are a number of questions, including sub-questions that shall be part of the group deliverable report.

Recommended pre-requisites and self-study material

Completed Python programming online course (it is highly recommended to the students to complete the the Sololearn course: https://www.sololearn.com/learning/1073). See Section 2.1. of the 'Introduction to Python Programming' manual.

Time schedule

The tutorials will take place on Tuesdays and Thursdays including the lectures that are specifically addressing programming techniques with Python (an introduction to Python programming, and optimisation techniques and Python tools), and an introduction to the computer practical assignments. For the detailed time-schedule, please check the course guide. During the tutorials, instructions will be provided about how to complete the assignment, and students will be able to question and/or discuss any issues with the tutorial instructors.

Deadlines

- **Submission of group draft report for Part A by October 1st, 2021 at 17:00**. Instructions about the submission will be provided via Blackboard before the deadline.
- **Submission of peer-review feedback for Part A by October 6th, 2021 at 17:00**. Each group will review the draft report and results of their fellow students and provide feedback. Specific guidelines for the peer-review process will be provided via Blackboard before the deadline for the submission of the reports. The responsible tutorial instructor will check the received peer-review comments, add any additional comments and provide any additional comments to the students prior to the final report submission deadline (at least 2 days in advance).
- **Submission of group final report for Part A by October 13th, 2021 at 12:00**. Instructions about the submission will be provided via Blackboard before the deadline.

Table of Contents

1. Part A: Building-integrated photovoltaics (BiPV)

1.1 Introduction

For this assignment, we will model the irradiance on different building façades. The calculated irradiance datasets will then be used to model the performance of several types of BIPV modules at these different building façades. The buildings for which we will model irradiance are shown in **Figure 1**:

- One office building, oriented on a NorthEast-SouthWest line (A)
  - Height: 100m
  - Width: 50m
  - Length: 60m

- Another office building, oriented on a North-South line (B)
  - Height: 30 m
  - Width: 50 m
  - Length: 30 m

- Two rows of houses, one on a North-South line (D), one on an East-West line (C)
  - Height of vertical façade: 6 m
  - Total height: 9 m
  - Roof incline: 40 degrees
  - Length of one row: 50 m

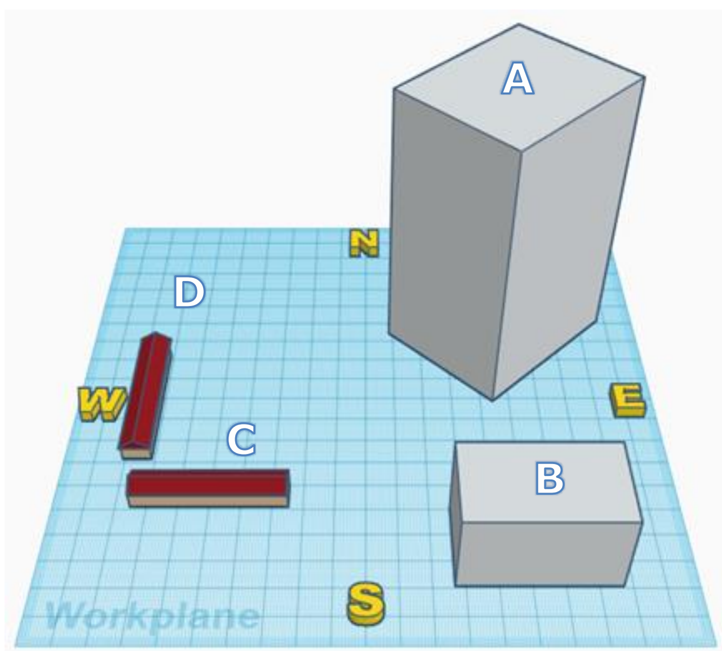

Figure 1. Illustration of different building façades for modelling the irradiance.

1.2 Learning objectives

- To use python to read and process datasets.
- To use data analysis to determine the most effective irradiance model.
- To perform modelling of solar irradiance on different tilted and oriented building surfaces.
- To perform modelling of the performance of PV systems on these building surfaces.
- To demonstrate how the energy yield of PV modules changes when the orientation and tilt of the modules is changed.

1.3 Irradiance Modelling

Provided Data

In the excel/csv file on Blackboard (Irradiance_2015_UPOT.csv) you can find a solar irradiance dataset that is measured at Utrecht Photovoltaic Outdoor Test facility (UPOT), at the Utrecht Science Park (USP), with a high time resolution (30 seconds). The dataset contains the following parameters:

- Global horizontal irradiance (GHI)
- Air temperature
- Direct Normal Irradiance (DNI)
- Diffuse Horizontal Irradiance (DHI)

Composition of Solar Irradiance

As measuring term of the solar irradiance on earth's surface the Global Horizontal Irradiance (GHI, see also this link) is used. It is measured by a pyranometer placed parallel to earth's surface (tilt=0°) and it is expressed in Watt or Watt-hours per square meter [W/m^2 or Wh/m^2], depending on the time interval or the situation. GHI consist of three components, the Direct Normal Irradiance (DNI, see also this link), the Diffuse Horizontal Irradiance (DHI, see also this link) (note the different meaning of the letter D in the two abbreviations) and the ground reflected radiation.

DNI is the direct irradiance from the sun, while DHI is the radiation reaching the earth's surface after having been scattered from the DNI by molecules or particulates in the atmosphere. However, because ground-reflected radiation is usually insignificant compared to DNI and DHI, for all practical purposes global horizontal radiation is assumed to be the sum of direct and diffuse radiation and it is expressed by the following formula:

$$GHI = DHI + DNI * \cos(Z)$$

where Z is the solar Zenith angle (see this link).

Analysis of Solar irradiance related to PV systems.

Data analysis of solar irradiance is the most important part in any study for a new solar PV system installation, either financial or environmental. Solar irradiance analysis takes place in order to be determined the most effective combination of tilt and orientation angles, based on the electricity production. For that reason the knowledge of the Plane of Array irradiance (POA, see this link) of any surface is important (in other literature it can be also named Global Tilted Irradiance (GTI)). Since measured data of any POA for any surface is impossible to be found, the POA has to be calculated through solar models and historical solar irradiance data.

The most common solar irradiance data one can find in the Netherlands is the GHI, provided for free by KNMI, and it can be downloaded from their website, along with more meteorological data.

Models are used to split the measured GHI into its components, DNI and DHI. The impact of the components on the titled surface is calculated and then summed up to calculate the POA. For question 1 and 2 you will follow these steps to calculate the POA on the surfaces that you need to install the PV systems on. Since you do not know which model is the best in order to split the GHI into its components, you will use the data from UPOT in order to determine the best model.

Question 1

In the Python *pvlib* module, there are four different models that can calculate the DNI component of GHI: DISC, DIRINT, DIRINDEX and Erbs. You can find these models under **pvlib.irradiance** (check the *pvlib* documentation to see which inputs are required for each one of these models).

Since all these models are empirical and developed in specific locations with different climate characteristics than the Netherlands, you have to determine the best one for local use. Since you have access to UPOT data, the testing facility of Utrecht University, where GHI, DNI and DHI data are individually measured, you can apply the models on the GHI and compare the modelled DNIs with the measured ones, in order to determine the best model for use in the Netherlands.

In order to calculate all the steps below you need to know the solar position, and specifically the solar zenith. You can calculate the solar position with the function **pvlib.solarposition.ephemeris()**, which takes as inputs time (e.g. **upot_data.index**), latitude, longitude, pressure and temperature. You can leave pressure at the default value.

1. Use GHI of the UPOT dataset to calculate the DNI with the four different models. Describe in your report what data preparation steps you performed.
   Note: if you find that one model creates very unexpected and illogical values, dive into your data to find the cause!

2. The most accurate model can be chosen using several error functions:
   - RMSE (root mean square error: $RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}}$, where $\hat{y}$ are modelled and y are the measured values)
   - MBE: the mean of errors $\hat{y}_i - y_i$, indicating if the model is on average biased positive or negative
   - MAE: the mean absolute error
   - R^2: the coefficient of determination (use e.g. scipy.stats.linregress(x, y) or r2_score from sklearn.metrics)

Which error function is most appropriate in this case? Discuss and compare the different error functions in your report.

3. Compare the calculated DNI values to the original measured data. Make a scatterplot of the modelled versus measured DNI irradiance for each model. Use appropriate line styles, axis labels, font sizes, etc. and make use of the subplot-function. Include the figures in your report. Which model is most accurate? Choose this model for further calculations.

Question 2

Now that you defined the most suitable model for the Netherlands, you are able to model the irradiance on the building surfaces, described in the introduction of the assignment (see also **Figure 1**). For most of the cases (façades of A & B, rooftops of C & D) there is a single option for the tilt and orientation. However, for the buildings A & B, some tilt and orientation values for the systems on the rooftops have to be decided based on modelling. For the building A this is both tilt and orientation, while for the building B it is only the tilt.

1. On the KNMI website, download the GHI (Globale straling), wind speed and temperature of a station of your choice, for the year 2019 (Jan. 1st to Dec. 31st). Based on the example in the provided video on Blackboard, fix the data in a proper form (create a datetime index, convert the irradiance data to W/m^2 etc.). Note: exclude the 'de Bilt' station and ensure that the station you selected for download actually provides data since some stations might not have available data. Also, think carefully about what timestamp is most appropriate for the provided data.
2. Determine the slopes (tilts) and orientations (surface azimuth) of the following building surfaces:
    a. Rooftops of C & D
    b. Façades of A & B (exclude façades facing North, NE & NW)

   For the report, create an overview table, showing the tilt and the orientation of each surface. For the houses, the vertical surfaces (façades) will not be covered, only the roofs. **Note:** Do not include the table in this part since you will fill it in later.
3. Using the function **pvlib.irradiance.get_total_irrad()**, calculate the POA irradiances[1], POA_total, POA_diffuse, POA_direct on each of the building surfaces of sub-question 2. As latitude and longitude values, pick a place close to your chosen KNMI station.

   Note: Do not use separate variables for each surface. Try to use functions, for-loops and dictionaries in order to model this efficiently. Check the video on Blackboard on dictionaries and ask the teaching staff for support here!

4. Now it is time to determine the best tilt and orientation for the non-obvious surfaces, thus the rooftops of A & B.
    a. Use the same process, used in sub-question 3
    b. For rooftop B, calculate the POA_total for a range of different tilts (from 10 to 45, with step of 5, thus 7 values)
    c. For rooftop A calculate the POA_total for the same tilt range and also different orientation cases. Take the two cases, SE(135°) and SW(225°). Thus 14 values in total.
    d. For each rooftop, present the annual irradiance sums of POA_total in a bar-chart (thus, 2 bar-charts, one with the 7 scenarios of rooftop B and one with the 14 scenarios of rooftop A). Choose the best scenario of each building.

---

[1] *In this question you will need to present only the POA_total, in plots at sub-questions 4 and 6. However you will need the rest POA values later, thus it is better to calculate them altogether from the . get_total_irrad() function.*

   e. For the report include only the bar-chart for each building and your comments for the best combination for each building surface.

5. Now that you know the best combination for tilt and orientation for rooftops of A and B, add them in the table of sub-question 2. Now you can add the table in the report. Furthermore, calculate the POA_total, POA_diffuse and POA_direct also for these surfaces.
6. Compare the annual POA_total irradiance sums (in kWh/m^2) for all the building surfaces in a bar-chart. Which building surface has the highest annual irradiance? Shortly discuss your results, about the impact of the tilt and orientation.

1.4 PV performance modelling

In this part of the assignment, you will model the performance of three types of PV modules on the different building surfaces.

Assume the following:

- The office façades can be covered by 30% with PV modules. Assume these modules to be integrated into the façade (BIPV).
- For the office rooftops, assume that the area of the systems will be equal to the 50% of the area of the rooftop. Assume these to be mounted on top of the existing roof with the proper tilt.
- For the houses, only the roofs (for 60%) can be used for PV modules. Assume these to be mounted on top of the existing roof, i.e. Building Attached Photovoltaics (BAPV).
- The PV modules we will compare are:
  - Sanyo HIT module
  - Yingli mono c-Si module
  - FirstSolar CdTe thin-film module

Question 3

Load the module parameters (Module_Parameters.xlsx) in your script with **pandas.read_excel()**. Check the data-frame at the variable explorer to verify it is correctly loaded (tip: the "Parameters" column should be index, capacity of one module is the value Wp). The DC output of a module can be modelled with a quite complex chain of functions. The basis is the function **pvlib.temperature.sapm_cell()**, which returns the values for one module. See **Figure 2** at the end of the assignment for an overview of the modelling steps, or check the 'Intro Tutorial' in the following link: https://pvlib-python.readthedocs.io/en/stable/introtutorial.html

1. Calculate the available installation area for each surface. For the three module types, calculate the possible installed capacity (separately for each surface), and the number of modules required (again for each surface).

2. Model the DC performance of the three PV module types (for a single module) on the different building surfaces. Make sure the correct PV cell temperature model is chosen according to the different installation types.

3. Which surface has the highest annual yield (for each module type)? Plot the annual sum of each surface for each module to check and save the figure(s). **Tip:** you can make one plot per surface, with all the modules, or one plot per module with all the surfaces. What is the best way to present the results to a third party? Other options are more than welcome.

4. Which module technology has the highest energy yield per unit of façade area? Is this the case for all the façades? Pick your "best" technology for each façade for further calculations. For the report, present them in a table, where you can include surface name, best module, total capacity, tilt and orientation. This is a typical static data table for PV systems.

Question 4

Herein you will model the total power output of the PV systems across the different buildings. To convert DC to AC power, use a simple inverter model based on NREL's PVWatts model:

$$\eta(P_{dc}) = -0.0162 \cdot \zeta - \frac{0.0059}{\zeta} + 0.9858 \quad \text{where} \quad \zeta = \frac{P_{dc}}{P_{dc0}} \quad \text{and} \quad P_{dc0} = \frac{P_{ac0}}{\eta_{nom}}$$

where P_{ac0} is the rated power of the inverter and η_{nom} is the nominal efficiency of the inverter. Here, we assume the inverter to be sized equally to the DC power of the system (thus P_{ac0} is equal to the rated DC power of the PV system) and the nominal efficiency of the inverter to be $\eta_{nom} = 0.96$.

The AC power output of the system is given by:

$$P_{ac} \begin{cases} \eta(P_{dc})P_{dc} & : 0 < P_{dc} < P_{dc0} \\ P_{ac0} & : P_{dc} \geq P_{dc0} \\ 0 & : P_{dc} = 0 \end{cases}$$

Define your own function in python that implements this inverter model. See the Python manual 'Introduction to Python programming' for more information about how to do this (or check the videos uploaded in Blackboard).

To analyse the impact of tilt and orientation on PV system output during the day, select 3 days based on irradiance data. The day of the higher irradiance during summer (Jun. - Aug.), similar for the spring (Mar. – May) and autumn (Sep. – Nov.).

1. Model the AC power of each surface, only for the "best" module.

2. Provide a bar-chart with the yearly AC energy generated from each façade and one with the total of each building

3. For each building make three plots, one for each day. Each façade should be represented with a different line

4. How is the orientation affecting the production during the day? What differences do you notice based on the period? If the chosen dates are very close (e.g. 23 May and June 10), maybe you should pick different days, with high irradiance but with some distance from each other.

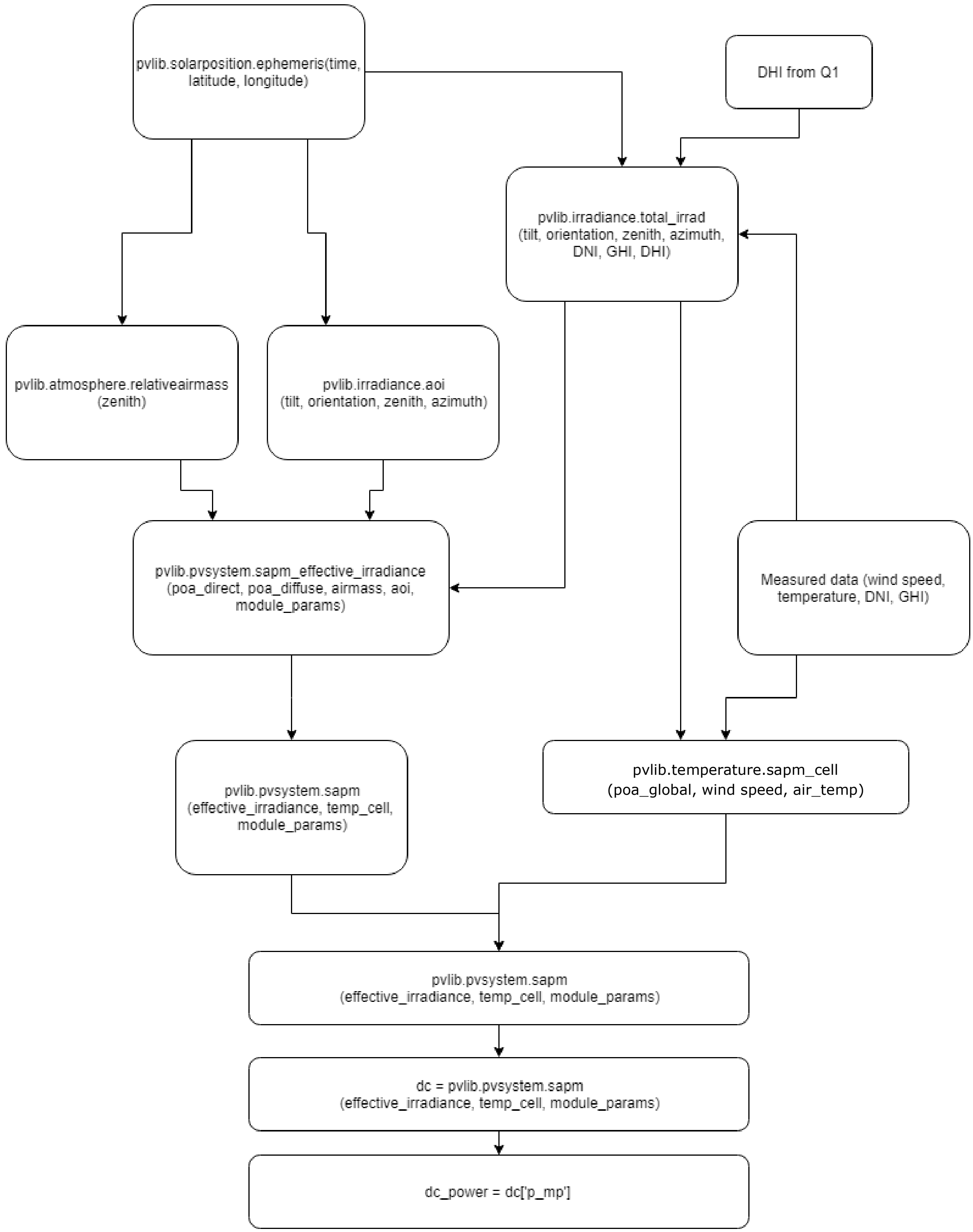

Figure 2. Overview of the modelling steps.