Shell.ai Hackathon for Sustainable and Affordable Energy

Windfarm Layout Optimisation Challenge

AEP Algorithm

We use the separately described method of wake modeling in the provided wind farm evaluator codes for calculating the AEP of a turbine layout. In this section, we break down the algorithm inside these codes. We suggest to follow along the Python file `Farm_Evaluator.py`.

1. Read input files.

Implemented in the function - `getTurbLoc`, `loadPowerCurve`, `binWindResourceData`

- `turbine_loc_test.csv`. (x, y) locations of 50 turbines.
- `power_curve.csv`. Power and thrust coefficient data.
- `wind_data_<year>.csv`. Wind data.

2. Construct wind instances and calculate their probabilities.

Implemented in the function - `binWindResourceData`

We first need to 'discretize' the entire wind resource data into small *wind instances*. To do this, we bin wind direction and speed values into bins of sizes 10° and 2 m/s respectively and 'trap' number of data points inside these binned wind instances. This helps us to estimate the probability of occurrence of these wind instances, which we do by dividing the number of data points 'trapped' by the total number of data points. We denote the probability of occurrence of j^{th} wind instance by p_j We present below the discretized wind instances in a tabular format below with their corresponding direction and speed bins (denoted by s).

	0<=s<2	2<=s<4	....	....	26<=s<28	28<=s<30
drct = 360	..	..	..	..	..	..
drct = 10	..	..	..	..	..	..
drct = 20	..	..	..	..	..	..
....	..	..	..	..	..	..
drct = 340	..	..	..	..	..	..
drct = 350	..	..	..	..	..	..

We remind that that entries in `drct` column of the wind data provided are in multiples of 10 i.e. 360°, 10°, 20°. . . . 340°, 350°. 360° and 0° are one and the same thing. We have in total 540 wind instances.

$$\sum_{j=1}^{540} p_j = 1.0$$

> *Note that the direction and speed bin sizes chosen here are coarser than the desired resolution degree. We have chosen their sizes for computational reasons while not compromising heavily on wake effect modeling.*

3. Iterate over the wind instances.

Step 1. Rotate the frame of reference according to the wind flow direction. Implemented in the function - `rotatedFrame`

For the given wind flow direction, θ (in radians), convert the euclidean turbines (x, y) coordinates to *downwind-crosswind* coordinates, (x', y'). The shift is done so that the wind flow direction aligns with the positive x-axis.

$$x' = x\cos\left(\theta - \frac{\pi}{2}\right) - y\sin\left(\theta - \frac{\pi}{2}\right)$$
$$y' = x\sin\left(\theta - \frac{\pi}{2}\right) + y\cos\left(\theta - \frac{\pi}{2}\right)$$

Step 2. Calculate effective wind speed at each turbine location using Jensen's PARK model. Implemented in the function - `jensenParkWake`

For the given free wind speed V_∞, now calculate the speed deficit experienced by a given turbine using Eq. (1) and (2) of the PARK model. The effective wind speed (V_{eff}) at each turbine location can then be computed as:

$$V_{eff} = V_\infty\left[1 - \left(\frac{\Delta V}{V_\infty}\right)_{\text{total}}\right]$$

Step 3. Estimate power production by the wind farm for this particular wind instance. Implemented in the function - `partAEP`

Use V_{eff} and power curve data from `power_curve.csv` to estimate the power produced by each turbine and obtain their sum. We denote the power produced by the wind farm for this particular j^{th} wind instance as P_j

Step 4. Repeat steps **Step 1.** to **Step 3.** for the next wind instance and record the respective P_j.

4. Calculate Wind Farm AEP. Implemented in the function - `totalAEP`

Multiply P_j with the corresponding frequency of occurrence of the ,wind instance (p_j). We provide the formula below for 540 wind instances. It is multiplied by a factor of 8760, which is the total number of hours in a year. A factor of 10^3 in the denominator converts the power to gigawatt-hours (GWh).

$$AEP = \frac{8760}{10^3}\left(\sum_{j=1}^{540} p_j P_j\right)$$