
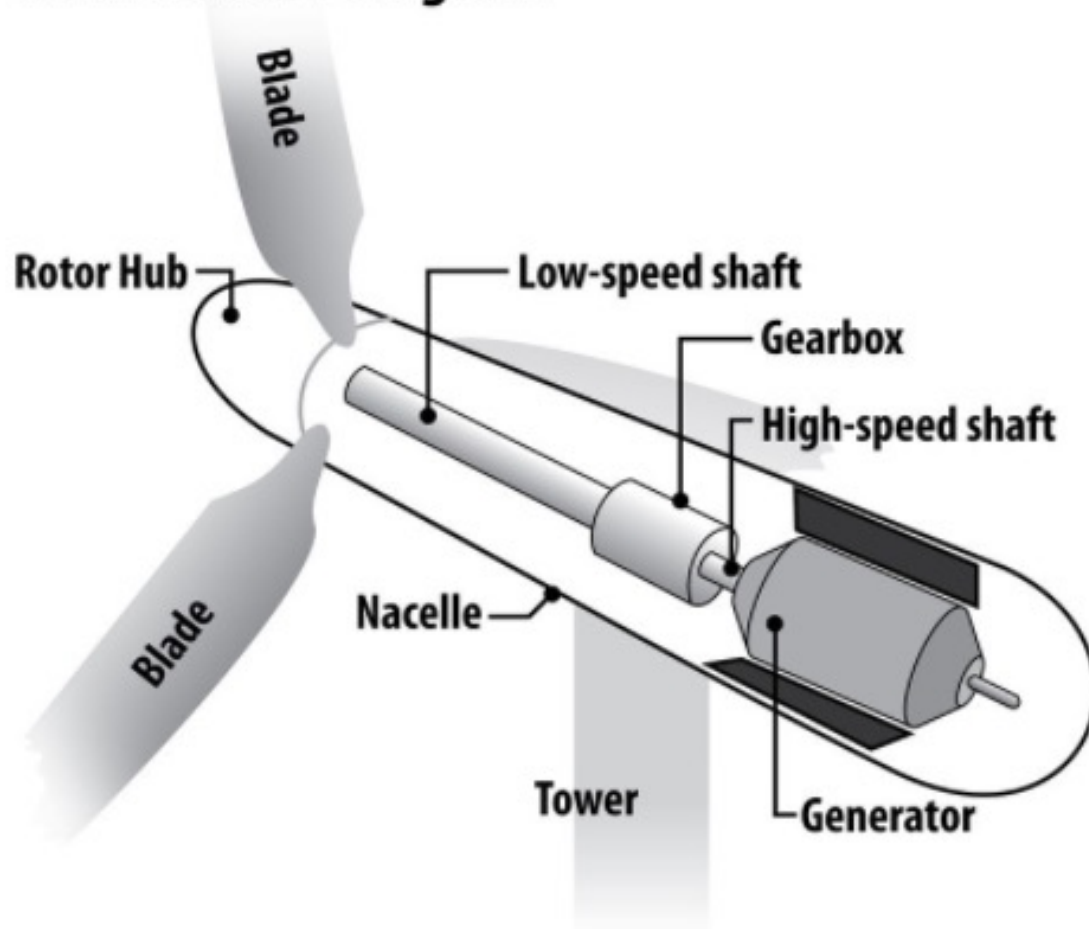
A landscape of wind turbines on a hillside at sunrise or sunset, with a sea of clouds below. The sky is a mix of purple, blue, and orange, and the sun is low on the horizon. The turbines are white and stand in a line across the hillside. The clouds are thick and white, filling the valleys and creating a sea of clouds effect. The overall scene is serene and beautiful.

The role of computer programming in wind energy

By: Kara White

A white circle graphic located in the bottom right corner of the page.

Wind Turbine Diagram




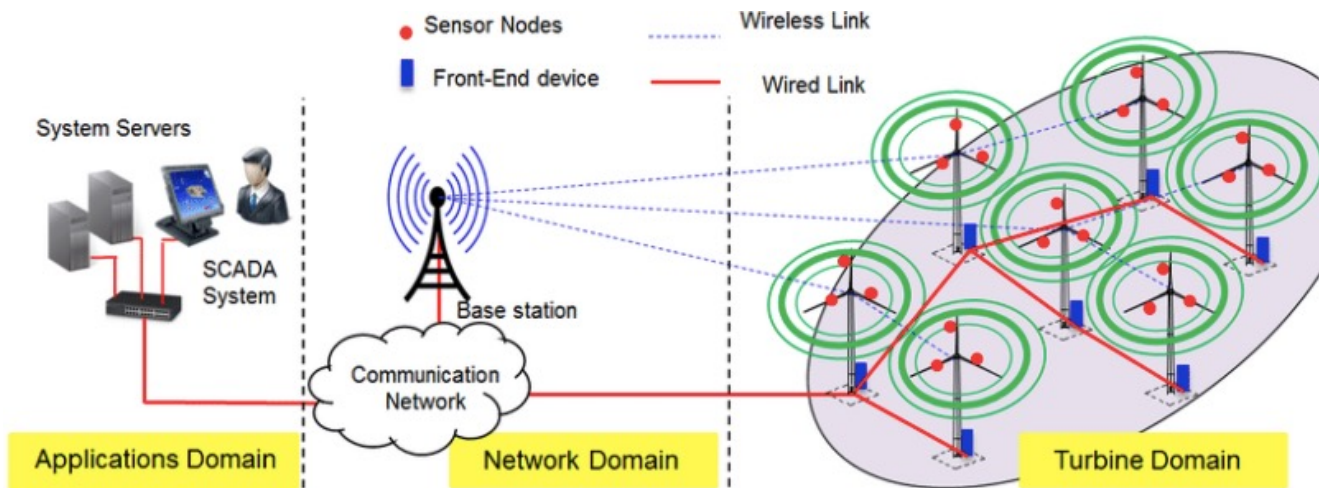
How does a wind turbine work?

- Wind turns blades around a rotor
- Rotor spins a generator and creates kinetic energy
- DC electricity is generated
- Electricity is converted to AC via inverter before entering the grid (or a home)



How is programming used for wind energy?

- Design turbine blades for maximum aerodynamic performance
 - Prevent the wind turbine from exceeding safe max speed
 - To determine Rated Power Capacity
- 



Supervisory Control & Data Acquisition (SCADA)

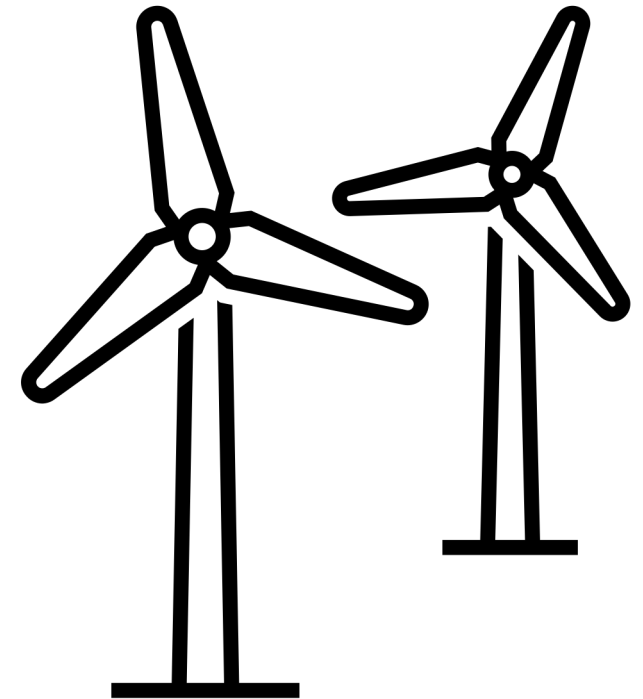


Using programming to build smarter wind energy

- "The National Wind Technology Center (NWTC) at NREL develops advanced computer-aided engineering (CAE) tools to support the wind power industries with state-of-the-art design and analysis capabilities." (NREL)

Airfoil data for aerodynamic performance of blades

- An airfoil is **the foundation of wind turbine blade design**
- AirfoilPrep (NREL)
 - Correcting the angles and direction of the blades for maximum efficiency
 - Uses characteristics of a turbine blade as parameters (weight, pitch (blade angle), drag)
 - Generates airfoil data files from 2-D data
 - Adjusts 2-D data for rotational augmentation (3-D effects)
 - PYTHON CODE (Source: NREL): <https://github.com/WISDEM/AirfoilPreppy/blob/master/airfoilprep/airfoilprep.py>



```
4 airfoilprep.py
5
6 Created by Andrew Ning on 2012-04-16.
7 Copyright (c) NREL. All rights reserved.
8
9
```

```
31
32 class Polar(object):
33     """
34     Defines section lift, drag, and pitching moment coefficients as a
35     function of angle of attack at a particular Reynolds number.
36
37     """
38
39     def __init__(self, Re, alpha, cl, cd, cm):
40         """Constructor
41
42         Parameters
43         -----
44         Re : float
45             Reynolds number
46         alpha : ndarray (deg)
47             angle of attack
48         cl : ndarray
49             lift coefficient
50         cd : ndarray
51             drag coefficient
52         cm : ndarray
53             moment coefficient
54         """
55
56         self.Re = Re
57         self.alpha = np.array(alpha)
58         self.cl = np.array(cl)
59         self.cd = np.array(cd)
60         self.cm = np.array(cm)
61
```

```
4 airfoilprep.py
5
6 Created by Andrew Ning on 2012-04-16.
7 Copyright (c) NREL. All rights reserved.
8
9
```

```
480 import matplotlib.pyplot as plt
481
482 p = self
483
484 figs = []
485
486 # plot cl
487 fig = plt.figure()
488 figs.append(fig)
489 ax = fig.add_subplot(111)
490 plt.plot(p.alpha, p.cl, label='Re = ' + str(p.Re/1e6) + ' million')
491 ax.set_xlabel('angle of attack (deg)')
492 ax.set_ylabel('lift coefficient')
493 ax.legend(loc='best')
494
495 # plot cd
496 fig = plt.figure()
497 figs.append(fig)
498 ax = fig.add_subplot(111)
499 ax.plot(p.alpha, p.cd, label='Re = ' + str(p.Re/1e6) + ' million')
500 ax.set_xlabel('angle of attack (deg)')
501 ax.set_ylabel('drag coefficient')
502 ax.legend(loc='best')
503
504 # plot cm
505 fig = plt.figure()
506 figs.append(fig)
507 ax = fig.add_subplot(111)
508 ax.plot(p.alpha, p.cm, label='Re = ' + str(p.Re/1e6) + ' million')
509 ax.set_xlabel('angle of attack (deg)')
510 ax.set_ylabel('moment coefficient')
511 ax.legend(loc='best')
512
513 return figs
```



```
4 airfoilprep.py
5
6 Created by Andrew Ning on 2012-04-16.
7 Copyright (c) NREL. All rights reserved.
8
```

```
515
516 class Airfoil(object):
517     """A collection of Polar objects at different Reynolds numbers
518
519     """
520
521     def __init__(self, polars):
522         """Constructor
523
524         Parameters
525         -----
526         polars : list(Polar)
527             list of Polar objects
528
529         """
530
531         # sort by Reynolds number
532         self.polars = sorted(polars, key=lambda p: p.Re)
533
534         # save type of polar we are using
535         self.polar_type = polars[0].__class__
536
537
```

```
4 airfoilprep.py
5
6 Created by Andrew Ning on 2012-04-16.
7 Copyright (c) NREL. All rights reserved.
8
9
```

```
106
107     def correction3D(self, r_over_R, chord_over_r, tsr, alpha_max_corr=30,
108                     alpha_linear_min=-5, alpha_linear_max=5):
109         """Applies 3-D corrections for rotating sections from the 2-D data.
110
111         Parameters
112         -----
113         r_over_R : float
114             local radial position / rotor radius
115         chord_over_r : float
116             local chord length / local radial location
117         tsr : float
118             tip-speed ratio
119         alpha_max_corr : float, optional (deg)
120             maximum angle of attack to apply full correction
121         alpha_linear_min : float, optional (deg)
122             angle of attack where linear portion of lift curve slope begins
123         alpha_linear_max : float, optional (deg)
124             angle of attack where linear portion of lift curve slope ends
125
126         Returns
127         -----
128         polar : Polar
129             A new Polar object corrected for 3-D effects
```

What aspects of this code have we already worked with?



Loops and
conditions



2-D arrays and
Numpy



Data analysis



Functions



Classes and
Objects

Works Cited

<https://www.earthtechproducts.com/faq-wind-power.html>

<https://www.ni.com/en-us/innovations/white-papers/08/wind-turbine-control-methods.html>

<https://www.intechopen.com/chapters/59756>

<https://www.axiomtek.com/Default.aspx?MenuId=Solutions&FunctionId=SolutionView&ItemId=45&Title=Wind+Power+Automation><https://sites.psu.edu/sustainabilitycis/c-project-wind-power/>

<https://www.forbes.com/sites/quora/2017/12/15/what-role-can-computer-scientists-play-in-reducing-the-effects-of-climate-change/?sh=467714616758>

<https://www.nrel.gov/wind/computer-aided-engineering-tools.html>

<https://phys.org/news/2011-07-optimizes-farm.html>

<https://www.sciencedaily.com/releases/2020/04/200414095737.htm>

<https://www.hindawi.com/journals/sv/2019/6405153/>