Simulating Transmission Power Infrastructural System Performance during Hurricanes with a Physics-based Data Generation Engine: A Package Installation Instruction (Mac OS)

Xiao Zhu[1], Fatemehalsadat Jafarishiadeh[2]

Ge Ou[1], Mostafa Sahraei-Ardakani[2], and Zhaoxia Pu[3]

Under extreme events, the resilience analysis of regional power systems is prevalent due to the crippling social impacts and substantial economic losses caused by bulk power outages. The study of a hurricane's impact on the power system infrastructure helps find the system's weak links and helps clarify the efficient restoration and recovery procedures. To evaluate the performance of the power systems under hurricane events, we have developed a data generation engine using Python programming language. An optimization solver is used to deal with a power system network change and dispatch the electrical loading (IBM CPLEX). Therefore, this package installation instruction provides guidelines for preparing a software environment to run the data generation engine successfully. This installation manual is for Mac OS users.

Contents

1. Python Programming Language Terminology 2
2. Python Installation 3
3. Python IDEs 4
4. IBM CPLEX Optimization software installation* 6
5. Setup IBM CPLEX Python API* 8
6. Install Python Packages 10

Acknowledgment 10

*For running only the offline version of the software, this step can be skipped.

[1] Department of Civil & Coastal Engineering, University of Florida Gainesville, FL 32611
[2] Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, UT 84112
[3] Department of Atmospheric Sciences, University of Utah, Salt Lake City, UT 84112

1. Python Programming Language Terminology

In this section, a few Python Programming language terminologies will be introduced to help understand the Python basics. The complete Python programming language introduction can be found in Python Introduction.

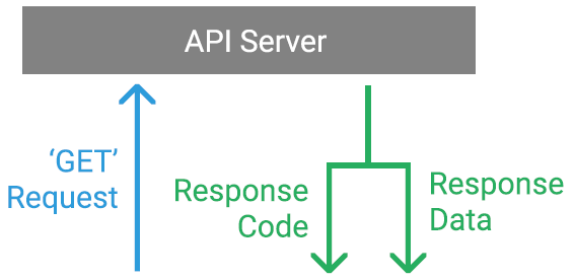
Python version: python.org will release a different version of Python every year. The main functionality is the same, but each version of Python has slightly different functionalities. The Python version is important when some libraries (packages) only support specific versions of Python. If Python has been installed, the version of Python can be checked by:

Open Terminal and enter: Python --version

IDE: IDE is short for Integrated Development Environment. Python built-in IDLE is helpful for running simple codes. But using IDEs, larger and more complex programming projects can be easily handled. In the market, there are lots of free and proprietary IDEs, for example (Pycharm and Spyder). All IDEs can help manage and organize the project, and the preference is based on personal choice.

Virtual Environment: Python virtual environment is a tool used for Python package management and project isolation. Python packages can be installed locally in the virtual environment directory for a particular project.

API: API is short for Application Programming Interface. It is a server that the user can use to retrieve and send data via code. The interaction between the code and API is illustrated in the following figure. The complete Python API introduction can be found in the Python API tutorial: Getting Started with APIs.



2. Python Installation

This software supports Python version Python3.6 or later. This section can be skipped if the user has installed Python version 3.6 later.

- Go to Python.org to download the latest Python version.

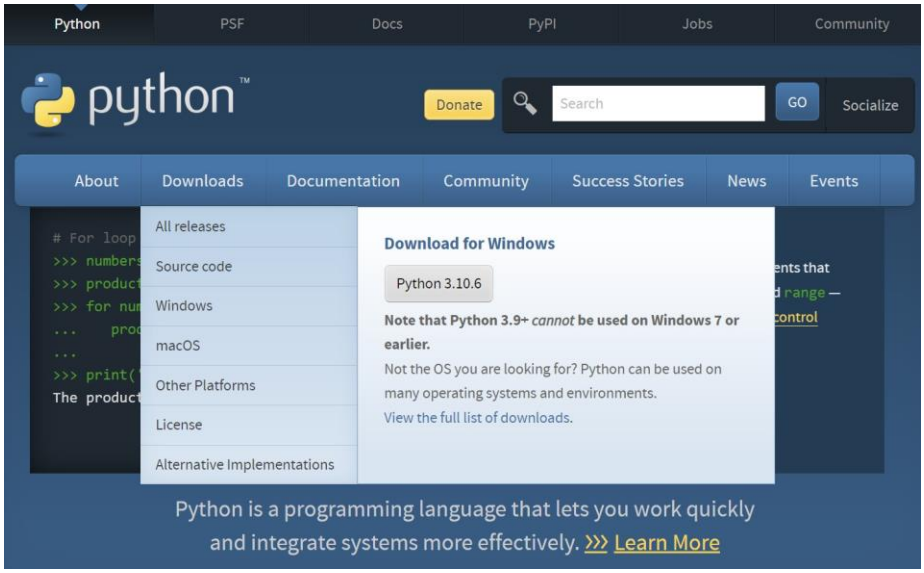


Figure 1. Python Download Page

- Click the downloaded .dmg file for Mac and follow up the default settings.
- Check whether Python has been successfully installed:

  Open Terminal and enter: Python --version

3. Python IDEs

Python IDE is an editor that helps manage large, complex Python projects. In the market, there are lots of free and charged IDEs, for example (Pycharm and Spyder). In this instruction, Pycharm is used as a demonstration for the current and later instructions. For other IDEs, the user can visit their website to know the details about installation.

- Go to Pycharm and click the download button. The download button is on the upper- right of the website.

- On the downloading page, choose which operational system (Windows, Mac, Linux) the current machine is used, and choose the *Community* version.

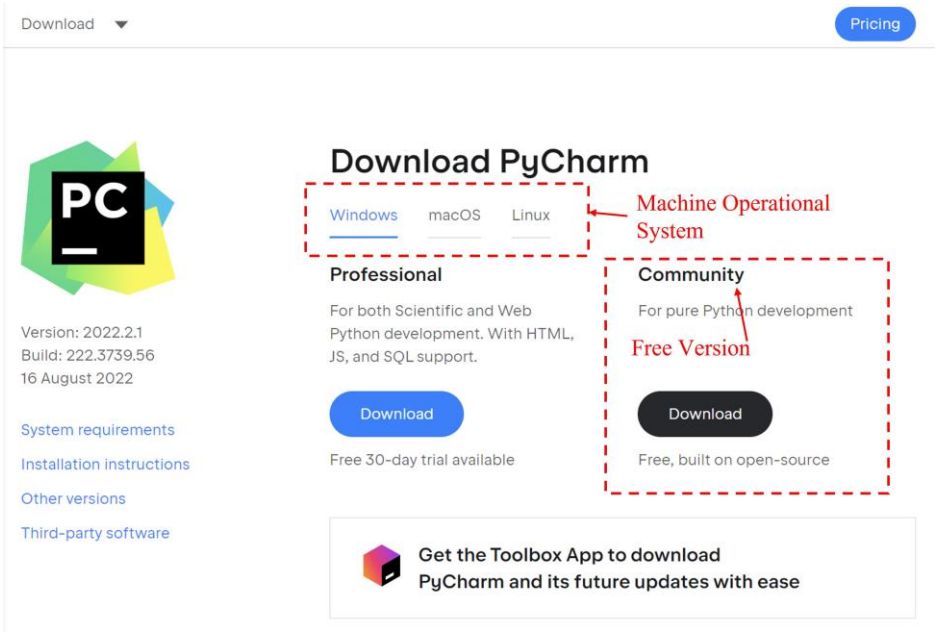


Figure 2. Pycharm Downloading

- Check the installation. If Pycharm has been successfully installed, after opening it, the Pycharm interface will show :

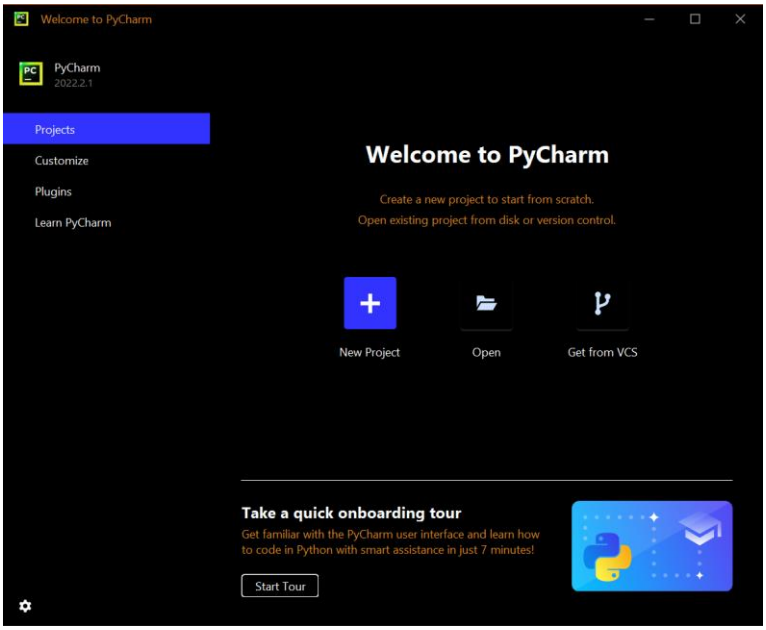


Figure 3. Pycharm Interface

- **Install Anaconda**: In this software, the virtual environment will be created using Anaconda. Therefore, the users need to install Anaconda. The Anaconda downloading page can be found at Anaconda.

4. IBM CPLEX Optimization software installation*

*For running only the offline version of the software, this step can be skipped.

Part of the software aims to find the optimal power dispatch strategy during hurricane events. Therefore, an optimization package is needed to perform such tasks. the IBM CPLEX package is used. The installation of IBM CPLEX is as follows:

- **Java**: Check whether Java is installed on the computer by:
  Open Terminal and enter: Java -version
  If Java is not installed, visit Java to download the latest version.
- IBM CPLEX: Go to IBM CPLEX Optimization Studio's official website.
  - Choose the no-cost academic edition on the website, as shown in the dashed-red box figure below.

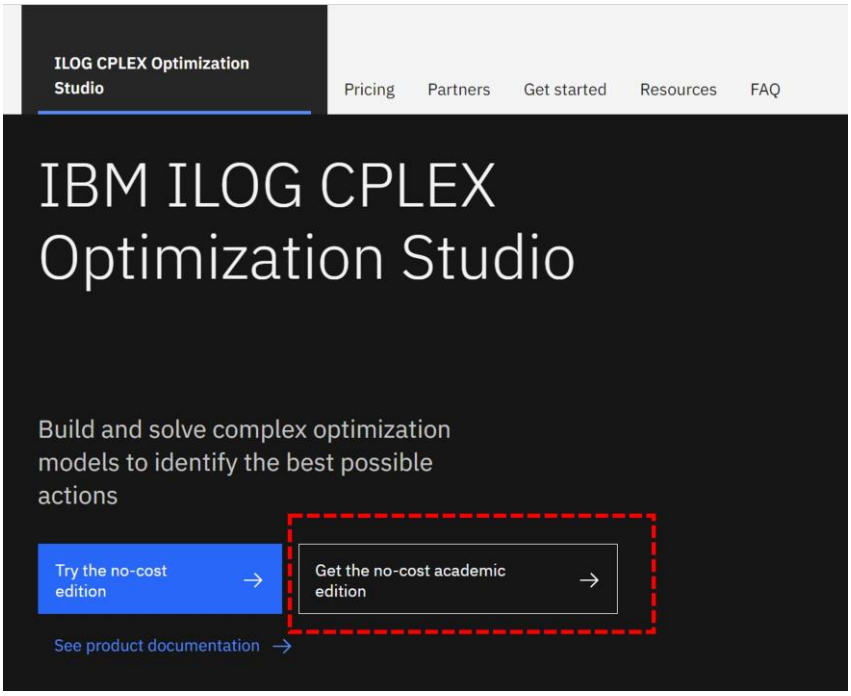


Figure 4. IBM CPLEX Optimization Studio Webpage

  - On the Data Science page, click Login if you have an IBM account. Otherwise, click Register
  - After Login, scroll down to the software option and find ILOG CPLEX Optimization Studio.

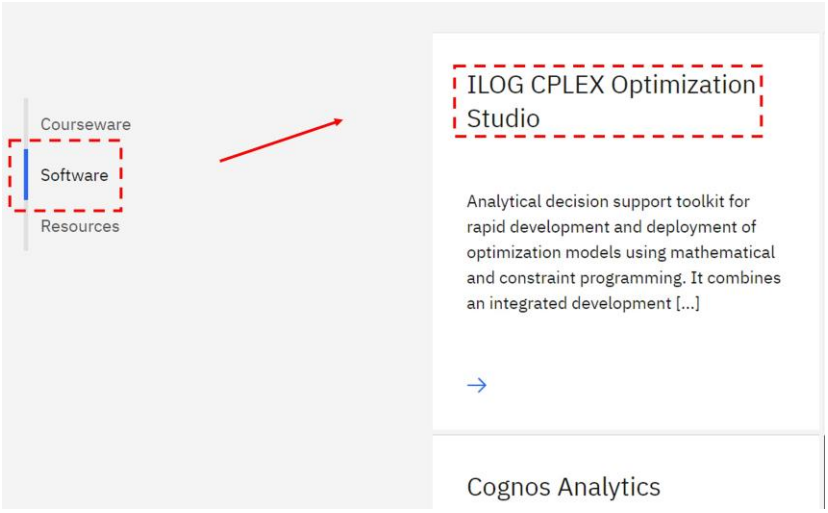


Figure 4. IBM CPLEX Optimization Studio Software Webpage

- Choose Download Director and the current machine operational system version on the downloading page.

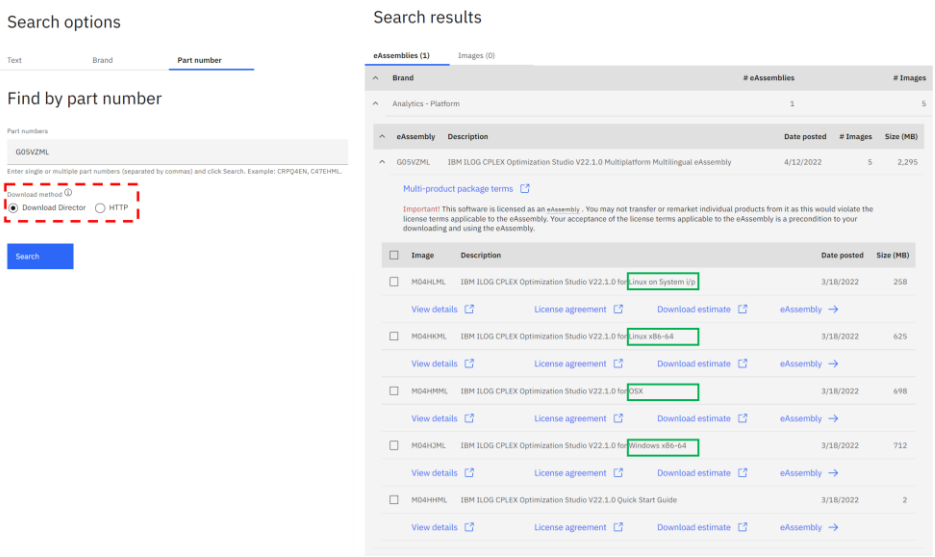


Figure 5. IBM CPLEX Downloading Webpage

- After downloading the IBM CPLEX software, follow the default setting to install. If the user wants to install it non-default, please remember the PATH for the installed CPLEX directory.

5. Setup IBM CPLEX Python API*

*For running only the offline version of the software, this step can be skipped.

Even though IBM CPLEX has been installed, to use this optimizer in Python, the users need to set up CPLEX Python API so that Python can find and use CPLEX. The instructions on setting up IBM CPLEX API is introduced separately for Windows and Mac operating system.

Creating Virtual Environment

- Open Pycharm → Open where the Python scripts are stored

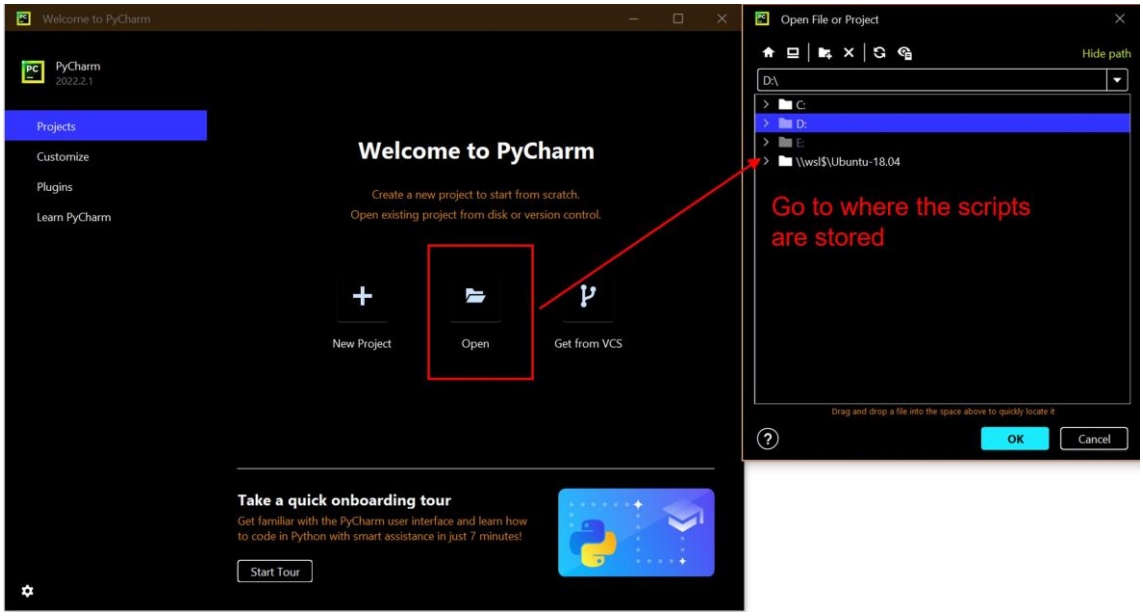


Figure 6. Creating Virtual Environments

- Open setting in Pycharm ribbon → Settings → Interpreter → Add Local Interpreter → Conda Environment. In the Conda Environment, select the Python version later than 3.6 and click Ok.

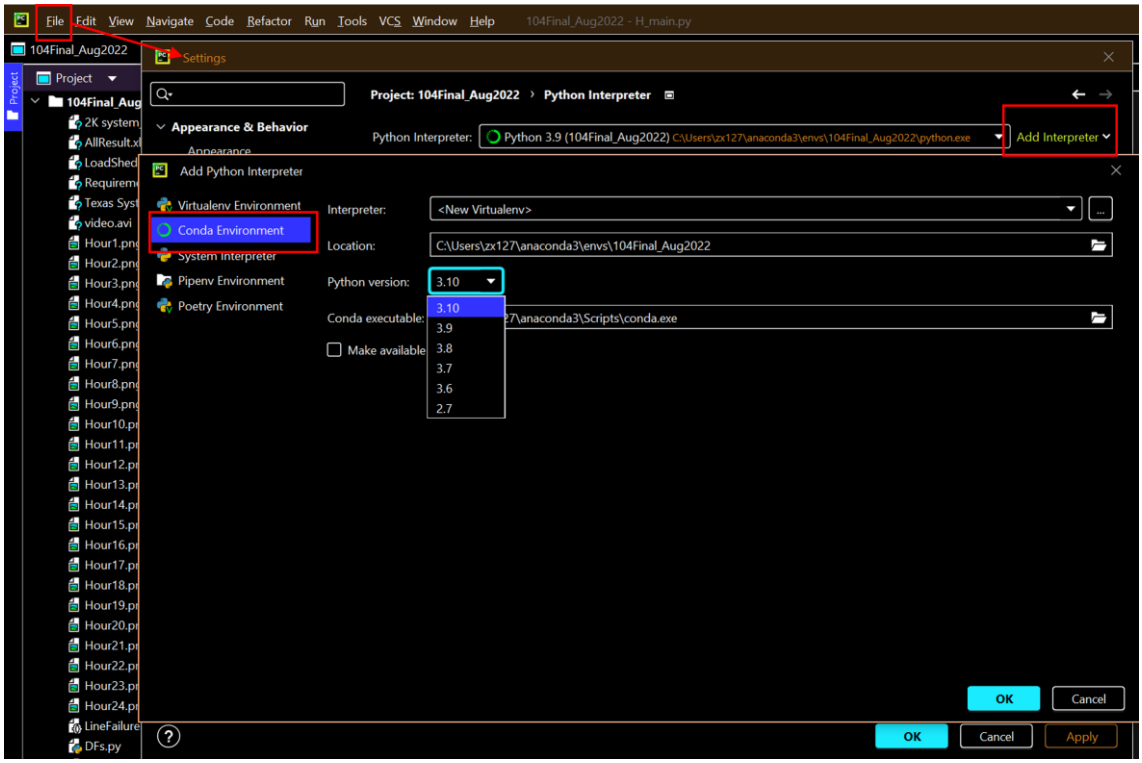


Figure 7. Creating Virtual Environments

Setup CPLEX Python API

- Open Terminal and enter:

```
conda env list
conda activate 'user created Virtual Environment name'
```

- If the user follows up on default settings, the setup file should be located in */Application/CPLEX_Studio211/python\python setup.py install*. In Terminal, enter the following commands:

```
cd /Application/CPLEX_Studio221/python
python setup.py install
```

Then the CPLEX API should be installed properly

```
python — -zsh — 98×23
(base) xiaozhu@Xiaos-MacBook-Pro python % conda env list
# conda environments:
#
104Final_Aug2022         /Users/xiaozhu/.conda/envs/104Final_Aug2022
base                  *  /Users/xiaozhu/opt/anaconda3

(base) xiaozhu@Xiaos-MacBook-Pro python % conda activate 104Final_Aug2022
(104Final_Aug2022) xiaozhu@Xiaos-MacBook-Pro python % cd /Applications/CPLEX_Studio221/python
(104Final_Aug2022) xiaozhu@Xiaos-MacBook-Pro python % python setup.py install
```

Figure 8. Setup CPLEX Python API - Successful

6. Install Python Packages

After installing the IBM CPLEX API, the remaining Python packages can be installed automatically by running the following commands in the Terminal:

```
Pip install -r requirements.txt
```


Acknowledgment

The development of this software and user manual was supported by the Cyberinfrastructure for Sustained Scientific Innovation (CSSI) program in National Science Foundation, under award 2004658, Elements: Open Access Data Generation Engine for Bulk Power System under Extreme Windstorms. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.