Introduction to Ember

KOVALEVA, Marina

College of Physical Sciences and Engineering, Cardiff University, Cardiff CF24 3AA, United Kingdom

* Corresponding author.
e-mail address: kovalevam@cardiff.ac.uk

CONTENTS

- Background
- Installation

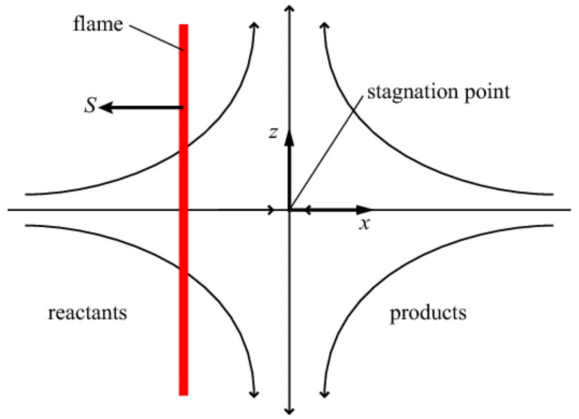
To follow the presentation and download the files, please go to:
https://github.com/marina8888/ammonia-combustion-workshop

BACKGROUND

The elemental flame model consists of a laminar flame stabilized in a stagnation flow. The boundary is defined by parameters in the ConfigFlame, with curvature controlled using xLeft, xRight, and strain also controlled. The flame shifts from the start to the end parameters in time.

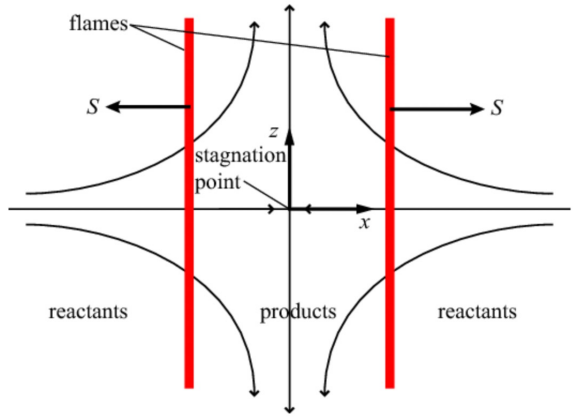
Some configurations included:
- Freely Propagating Flame
- Burner-Stabilised Stagnation Flame
- Counterflow Flame Configurations
- Tubular Flames

Single Opposed-Jet Flame



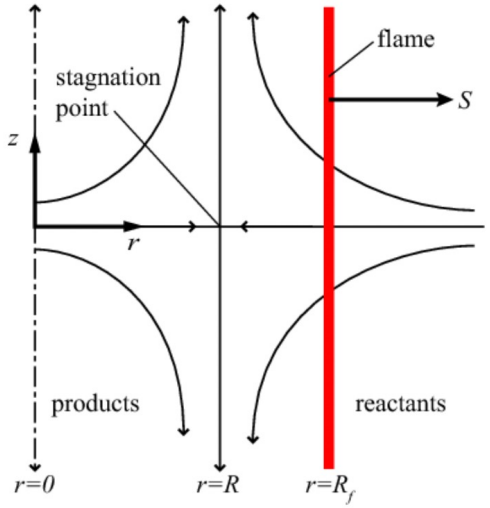
Schematic of the single opposed-jet flame

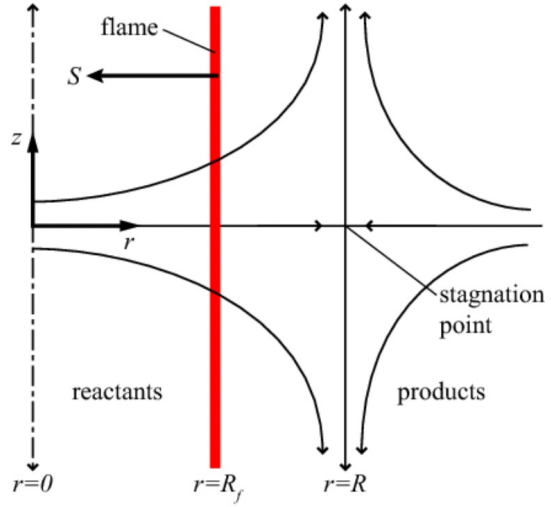
Twin Opposed-Jet Flame



Schematic of the twin opposed-jet flame

Tubular Flame at Finite Stagnation Radius

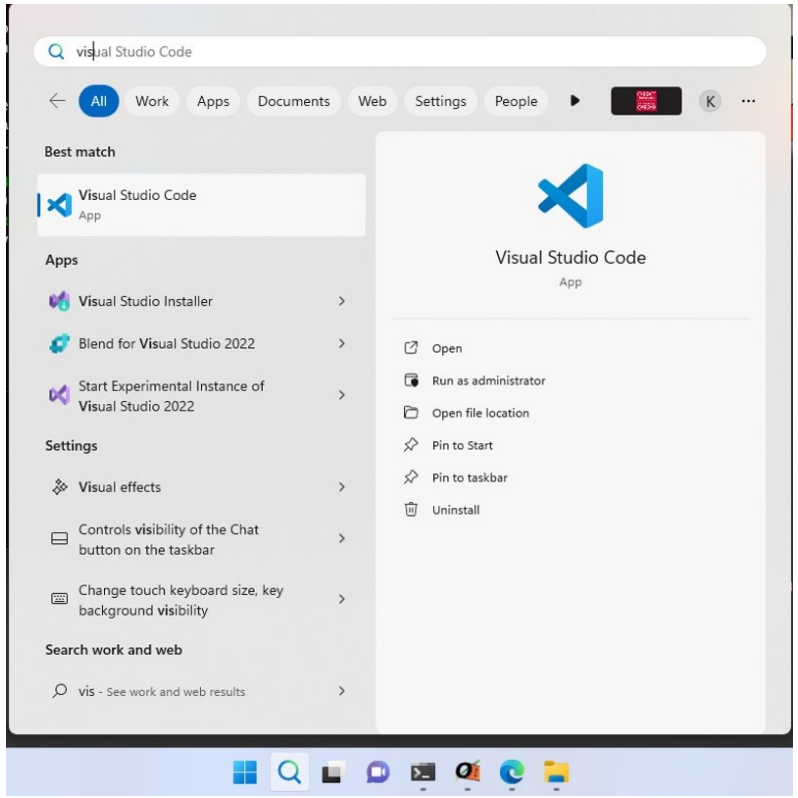


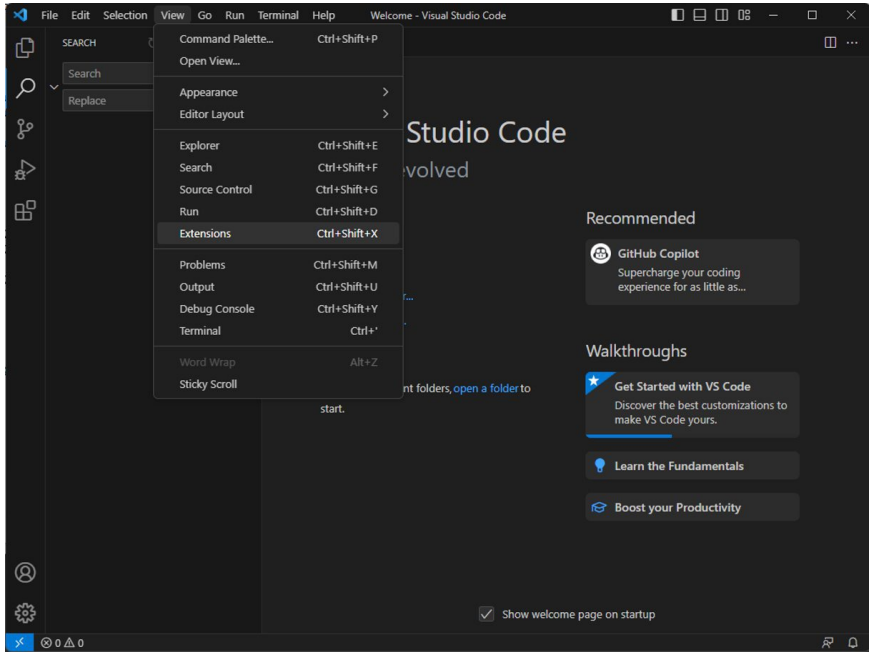


Installation

1. Start menu > Search “visual studio code” (not visual studio!)

2. In the visual studio code menu at the top of the screen, go View>Extensions

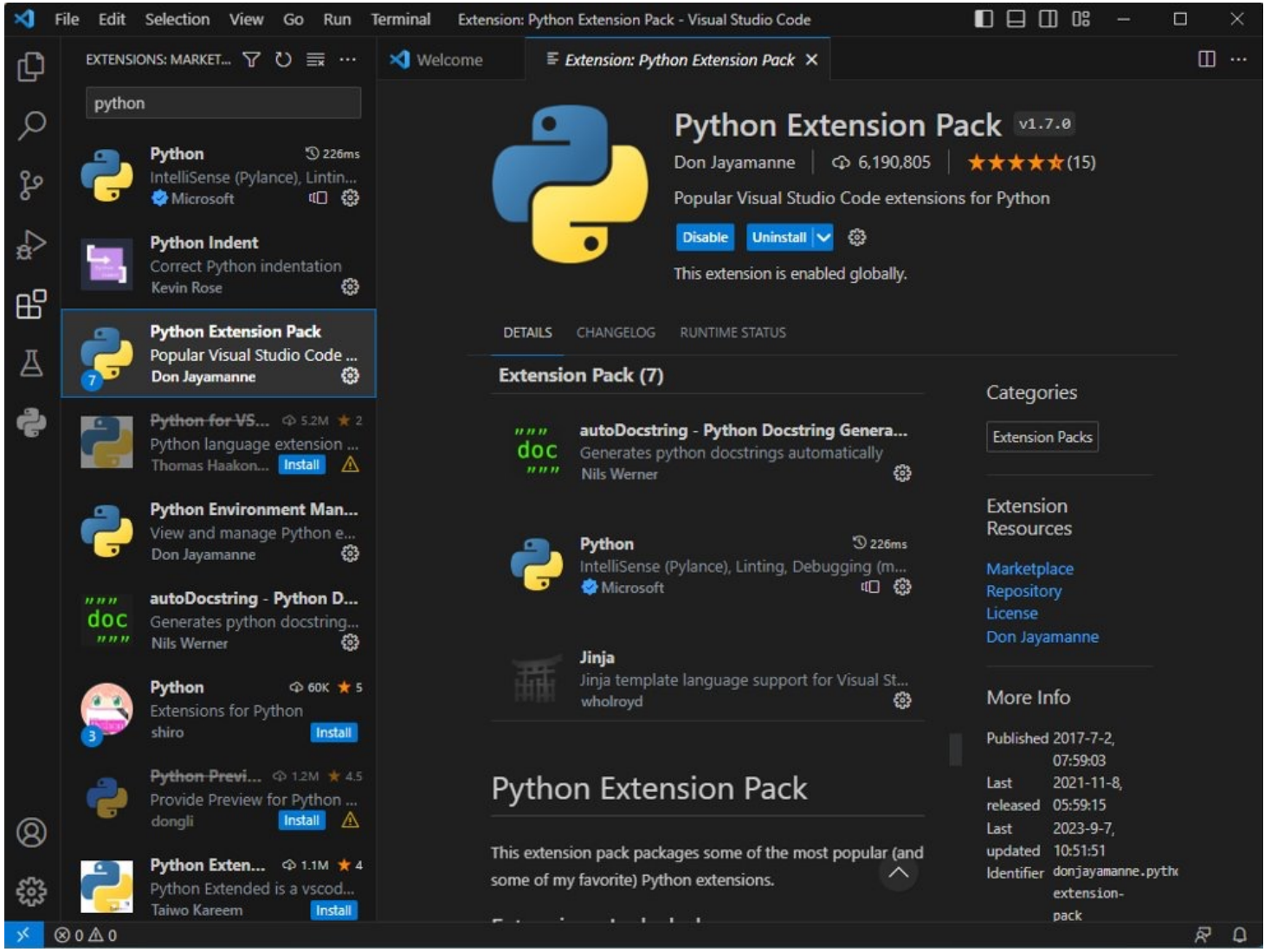




Installation

3. Under extensions, search ‘Python Extension Pack’

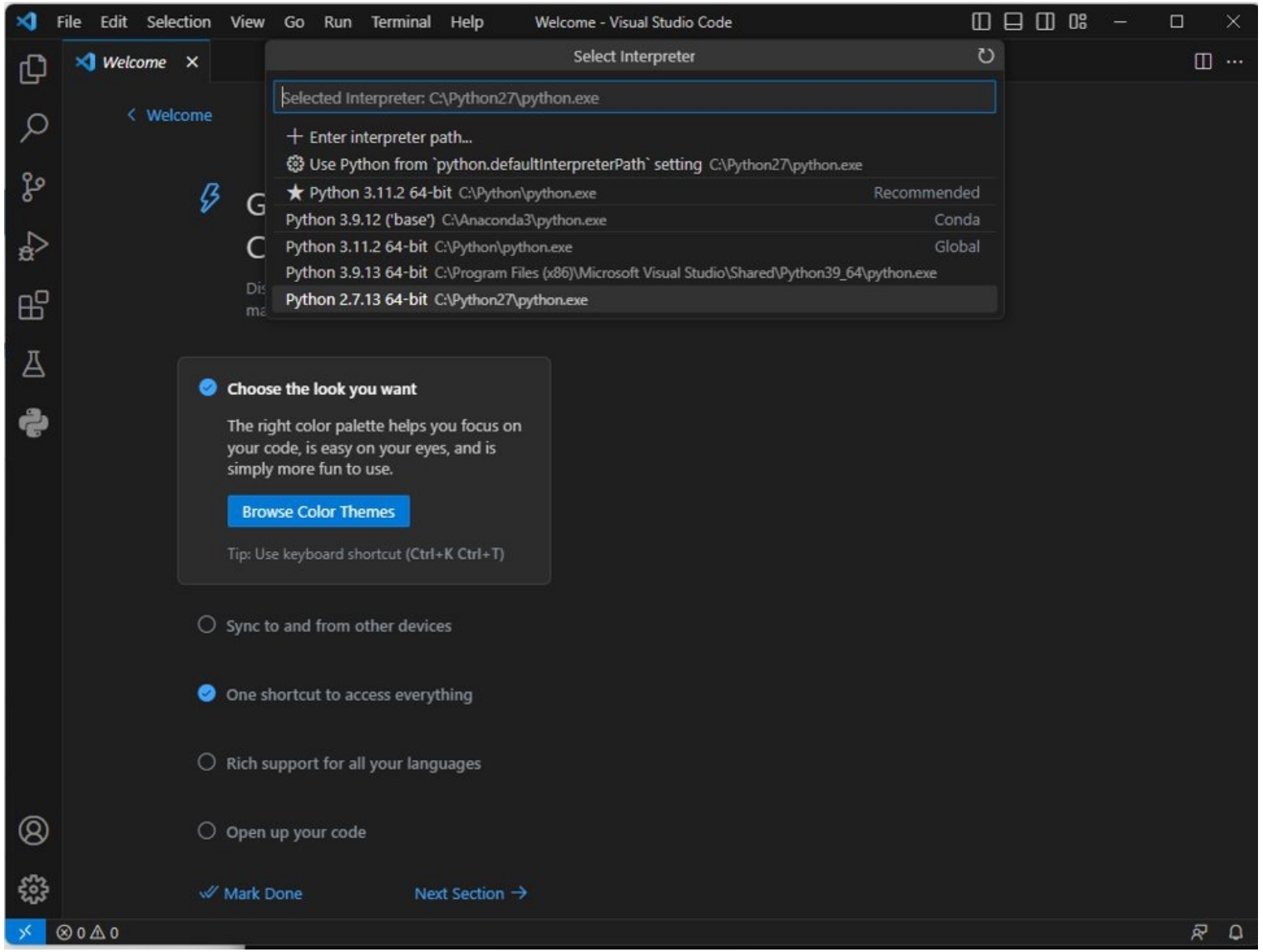
4. Once it has finished installing, it should show (0→10) in the list of already installed extensions.



Installation

5. Type ctrl+shift+P and search in the search bar ‘Python: Select Interface’

6. You will have a selection of Python environments to choose from. Select Python 2.7



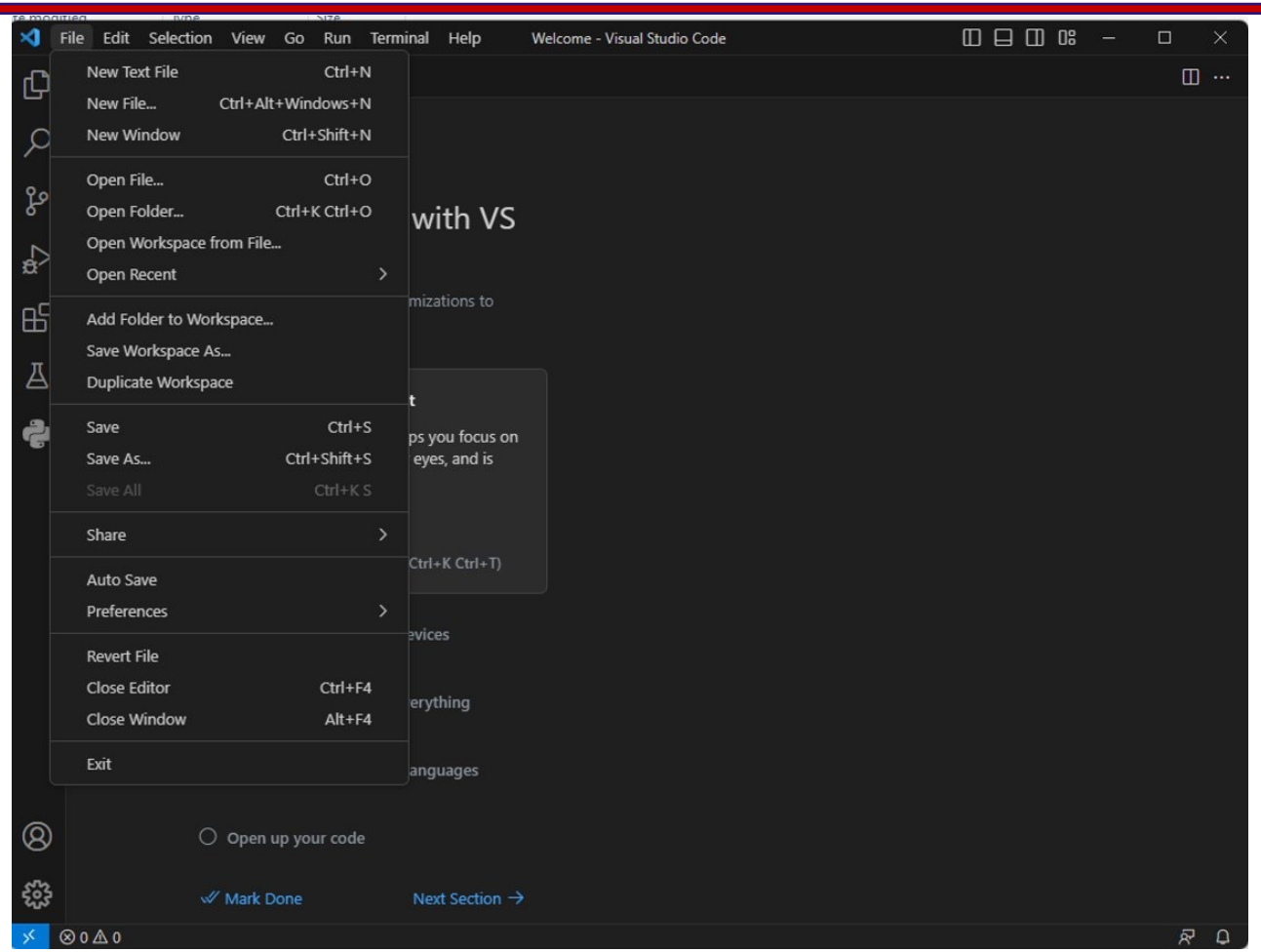
Installation

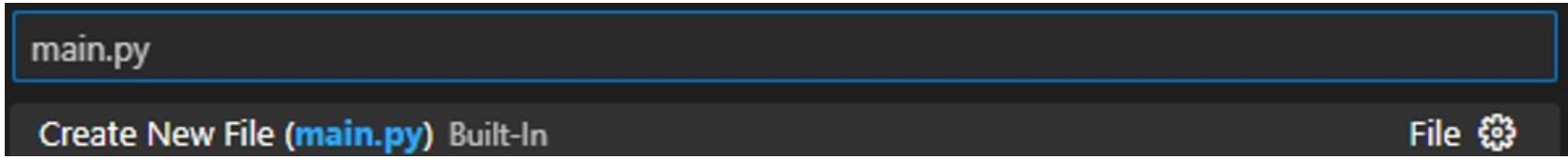
7. Create in Documents a folder called ‘ember-files’.

8. Download from github the mechanism folder, and add it to ’ember-files’.

https://github.com/marina8888/ammonia-combustion-workshop

9. File>create file, save as main.py in the ember-files folder. Copy the examples into your main.py folder. For all examples, replace {your username} with your actual username.





Installation

Ember has a github repo where all other examples can be found:

https://github.com/speth/ember/tree/main/python/ember/examples

Documentation and installation files for python3 version:

https://speth.github.io/ember-doc/sphinx/html/index.html

Installation

All ember runs by first creating a **Config class.** Then you run the run() function on the instance of the class created.

Ember and saves the files to a h5 file folder by timestep.

You need to extract data from h5 files into arrays using the utils.load() function.

Arrays which are reconstructed:

- **grid properties:** *hh, cfp, cf, cfm, rphalf, dlj*
- **thermodynamic properties:** *rho, cp, Wmx, W*
- **kinetic properties:** *wdot, q, creation_rates, destruction_rates, forward_rates_of_progress, reverse_rates_of_progress, net_rates_of_progress*
- **transport properties:** *rhoD, k, mu, Dkt, jFick, jSoret, jCorr*
- **other:** *X* (mole fractions)

In addition to these arrays, you can create a Solution() object. The data is compatible with Cantera Solution class, so the usual post-processing functions of Cantera are available.

Installation – Flux Calculator

In python, two methods of manipulating data exist for 1D flames, either through directly accessing the flame object and its in-built functions. The second is through creating a Solution object.

1. Set the properties of the Solution object (like temperature, pressure etc.) using the data from ember arrays.
https://cantera.org/tutorials/python-tutorial.html

2. Call functions available in the Cantera Solution and ThermoPhase class.
https://cantera.org/documentation/dev/sphinx/html/cython/thermo.html