

Installation I

Install a scientific Python 2.7 distribution

Any “Scientific Python” should do, but it must include NumPy, SciPy, matplotlib, and Tkinter.

Be sure to choose 2.7 and not 3.x

Spyder (a Python IDE) is very helpful but not required.

See the [Scientific Python Options](#) slides for some good choices

Download our mpctools Python package

Download zipped package:

<https://bitbucket.org/rawlings-group/mpc-tools-casadi>

Click “Downloads” (cloud icon on left)

Choose mpc-tools-casadi.zip

Unzip to a convenient location.

Installation II

Download CasADi (Version ≥ 3.0)

Windows/Linux/Mac zip file available at

<http://files.casadi.org>

Choose 3.1.1, pick OS, and download
`casadi-py27-*.zip`

Unzip `casadi`, to the `mpc-tools-casadi` folder from the previous step.

Optional: Add CasADi and `mpctools` to your Python path

Open a Python interpreter (run `python` from a terminal/command prompt)

Run the commands `import site; print site.getusersitepackages()` to see where your user site packages are stored

Move the `casadi` and `mpctools` folders to that location.

You should make the folders if they do not already exist

Scientific Python Options I

Anaconda (Linux, Windows, Mac)

Download from <https://www.continuum.io/downloads>

Perhaps a bit bloated (contains packages we do not need)

Typically the easiest version to work with.

APT (Ubuntu/Debian)

Use `sudo apt-get install spyder` to get Spyder and all dependencies.

Includes `python-numpy`, `python-scipy`, `python-matplotlib`, and `python-tk`.

Scientific Python Options II

Miniconda (Linux, Windows, Mac)

Smaller version of Anaconda

Download from <http://conda.pydata.org/miniconda.html>

After install, need to install additional components

Packages: `conda install numpy scipy matplotlib tk`

IDE: `conda install spyder`

Both commands should be entered in a
terminal/command prompt

Making Sure Everything Works

First, open a Python interpreter¹ and run `import casadi, mpctools`.

If this doesn't work, make sure your CasADi folder shows up in `import sys; print sys.path`.

If you have multiple Python distributions on your machine, don't (or at least make sure you're using the one you think you are).

Make sure you are using Python 2.7 (not 3.x).

Then, try to run the examples in `mpc-tools-casadi`.

In the Python interpreter, use `execfile("filename.py")`.

`runall.py` will run everything and tell you if there are errors, but you won't see any plots.

¹Open a command prompt/terminal in the `mpc-tools-casadi` folder and enter `python`

Path Considerations

Unless you perform the optional step, `casadi` and `mpctools` are only accessible locally.

- Avoids cluttering Python system paths

- Doesn't need admin access

- To update, simply delete folders and re-download

- Python interpreter (or IDE) must be started from this folder

If you perform the optional step, `casadi` and `mpctools` will be accessible everywhere on your machine.

- Any Python interpreter can load the packages

- May be difficult to keep track of versions

- A local install will override a global install in the local directory

Special Notes

The CasADi version you download must match the bitness of your Python installation (i.e., 32 vs. 64 bit).

For 32 bit: `casadi-py27-np1.9.1-v3.1.1.zip`

For 64 bit: `casadi-py27-np1.9.1-v3.1.1-64bit.zip`

You can check Python bitness in an interpreter with

```
import sys; print 64 if sys.maxsize > 2**32 else 32
```

Software Relationships

