

Setup Guide for “Understanding and Using the Finite Element Method”

Kyle Morris

March 16, 2023

1 Introduction

Welcome to the setup guide for my project “Understanding and Using the Finite Element Method”. This guide should get you ready to use the notebooks provided. This guide is intended for Windows users only. If you are on Mac or Linux, the overarching process will be similar, but the individual steps will be different. You will not need WSL2 or Hyper-V, you only need to install Docker itself. This guide in the Docker documentation should help you complete the installation <https://docs.docker.com/engine/install/> if you must use one of these operating systems. From there running the container should be fairly similar - use the same link to Jørgen’s image. The rest of this guide is intended for Windows users.

2 Virtualisation Environment

Docker is a software that allows us to run applications in a virtual environment called a **container**. It allows us to obtain the many dependencies necessary to work with FEniCSx (Our PDE solving software), without having to set them up ourselves. Instead of installing all the necessary modules ourselves, we can use someone else’s Docker **image** that installs them on a container for us. Then we can run Python inside the container, and run our Notebooks hassle free.

Working with Docker is the recommended way to use FEniCSx; it is widely-used in running development environments with complicated dependencies. The method of installing Docker depends on your Windows version. This is because Docker needs a way of emulating the containers in a virtual environment and different versions of Windows have different virtualisation environments. You can check your Windows version by pressing Win+R and typing `winver`, or you can right-click on This PC and choose Properties. If you are on Windows Home, see the WSL2 section. Otherwise, see the Hyper-V section.

2.1 WSL2

WSL stands for Windows Subsystem for Linux. It allows us to run a Linux environment inside of Windows, which Docker will use to run the containers. To install WSL2, go to the command line and execute the command `wsl.exe --install` [1]. This will install Ubuntu by default. After this, restart your computer. When you reboot you will be prompted to set up a user on your new Ubuntu system. After doing this, WSL2 will be installed. It can be used by searching for Ubuntu in the search bar or with the `wsl` command in the command line.

2.2 Hyper-V

If you use Windows 10/11 Pro, Enterprise, or Education, you can use Hyper-V instead (although WSL2 will still work). Hyper-V stands for Hyper-Virtualisation and is a virtualisation software built into Windows. It is another software that allows you to create virtual machines inside of Windows. To use it, you will need to ensure Hyper-V is enabled on your system. To do this search Hyper-V in the search bar, and it should come up with the result “Turn Windows features on or off”. Click this, find Hyper-V in the list and enable it by ticking the box and pressing Ok. [2]

3 Docker Desktop

Now we are free to download Docker Desktop. You can get it from <https://www.docker.com/products/docker-desktop/>. Run the executable file. If prompted, choose the backend you downloaded in 2 (either WSL or Hyper-V). You should now have Docker Desktop installed. You can find it by searching for Docker Desktop in the search bar.

4 Creating The Container

Now we can create our container. The docker container will inherit from a prebuilt image, created by Jørgen S. Dokken, creator of The FEniCSx tutorial [3]. His image includes many dependencies, some of which we will use and others we will not. The list is as follows:

- Basic dependencies: Python, JupyterLab, Pip, Setuptools
- PDE solving: DOLFINx, UFL, Basix, FFCX
- Data visualisation: Matplotlib, Seaborn, Pyvista, Pythreejs and Ipy-gany
- Numerical computation: PETSc, Pandas, H5py, Meshio, Tqdm, MPI

To create and run the container, type `docker run -p 8888:8888 ghcr.io/jorgensd/dolfinx-tutorial:v0.5.1` in the command line. This should launch a JupyterLab session. You can access the JupyterLab by copying the bottom URL in the command line and pasting it into your browser. Alternatively, you can go to <http://localhost:8888/lab> and copy the token from the console. (The token is the long string after “token=” in the url).

You may see a file called `requirements.txt`. This can be removed. To check everything has worked correctly, you can open the terminal and run `python3`. If this works, check dolfinx is installed by running `import dolfinx`. If there are no errors congratulations! You are now ready to begin using FEniCSx.

4.1 Future use

To stop running the container, the easiest way is to open Docker Desktop. Then go to containers and find the one which inherits from Jørgen’s image. It will have a random name. Press the stop button located to the right of it’s entry. In future, launching the container is as simple as pressing the play button where the stop button used to be.

References

- [1] Joey Sneddon (2023) *How to Install WSL 2 on Windows 10 (Updated)*, Omg!Ubuntu!
<https://www.omgubuntu.co.uk/how-to-install-wsl2-on-windows-10>
- [2] Scoley et. al (2022) *Install Hyper-V on Windows 10*, Microsoft Learn, Windows Virtualisation Guides
<https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/quick-start/enable-hyper-v>
- [3] Jørgen S. Dokken (2023) *The FEniCSx tutorial*, jsdokken.com,
<https://jsdokken.com/dolfinx-tutorial/index.html>