Additive-Manufacturing Code User Guide

by Swapnil Shadanan Bhure

⭐ This code has been developed by Ravi Kumar Singh as a part of his Master's Thesis at Materials Modelling Group , Indian Institute of Science, Bangalore, India. Digital copy of his thesis is available on our group's GitHub repository.

⚠️ Recommeneded way of using OpenFOAM is with OpenFOAM-in-Box which works on any Linux OS. This code works with OpenFOAM-in-Box-18.10 or older.

The code has been successfully tested on following versions of OpenFOAM on Ubuntu 18.04, CentOS 6 (Cluster) and CentOS 7 (Cluster):

1. OpenFOAM-4.x with swak4Foam-dev

2. OpenFOAM-6 with swak4Foam-dev

3. OpenFOAM-in-Box-18.10v1

4. OpenFOAM-in-Box-17.10

The code does not work with newer versions of OpenFOAM and OpenFOAM-in-Box than mentioned above.

Table of contents:

Download and install OpenFOAM-in-Box:

Download:

Download OpenFOAM-in-Box-18.10 using the following link:

> OpenFOAM-in-Box-18.10v1-r1235-linux64-glibc-special.sh
>
> 📁 https://drive.google.com/file/d/1gzK8ipaC-be_dvljckus6uFizazkO1eU/view?usp=sharing

Note: On official website only the latest version is available (20.09 at the time of writing this document) and this code doesn't work with it.

Install:

1. Copy the installation package in your installation path. You can do that either manually or use following linux commands:

```
mkdir -p ~/OpenFOAM/OpenFOAM-in-Box
# Go to location where openfoam-in-box installation file is located and then execute the following:
cp OpenFOAM-in-Box-18.10v1-r1235-linux64-glibc-special.sh ~/OpenFOAM/OpenFOAM-in-Box/
```

2. Install the package, e.g.:

```
cd ~/OpenFOAM/OpenFOAM-in-Box/
bash OpenFOAM-in-Box-18.10v1-r1235-linux64-glibc-special.sh -install
```

When opening a new terminal window the OpenFOAM environment (system) variables need to be loaded:

```
source  ~/OpenFOAM/OpenFOAM-in-Box/OpenFOAM-in-Box-18.10v1/OpenFOAM-dev/etc/bashrc
```

For easier access, you can create an alias in your system's .bashrc file by using the following command.

```
echo "alias ofbox18='source ~/OpenFOAM/OpenFOAM-in-Box/OpenFOAM-in-Box-18.10v1/OpenFOAM-dev/etc/bashrc" >> ~/.bashrc
source ~/.bashrc
```

This will create and alias named `ofbox18` in .bashrc file. Next time when you want to source OpenFOAM environment variables just run `ofbox18` in a new terminal.

Alias is created instead of directly adding the source line in .bashrc because we might have different versions of OpenFOAM on our system and each might have its own set of home directory, environment variables and commands. This functionality of sourcing every time helps us choose the appropriate OpenFOAM version on the go.

OpenFOAM-in-Box comes with in-built swak4Foam so no separate installation is required.

Reference:

Openfoam In Box - CFD SUPPORT

OpenFOAM for Linux independent on system libraries

https://www.cfdsupport.com/openfoam-in-box.html

Using Additive-Manufacturing code

💡 **For the following guide on additive codes, it is assumed that you are well conversant will the basic operation of OpenFOAM. You should be able to edit OpenFOAM cases, compile solvers, run cases and view results in Paraview.**

Compiling Solvers:

1. Open a new terminal window and run:

```
ofbox18
```

💡 Run this command every time you open a new terminal window or tab. This makes sure that the OpenFOAM-in-Box commands and environment variables are sourced. This is crucial in case if we have multiple versions of OpenFOAM.

2. **Compiling the solver:**

   a. Go to `marangoniBoundary_Condition` directory and run

   ```
   wmake
   ```

   b. Go to `Additive_manufacturing_solver` directory and run:

   ```
   wmake
   ```

Both of these should compile without any error for the first time or else it would mean that the installation is not done correctly.

Preparing additiveTestCase:

1. Go to additiveTestCase directory via terminal. Clean the old/previous simulation data (if any) by running the following command:

```
./Allclean
```

If this command does not work, make sure you ran `of4x` in the terminal and that the `Allclean` file is executable. To make the file executable run the following command:

```
chmod +x Allclean
```

You need to make a file executable just once. It will remain that way until you move around to an external drive or share via email, etc. In such a case, just run the above command.

2. `additiveTestCase` directory is similar to any `OpenFOAM case` directory which contains directories like `0, constant, system` and few other files. Edit them as per your problem.

> 💡 **Note:** Remember that computation uses physical cores and not hyper-threads. So specify number of physical cpu cores you want to use in `numberOfSubdomains` in `decomposeParDict` file. Beware, if you have a quad-core processor on a laptop then for testing prefer using 2 cores. Using all the 4 cores on a laptop might slow down other apps or heat-up the laptop. You may use all cpu cores if you have a gaming or workstation laptop. Make a decision based on what your laptop can handle. For lab desktops and clusters you may use all the physical cores.

Running additiveTestCase:

> 💡 Remember to run `of4x` if using a new terminal tab or window as explained here.

To run the case, use:

```
./Allrun
```

Important points to remember:

1. There should not be any spaces in the file path in OpenFOAM.

2. Whenever you open a new terminal, source the OpenFOAM environment variables from the version that we want to use. For OpenFOAM-in-Box use:

```
ofbox18
```

3. To unload the environment variable, use:

```
wmUNSET
```

Thus, we don't have to close the window to start another version of OpenFOAM.