



Instructions for Channelflow 2.0

Channelflow 2.0 is version 2.0 of [Channelflow](#) software based on language C++ and parallelized for numerical analysis of the incompressible flow in channel geometries. Channelflow 2.0 is developed by the Emergent Complexity in Physical Systems Laboratory ([ECPS](#))

Installation

Channelflow 2.0 requires some external libraries:

- MPI - a standardized and portable message-passing standard designed to function on parallel computing architectures. We can use OpenMPI or MPICH for this.

```
sudo apt-get install libopenmpi-dev
```

- Eigen3 - a library for linear algebra: matrices, vectors, numerical solvers, and related algorithms

```
sudo apt-get install libeigen3-dev
```

- FFTW3 - a library for computing the discrete Fourier transform (DFT) in one or more dimensions, of arbitrary input size, and of both real and complex data

```
sudo apt-get install libfftw3-dev
```

- FFTW3-MPI - a sub-library of FFTW3 package for MPI

```
sudo apt-get install libfftw3-mpi-dev
```

- NetCDF - a set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.

```
sudo apt install netcdf-bin libnetcdf-dev
```

Now, let us build the Channelflow 2.0. We first need to clone official source code via [GitHub](https://github.com/epfl-ecps/channelflow).

```
git clone https://github.com/epfl-ecps/channelflow.git
```

and build it by using syntax as follows

```
mkdir build
cd build
cmake PATH_TO_SOURCE -DCMAKE_BUILD_TYPE=<debug/release> (configuration options)
make -j
make install
```

Sometimes, we can get a few errors. Here, I will list some common errors, which I got when installing, and how to fix it.

- If get errors regarding `undefined reference to "fftw_..."`, add below flag after `cmake`

```
-DCMAKE_CXX_FLAGS_RELEASE:String=" -lfftw3 -lm "
```

- If get errors regarding `fPIC`, add `-fPIC` in above flag or rebuild FFTW with flag `--enable-shared`

```
-DCMAKE_CXX_FLAGS_RELEASE:String=" -fPIC "
```

- If get errors regarding library finding (e.g. `libfftw3.a`, `libfftw3_mpi.a`) when compiling. We need to rebuild FFTW3 by hand with both flags `--enable-mpi` and `--enable-shared` after `./configure`. Pls read detailed instructions for installation [here](#).

Channelflow supports, beneath other standard cmake flags, the following options

Option	Values	Default	Description
<code>-DCMAKE_INSTALL_PREFIX</code>	path	usr/ local	Installation path for make install
<code>-DUSE_MPI</code>	ON/ OFF	ON	Enable MPI
<code>-DWITH_SHARED</code>	ON/ OFF	ON	build shared channelflow and nsolver libraries
<code>-DWITH_STATIC</code>	ON/ OFF	OFF	build static libraries (also enables linking to static libraries)
<code>-DWITH_PYTHON</code>	ON/ OFF	OFF	build a python wrapper for flowfields, disabled by default because it requires boost-python
<code>-DWITH_HDF5CXX</code>	ON/ OFF	OFF	enable legacy .h5 file format (using HDF5 C++)

A sample installation, with all features enabled, might look like this:

```
cmake ../channelflow -DCMAKE_BUILD_TYPE=release -DCMAKE_CXX_COMPILER=/usr/bin/mpicxx -DCMAKE_CXX_F
make -j4
sudo make install
```