This directory contains the source code for the Magnetic Flux Eruption (MFE) magnetohydrodynamic (MHD) code. The MFE code solves the single fluid magnetohydrodynamic equations in a Cartesian x-y-z grid domain or a spherical r-θ-φ grid domain, with the energy equation incorporating the non-adiabatic effects of the solar corona that may include the field-aligned electron heat conduction, optically thin radiative cooling, and an empirical coronal heating. A description of the equations solved, and the numerical algorithms used for the MFE code can be found in Fan (2017, ApJ, 844:26, <https://doi.org/10.3847/1538-4357/aa7a56>).

The code is developed at HAO/NCAR and has been used to model magnetic flux emergence from the interior into the corona (e.g. [Fan 2009, ApJ,697:1529](http://dx.doi.org/10.1088/0004-637X/697/2/1529)), the initiation of coronal mass ejections (CMEs)(e.g. [Fan 2022, ApJ, 941:61](https://doi.org/10.3847/1538-4357/aca0ec), [Liu et al. 2022, ApJ, 940:62](https://doi.org/10.3847/1538-4357/ac961a), [Afanasev et al. 2023, ApJ, 952:136](https://doi.org/10.3847/1538-4357/acd7e9)), and solar prominence eruptions in the large-scale solar corona (e.g. [Fan 2018, ApJ, 862:54](https://doi.org/10.3847/1538-4357/aaccee), [Fan and Liu, 2019, Front. Astron. Space Sci. 6:27](https://doi.org/10.3389/fspas.2019.00027)). Recently, the model has been improved to carry out boundary data-driven simulations of the observed solar CME events using the observed photospheric vector magnetograms as the lower boundary condition.

Currently in this public repository of the code, *only* the setups (and instructions) for running the standard Orszag-Tang Vortex test and a CME event simulation published in Liu et al. (2022) are provided (in the probs/ subdirectories). Setups and instructions for running other published simulations will be continually added in the future.

Here after we will refer to the cloned MFE\_mirror directory as the root directory.

1. Instruction to do the Orszag-Tang Vortex Test (Orszag & Tang, J. Fluid Mech., 90, 129, 1998) run with the MFE code on a Mac running macOS with gfortran and openmpi installed:

1) Copy files (ModPar.F, ModUserSetup.F, ModNonIdealRHS.F makefile) from the probs/ orszag-tang\_xy directory to the root directory

2) Compile the code by typing “make xmfe” in the root directory, which creates the executable code: xmfe

3) Create a separate run directory <RUNDIR> and copy the executable xmfe to <RUNDIR>

4) Copy all the files (inparam, jobscript.8cpu) in the probs/orszag-tang\_xy/rundir/ to the <RUNDIR>

5) cd to <RUNDIR> and run the test by typing: source jobscript.8cpu

Note: The Orszag-Tang test is a 2D problem. The above instruction is for running the Orszag-Tang Vortex test in the x-y plane with z being the invariant direction. Since the MFE code is a 3D code, one can also run the test in the y-z plane with x being the invariant direction, or in the z-x plane with y being invariant. The instructions for running with the latter 2 setups are the same as the above, except using the files in the probs/orszag-tang\_yz or probs/orszag-tang\_zx directories instead of the probs/orszag-tang\_xy directory. The results for the 3 setups should be identical.

2. Instructions on running the CME event simulation published in Liu et al. (2022) <https://iopscience.iop.org/article/10.3847/1538-4357/ac961a> on NCAR’s Derecho supercomputer at NWSC:

1) Copy files (ModPar.F, ModUserSetup.F, ModNonIdealRHS.F makefile) in the probs/Liu2022 directory to the root directory.

2) Compile the code by typing "make xmfe" in the root directory which creates the executable code: xmfe

3) Create the run directory <RUNDIR> on the scratch disk and copy the executable xmfe to <RUNDIR>

4) Copy the files parameter file and the PBS runscript (inparam, pbsscript\_derecho)in probs/Liu2022/rundir to the <RUNDIR>. Make the necessary edits in the pbsscript\_derecho PBS script file for your user information.

5) Dowload the initial state data files to <RUNDIR> via GLOBUS from the

public GLOBUS shared point given in the data\_initial\_state\_link file in

probs/Liu2022/rundir.

6) Run the code in the <RUNDIR> by submitting the PBS script:

qsub pbsscript\_derecho

3. Simulation output data:

After a simulation with the MFE code is completed, the data files written in the <RUNDIR> include: b1\_XXXX.dat b2\_XXXX.dat, b3\_XXXX.dat, v1\_XXXX.dat, v2\_XXXX.dat, v3\_XXXX.dat, d\_XXXX.dat, e\_XXXX.dat....,

where XXXX are the output time steps. These are unformatted binary data files of the individual variables at the XXXX output step. There is also a file for the grid information grid.dat, and a file physparams.dat for the various physical parameters and the units for the various quantities used by the simulation.

The data for each time step and for each variable can be read into IDL using the IDL routines in the idlcodes/ directory under the root directory. See the readme\_idl file in the idlcodes/ directory for explanations on how to call the IDL routines to read the data cubes along with the grid and physical units’ information into IDL.

The data can also be read into Python using the python code given in the python/ subdirectory in the root directory. See the header of pyMFE.py in the python/ directory for explanation on how to use it to read the data into python.