MHIT36

Alessio Roccon

February 6, 2025

Part I
Formulation

1 Governing equations

A first tentative try of the governing equation for boiling flow reads as [adapted from Karniadakis]:

$$\nabla \cdot \mathbf{u} = 0\,, \tag{1}$$

For the Navier-Stokes equations, by assuming constant viscosity and density, the following equation can be derived:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \frac{\sigma}{\rho} k \mathbf{n} + \mathbf{f} \tag{2}$$

The interface is captured using a second-order phase field method:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) = \nabla \cdot \left[\gamma \left(\epsilon \nabla \phi - \phi(1-\phi) \frac{\nabla \phi}{|\nabla \phi|} \right) \right]\,, \tag{3}$$

The first two terms at the right hand side are the classical sharpening and diffusive terms.

Likewise, ϵ should be set equal to:

$$\epsilon > 0.5 \Delta x \tag{4}$$

2 Numerical implementation

2.1 NS solver

Projection-correction + Poisson solver based on 3D FFT along all directions The first step is to rewrite the NS equations as follows:

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla \cdot (\mathbf{u}\mathbf{u}) - \frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \frac{\sigma}{\rho} k \mathbf{n} + \mathbf{f} \tag{5}$$

Then, we perform the projection step where the right hand side is computed using the right hand side evaluated at time step n. As pressure is not known, the pressure gradient term is ignored during this first step. Discretizing in time the equation and ignoring the pressure gradient term as well as the surface tension term (for simplicity), we have:

$$\frac{\mathbf{u}^* - \mathbf{u}^\mathbf{n}}{\Delta t} = -\nabla \cdot (\mathbf{u}\mathbf{u})^n + \nu \nabla^2 \mathbf{u}^n + \mathbf{f}^n \tag{6}$$

Thus, we can obtain the provisional field $\mathbf{u}^*$ as follows:

$$\mathbf{u}^* = \mathbf{u}^n + \Delta t(-\nabla \cdot (\mathbf{u}\mathbf{u})^n + \nu \nabla^2 \mathbf{u}^n + \mathbf{f}^n)\,, \tag{7}$$

This provisional field (result of the the projection step) is however not divergence free and we must correct so to obtained the filed $n+1$, which should be divergence free. We can correct the field as follows:

$$\mathbf{u}^{\mathbf{n+1}} = \mathbf{u}^* - \frac{\Delta t}{\rho} \nabla p^{n+1}\,, \tag{8}$$

where this equation comes from:

$$\frac{\mathbf{u^{n+1}} - \mathbf{u}^*}{\Delta t} = -\frac{1}{\rho}\nabla p^{n+1} \tag{9}$$

to make clear that the algorithm is really just an operator splitting approach in which one considers the convective and viscous forces (in the first half step) and the pressure forces (in the second half step) separately. Computing the right-hand side of the second half step requires knowledge of the pressure at $n+1$. This is obtained by taking the divergence and requiring that the new flow field $n+1$ is divergence free.

$$\nabla^2 p^{n+1} = \frac{\rho}{\Delta t}\nabla \cdot \mathbf{u}^* \,, \tag{10}$$

This is a Poisson equation (most expensive part of the entire scheme), which should be solved using FFT-based solvers (present case) or iterative/multigrid methods.

2.2 CAC solver

Euler explicit + FD2. Solver is totally explicit

2.3 Forcing of turbulence

When performing direct numerical simulations of homogeneous turbulence, one would like to force turbulence for two reasons. First, it permits to reach higher Reynolds numbers than in freely decaying turbulence. Second, under some assumptions, statistics can be obtained with time-averaging rather than ensemble averaging, which would be very costly considering the fact that refined statistics require a large number of samples. To study statistically stationary turbulence, many velocity forcing schemes have been used so far in numerical simulations. In homogeneous spectral simulations, large-scale forcing methods consist in providing energy to the low wavenumber modes, which is consistent with the concept of Richardson cascade. For example, considering a working an external force $\mathbf{f}$ which is a linear combination of sines and cosines with wavevectors of modulus lesser than or equal to k_f, the forcing contribution in both Craya's equation and energy spectral density equation vanishes at wave numbers greater than k_F. Therefore, in statistically stationary turbulence, this external force feeds low wavenumbers and then part of this energy is transferred to larger wavenumbers through the nonlinear term.

2.3.1 ABC forcing scheme

The ABC forcing is defined as follows:

$$\mathbf{f} = [B\cos(k_F y) + C\sin(k_F z)]\mathbf{i} + [C\cos(k_F z) + A\sin(k_F x)]\mathbf{j} + [A\cos(k_F x) + B\sin(k_F y)]\mathbf{k}\,, \tag{11}$$

for a given large scale wavenumber k_F. Since ABC is an eigenfunction of the curl operator with eigenvalue k_F, the corresponding contribution in the helicity equation is nonzero and thus the ABC forcing injects helicity, in addition to energy, in the flow [See Intermittency in the isotropic component of helical and non-helical turbulent flows] Usually $A = B = C = 1$, other possible choice: $A = 0.9$, $B = 1$ and $C = 0.9$, see paper listed above.

2.3.2 TG forcing scheme

The TG forcing is defined as follows:

$$\mathbf{f} = f_0[\sin(k_F x)\cos(k_F y)\cos(k_F z)\mathbf{i} - \cos(k_F x)\sin(k_F y)\sin(k_F z)\mathbf{j}]\,, \tag{12}$$

Here f_0 is the forcing amplitude, which can be set to have in the turbulent steady state all runs with r.m.s. velocities near unity.

2.3.3 Lundgren scheme

A local force proportional to the local velocity is imposed on the fluid:

$$\mathbf{f} = A_f \mathbf{u}\,, \tag{13}$$

where the coefficient A_f is calculated as follows:

$$A_f = \frac{\epsilon}{3\langle u_{rms}^2 \rangle} \tag{14}$$

To generate turbulence, the Lundgren forcing requires a non- zero initial velocity field

2.3.4 Mallouppas-type force

The forcing scheme proposed by Mallouppas, George, and van Wachem reads as follows:

$$\mathbf{f} = \frac{\rho}{\Delta t}\frac{\sqrt{k_{wanted}} - \sqrt{k_{computed}}}{\sqrt{k_{wanted}}}\mathbf{u}_{triggered} \tag{15}$$

where Δt is the time step, k_{wanted} is the specified turbulent kinetic energy, $k_{computed}$ is the computed turbulence kinetic energy of the domain, and $\mathbf{u}_t riggered$ is a pseudo-velocity field, which is carefully chosen and synthesized from a model spectrum. Following the explanation of Cant we use the procedure proposed by Kwak, Reynolds, and Ferziger to generate a divergence-free isotropic velocity field corresponding to a given energy spectrum. The Batchelor–Townsend energy spectrum was used as an initial energy spectrum to create the velocity field. This spectrum represents the later stages of the decay of grid turbulence and is expressed as follows: Code to generate a synthetic velocity field corresponding to the Batchelor–Townsend energy spectrum available here https://web.stanford.edu/∼hjbae/CBC.

2.4 Simulation setup

The governing equations are solved in a triple-periodic domain with length $L = 2\pi$. The variable are defined on a staggered grid with scalars (pressure and phase-field variables) defined at cell center (empty dot in figure 1) and velocity components at cell faces (full circles in figure 1). Periodicity is implicitly applied as shown in figure 2.

3 Code structure and programming

Code can exploit a single GPU using openACC and managed memory feature. Arrays for FFT are pinned so to exploit maximum bandwidth between CPU and GPU. Possible improvement can be obtained profiling the code using Nvidia Nsight.

Possibilty to go multiGPU using the cuDecomp library? the library should be able to deal with halo exchanges (of any dimension) plus transposition. The only doubt is about the openACC compatibility and managed memory feature.

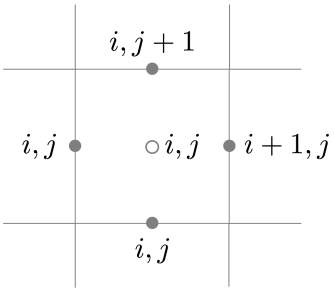

Figure 1: Staggered grid in 2D and examples of node numbering used in MHIT36.

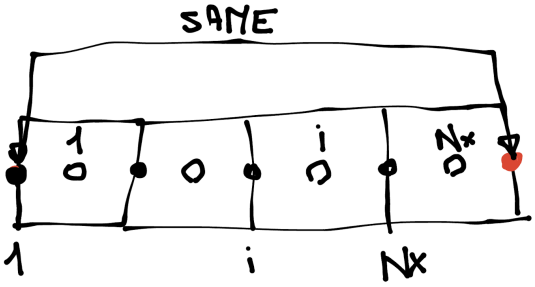

Figure 2: Periodicity handling in MHIT36.

3.1 Input file

An input.dat file is present in the src folder. This file can be used to set the parameters and also change/disable the forcing.

3.2 Restart of a simulation

To restart a simulation, you need to modify the input.dat solely. If the restart option is set equal to 1, the code will look for the corresponding iteration number file (itstart) in the src/output folder. There are no ad-hoc restart files; this is to simplify the management of the simulation files and avoid too much I/O operations. If restart is equal to 1, the subroutine readfield_restart will be called instead of the readfields, which only read file from the src/init folder which have the generic name (u.dat, etc.). Same apply when phase-field is present.

4 HIT fundamentals

The computation of some fundamentals length and time scales is reported in the following. These quantities should be checked that simulation results are in agreement with theoretical results and archival literature. The first one is the spectrum of turbulent kinetic energy, as shown in figure 3. We can distinguish between the inertial range and the dissipative range (viscous effect dominant). We can also see the energy cascade from small wavenumber (larger vortex) to high wavenumber (small vortex). In the inertial range, the spectrum should agree with the following scaling:

$$E(k) = \alpha \epsilon^{2/3} k^{-5/3}\,, \tag{16}$$

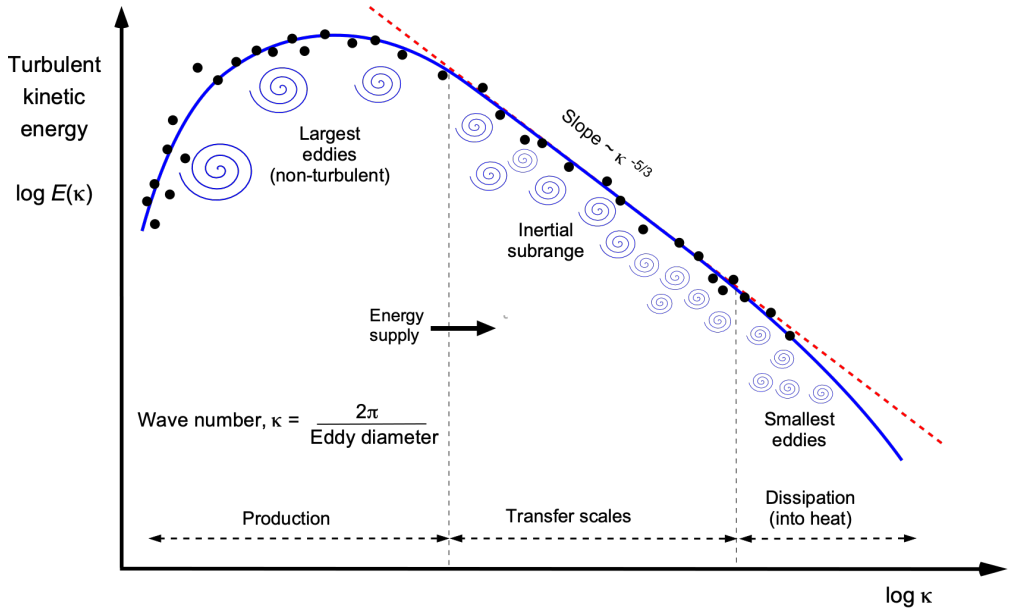

Figure 3: Example of HIT spectrum.

where $\alpha = 1.6$ is the Kolmogorov constant and ϵ is the dissipation rate.
Other quantities of interest: Taylor micro-scale (length scale):

$$\lambda = \sqrt{\frac{15\nu}{\epsilon}} u' \tag{17}$$

or also

$$\lambda = \sqrt{\frac{u'}{\left(\overline{\frac{\partial u'}{\partial x}}\right)^2}} \tag{18}$$

where we used the definition of dissipation for HIT:

$$\epsilon = 15\nu \left(\overline{\frac{\partial u'}{\partial x}}\right)^2 \tag{19}$$

Kolmogorov length scale:

$$\eta_k = \left(\frac{\nu^3}{\epsilon}\right)^{1/4} \tag{20}$$

where one should check that a simulation is well resolved if $k_{max}\eta_k > 1$ (equal to ensures the best accuracy). The value k_{max} is the highest wavenumber resolved and can be computed as follows:

$$k_{max} = \pi/\Delta \tag{21}$$

where $\Delta = 2 * pi/N_x$ where 2π is the length of the domain.
Correlation to obtain some fast estimates:

$$\epsilon \simeq \frac{u'^3}{L} \tag{22}$$

where L is the integral time scale.