Pseudo-spectral DNS - parallel code user manual

Parallel code

It is possible to use this parallel code on every computer or cluster (i.e. HPC or multicore processor pc) on which are installed MPI parallel libraries. The code outputs every saved instant, three files u1.xy.bin, u2.xy.bin,u3.xy.bin in single precision format. Moreover it is not necessary compile again codes when resolution is changed: number of points is read when codes are running and variables are dinamically allocated depending on resolution (FORTRAN 90).

1 Code compiling

Compilor of the serial code is gfortran, while compilor of the parallel code ìs mpif90 (fortran+librerie mpi).
All the instructions are written in the Makefile. The source code are:

Running of the simulations

- dns14_mpi.x → Navier-Stokes equations solution;
- dns14_scal.x → Navier-Stokes equations and one (or more) passive scalars solution.

Pre-processing e e post-processing

- dns14_mpi.x → computes 3D spectrum, integral scalle, Taylor scale, energy and dissipation for isotropic turbulence;
- dns14_ske.x → compute sedcond moment, skewness and kurtosis of the three components of velocity (average on planes with $x_3 = \text{cost}$);
- dns14_ske_scal.x → Computes second moment, skewness and kurtosis of the passive scalar (average on planes with $x_3 = \text{cost}$);
- dns14_genera.x → outputs initial data of isotropic turbulence (using a generator of random number);
- dns_genera.x→ similar to dns14_genera.x, ma non è un codice parallelo.

2 Codes running

In the same directory:

1. Main code (file .x);
2. files has parametesr and/or initials data needed by main code);
3. file with "script" to run the main parallel code.

dns14_mpi.x

It solves Navier-Stokes equations (pseudospettrale+Runge-Kutta 4 method) and main source is the file dns14_conf.f90.
Input file:

- param_ns.txt → contains following parameters:

N	points number in directions 1, 2;
$N3$	points number in directions 3;
Δt	temporal step;
$Ntot$	total steps number;
$Nsalva$	steps number saving;
Re	Reynolds number;
$nomefile$	file name with three velocity components (each for every component) to use as initial conditions.

- file containing initial consitions.

Outputs file:

- avvio.txt → parameters file;
- the three componets of velocity for each saved instant u1.xy.bin, u2.xy.bin, u3.xy.bin →.

Notes

1. The domain is a parallelepiped with dimensions $2\pi \times 2\pi \times 2\pi \frac{N_3}{N}$ (fig.1).
2. It is not necessary recompile code if data dimensions are changed. It is only necessary update file param_ns.txt.
3. It is not limited to a cube.

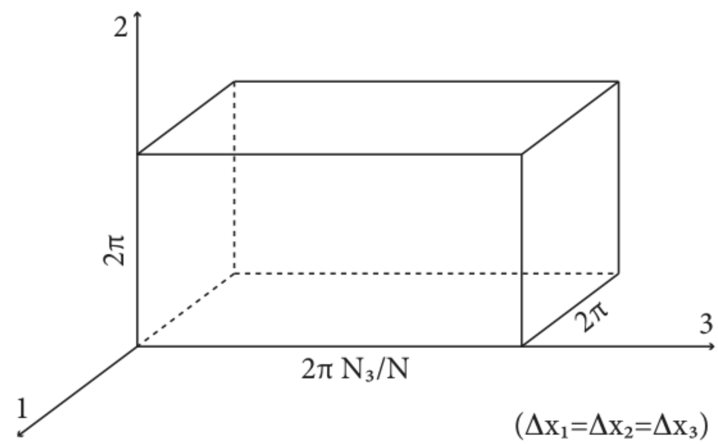

Figura 1: Computational domain.

dns14_scal.x

It solves Navier-Stokes equations as dns14_mpi.x; Moreover it solves passive scalar transport equations[1]:

$$\partial_t \vartheta + \nabla \cdot (\mathbf{u}\vartheta) = \frac{1}{\mathrm{Sc} \cdot \mathrm{Re}} \nabla^2 \vartheta$$

in adimensional form, with ϑ is the passively transported quantity by the fluid, with Re Reynolds number, Sc Schmidt number,as:

$$\mathrm{Re} = \frac{UL}{\nu}$$

$$\mathrm{Sc} = \frac{\nu}{D}$$

with U fluid velocity in one directions, L characteristic dimension, ν cinematic viscosity and D trasported scalar diffusivity.

In adimensional form:

$$\partial_t \vartheta + \nabla \cdot (\mathbf{u}\vartheta) = D \nabla^2 \vartheta.$$

Input files:

- param_ns_scal.txt → contains the following parameters:

N	points number in directions 1, 2;
$N3$	points number in directions 3;
Δt	temporal step;

[1] Vectors are in bold.

$Ntot$	total steps number;
$Nsalva$	steps number saving;
Re	Reynolds number;
$nomefile$	file name with three velocity components (each for every component) to use as initial conditions.
N	points number in directions 1, 2;
$N3$	points number in directions 3;
Δt	temporal step;
$Ntot$	total steps number;
$Nsalva$	steps number saving;
Re	Reynolds number;
$nomefile$	file name with three velocity components (each for every component) to use as initial conditions.
$Nscal$	number of scalars;
$SC1$	Schmidt number of the firts scalar;
SCn	Schmidt number of the n-esimo scalr

- nomefile(1), nomefile(2), nomefile(3) $\rightarrow$ components velocity file.

If *Izero=0* initial conditions for the passive scalar are whole generated[2]:

$$\vartheta_1(x_1,x_2,x_3) = \begin{cases} 1 & \text{se} \quad x_3 < L/2 \\ \\ 0 & \text{se} \quad x_3 > L/2 \end{cases}$$

$$\vartheta_2(x_1,x_2,x_3) = \begin{cases} 0 & \text{se} \quad x_3 < L/2 \\ \\ 1 & \text{se} \quad x_3 > L/2 \end{cases}$$

$$\vartheta_3(x_1,x_2,x_3) = 1 \qquad \forall x_1,x_2,x_3$$

$$\vartheta_k(x_1,x_2,x_3) = \vartheta_1(x_1,x_2,x_3) \quad \forall k \geq 4$$

If *Izero>0* initial conditions are read by files scalk-xy.old, where k is the reference number of the scalar and xy value *Izero* with two digit.
Outputs files:

- u1.xy.bin, u2.xy.bin, u3.xy.bin;
- scalk-xy.bin $\rightarrow$ scalar saved for each instant *xy*.

[2]Con L it is defined the dimension of the domain in the direction 3.

dns_genera.x

Outputs data in pseudo-random way in a cube with N^3 points. The velocity field is:

$$\mathbf{u} = \nabla \wedge \mathbf{A}$$

with $\mathbf{A}$ calculated by means generation of the ramndom number.
So:$\nabla \cdot \mathbf{u} = 0$.
We can write:

$\text{Re}\left[\hat{A}_i\left(k\right)\right] = \alpha\left(k\right)\cos\left(2\pi\xi\right)$

$\text{Im}\left[\hat{A}_i\left(k\right)\right] = \alpha\left(k\right)\sin\left(2\pi\xi\right)$

where:

$\hat{A}_i$ is the Fourier transform of A_i;

$\alpha\left(k\right)$ is the magnitude calulated by means the energy spectrum as:

$$\alpha\left(k\right) = \sqrt{\frac{E\left(k\right)}{k}}$$

ξ is the random number computed by an uniform distribution with an uniform probability density densità in $[0, 1]$ range.

Spectrum is chosen as fig.2.

$$\begin{cases} E\left(k\right) \sim k^S & \text{per} \quad k < k_0 \\ E\left(k\right) \sim k^{-5/3} & \text{per} \quad k < k < k_{max} \end{cases}$$

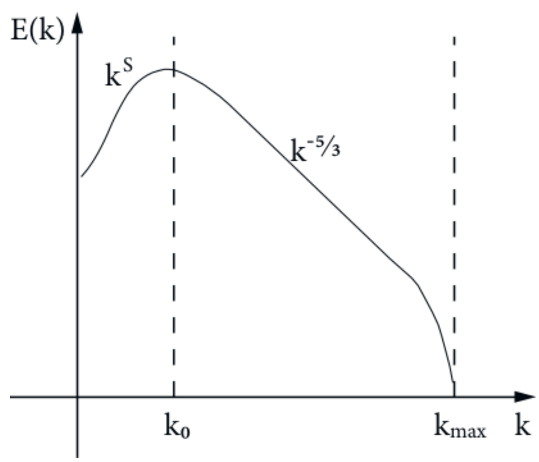

Figura 2: Energy spectrum.

k_0 value is chosen by user, while k_{max} can be chosen as $N/2$ or $N/3$.

This scalar code can be executed with *./dns_genera.x*.
The parameters must be input by clipboard. Outputs files are: u1.99.bin, u2.99.bin, u3.99.bin.

dns14_genera.x

Similar to dns_genera.x but is a parallel code. Parameters are written in file param_genera.txt, that contains:

N	number of points;
k_0	energy spectrum maximum;
S	low wavenumber spectrum exponent;
$jtutti$	$1 \rightarrow k_{max} = N/2 \quad 0 \rightarrow k_{max} = N/3$;
u'_{rms}	root mean square of velocity fluctuations ($E = \frac{3}{2}u'^2$);
$nomefile$	files name with three velocity components of the generate field;
1	not used parameter.

---

dns14_spettro.x

Executable file for computing of spectrum and scales.
Input files:

- param_ns.txt;
- uxy.bin $\rightarrow$ simulation data, con xy temporal instant;
- in.skder$\rightarrow$ parameters:

$Imin$	index of the first computing file;
$Imax$	index of the last computing file;
Ip	step between files ($1\rightarrow$ all the files $\quad n\rightarrow$each file every n).

Outut files:

- spettro_xy.txt $\rightarrow$ spectrum;
- decadimento.txt $\rightarrow$ contains energy, dissipation and scales in function of the time.

dns_ske_u.x

It computes energy, skewness, kurtosis for three velocity components. It executes average in planes (x_1, x_2), as showed in fig.3.

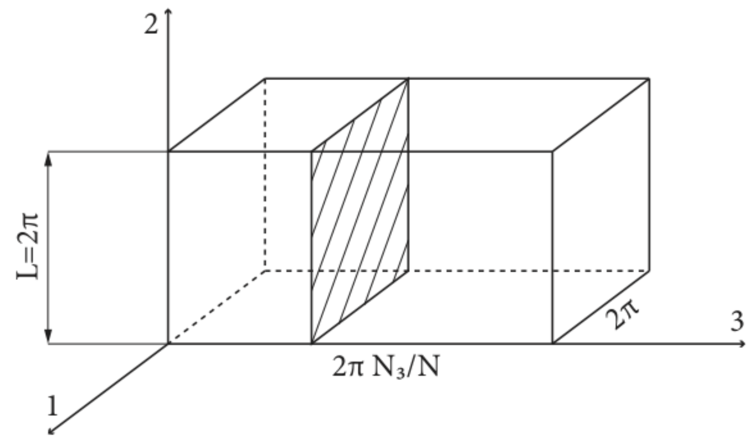

Figura 3: Calculation of the average property on a plane in $x_3 = \text{cost}$.

$$\bar{f}(x_3) = \frac{1}{L^2}\int_0^L\int_0^L f(x_1, x_2, x_3)\, dx_1 dx_2$$

Inputs file

- param_ns.txt;
- uxy.bin → simulation data, with xy temporal instant;
- in.skder → parameters:

$Imin$	index of the first file to compute;
$Imax$	index of the last file to compute;
Ip	step between of files (1→all the files n→each file every n).

Outputs files:

- ske_1_xy.txt, ske_2_xy.txt, ske_3_xy.txt → files with four columns (in order x_3, second moment $\overline{u'^2}$, skewness S and kurtosis K); It is written the scalar component of $\mathbf{u}$ (1, 2 o 3).

Nota It can be $N \neq N_3$.

dns14_ske_scal.x

dns14_ske_scal_u.x

They compute passive scalar statistics.
Output files give the following values:

- x_3, $\overline{\vartheta}\,(x_3)$, $\overline{\vartheta'^2}\,(x_3)$, $S_{\vartheta'}$, $K_{\vartheta'}$;
- x_3, $\overline{u\vartheta'}\,(x_3)$, $\overline{u^2\vartheta'}\,(x_3)$.

The avergae is calculated on planes at $x_3 =$ cost as dns14_ske.x.

3 Visualizzations

To visualize data we use the software VisIt. It needs convert files into the format .vtk (*visual toolkit*):

1. from initial file .bin to file .bin with each antitransformed variable in space for every file.
   grid.out is generated with coordinates.

2. passaggio dal file .bin al file .vtk tramite the source code bin2vtk. convert files .bin into .vtk

To execute the code it is used the following sintax:

bin2vtk nomefile -float -vname NOME