Instructions to run the case

Please find below the steps required to configure and run the case, as well as how to ensure results are still valid between runs after the code has been modified.

Enabling the case

SOD2D cases are configured within the code itself. To enable the TGV case, the file `src/app_sod2d/sources/sod2d.f90` must be set as follows:

```
! 0: disabled, 1: enabled
! Only 1 case can be enabled at a time
#define _tgv_ 1
#define _tgv_multi 0
#define _tgv_comp 0
#define _tgv_incomp 0
#define _channel_ 0
#define _channel_incomp 0
#define _bluff_ 0
#define _bluff_incomp 0
#define _bluff3d_ 0
#define _bluff3d_incomp 0
#define _bl_ 0
#define _abl_ 0

program main
...
```

Configuring the case

A default configuration of the TGV case can be found in `src/lib_mainBaseClass/sources/TGVSolver.f90`. We recommend that the following be set:

```
    subroutine TGVSolver_initializeParameters(this)

        ...

        ! Case path, usually the same as the executable path
        write(this%mesh_h5_file_path,*) ""

        ! Case name, no partition number or extension
        write(this%mesh_h5_file_name,*) "cube"

        ! Path of results files
        write(this%results_h5_file_path,*) ""

        ! Name of results files, written as results_case-part_step.hdf
```

```fortran
        write(this%results_h5_file_name,*) "results"

        ...

        ! Tracking parameters for the TGV case
        this%doGlobalAnalysis = .true.

        ! Measures time/time-step and the average time across all steps
        this%doTimerAnalysis = .true.

        ! Saves a result file before any computation is carried out
        this%saveInitialField = .false.

        !---------------------------------------------
        !  -------------  I/O params -------------

        ! Number of time steps to perform
        this%final_istep = 500001

        ! Maximum physical time
        this%maxPhysTime = 20.0_rp

        ! When to start logging, and at what frequency (includes the global
analysis)
        this%save_logFile_first = 1
        this%save_logFile_step  = 10

        ! When to start outputting results, and at what frequency
        this%save_resultsFile_first = 1
        this%save_resultsFile_step = 1000

        ...

        ! numerical params

        ! Activate LES model (DNS otherwise)
        flag_les = 0

        ...

        ! Control the time-step size based on safety factor
        this%cfl_conv = 0.5_rp
        this%cfl_diff = 0.5_rp

        ...
```

Finally, the element order in use must be informed to SOD2D by setting the parameter in
src/lib_sod2d/sources/mod_constants.f90:

```fortran
  module mod_constants
```

```
  implicit none

  ! Controls floating point precision, fp32 by default
  integer(4), parameter :: rp = 4 !(4/8)

  ...

  !
  ! Element characteristics
  !

  ! set the order to the same used in gmsh (p=4 in this case)
  integer(4), parameter :: porder=4

  ...
```

These settings are sufficient to run the case using the explicit RK4 scheme.

Running the case

Put the executable in the same folder as the mesh and run:

```
mpirun -np N ./sod2d
```

Where N is the number of MPI processes to use. The results will be written to the folder specified in `this%results_h5_file_path`. The number of processors is determined by the number of partitions in the mesh file. For example, for `cube-1.hdf`, N=1, for `cube-2.hdf`, N=2, etc. In MN4, the typical slurm configuration can be used.

Validating the results

For the TGV case, the global analysis will produce a series of metrics that can be used to ensure results are still valid after code modifications. These are written to the file `analysis_casename-N.dat` in the results folder. The file contains 6 columns:

1. Time
2. Kinetic energy
3. Vorticial enstrophy
4. Dilational enstrophy
5. Total enstrophy
6. Maximum Mach number in domain

Across different runs, it is important to verify that the values for 2, 5 and 6 are the same despite code modifications to accommodate OpenMP (3 and 4 are guaranteed if 5 is OK). The `gnuplot` tool can be used to produce plots of these variables:

```gnuplot
gnuplot
plot "analysis_casename-N.dat" using 1:2 with lines title "Kinetic energy", \
     "analysis_casename-N.dat" using 1:5 with lines title "Total enstrophy", \
     "analysis_casename-N.dat" using 1:6 with lines title "Maximum Mach number"
```

We recommend storing the analysis file of each run separately so that runs can be compared later on.

Performance analysis

SOD2D also logs out the time taken to perform a time-step and the average time across all time-steps. This is written to the file `timer_casename-N.dat` in the results folder. The file contains 3 columns:

1. Time-step number
2. Time taken to perform the time-step
3. Average time across all time-steps

Similarly to the global analysis, these can be plotted using `gnuplot`:

```gnuplot
gnuplot
plot "timer_casename-N.dat" using 1:2 with lines title "Time-step time", \
     "timer_casename-N.dat" using 1:3 with lines title "Average time"
```