MFiX Tutorial
Pulsating 2D fluidized bed

Sathish Sanjeevi and Jeff Dietiker

National Energy Technology Laboratory, USA

November, 2019

1 Introduction

In this tutorial, a pulsating 2D two-fluid-model (TFM) fluidized bed is simulated using user-defined functions (UDFs). The objectives of this tutorials are:

- Use keyframe data to compute the jet inlet velocity at different timesteps.
- Apply interpolated inlet velocity at each timestep and generate a pulsating inflow boundary condition.

2 Keyframes

This section provides a general description of keyframe data usage. Keyframes are tables with numerical data to be used in simulations, and are currently implemented through UDFs. Such tables are useful to perform dynamic simulations, for example, a time dependent inflow boundary. Each keyframe is titled in the form `data_kf_<ID>.txt`. Here, ID is a four-padded integer representing the keyframe ID, which will typically be associated with the same boundary condition ID (BCIDs), although ID and BCID are independent from each other (it is the user's responsibility to decide how to use the keyframe data). The user-defined keyword `read_kf(ID)` allows reading a given keyframe file. Each keyframe file has the following structure:

```
nrows, nvar
interpolation_type
!  header
 x_1  y_{11} ...  y_{1n}
 .    .           .
 .    .           .
 x_m  y_{m1} ...  y_{mn}
```

Here, `nrows(=m)` is the number of data rows in the keyframe file, and `nvar(=n)` is the number of dependent variables. The first column defines a list of increasing values for the independent variable `x`. In this tutorial, we are using keyframes to define transient data, so `x` correspond to time. Other columns define values of the dependent variables `y`, for example velocity at corresponding time. The data in all columns are floating point (real) scalars. For vector quantities, users need to specify the 3 components of the vector using 3 columns (for example, if two scalars and one vector are defined, then `n=5`). `interpolation_type` defines the available interpolation types, namely `linear` or `step`. Once the keyframe data is defined, users can evaluate `y` at any `x`-value between the data points. Two interpolation schemes are available: 1) Linear interpolation, where `y` is assumed to vary linearly between two consecutive points (interval $[x_i; x_{i+1}]$) , and 2) Step function which returns the constant y-value y_i when the `x` value lies within the interval $[x_i; x_{i+1}]$. The actual keyframe reading (`read_keyframe_data`) and interpolation (`interpolate_keyframe_data`) functions are located in `usr_mod.f` file.

2.1 A keyframe example

The following example explains the actual value of a variable `var1` depending on the `interpolation_type`. Consider the following keyframe file:

```
6, 1
<interpolation_type>
!x                    var1
0.0                   0.5
2.0                   0.0
4.0                   2.5
6.0                   6.0
8.0                   4.0
10.0                  3.0
```

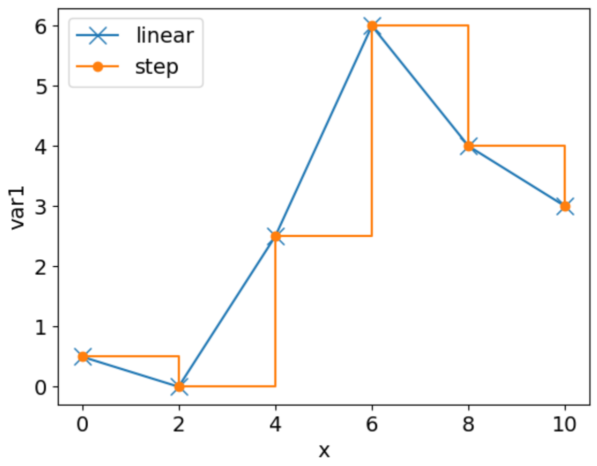


Figure 1: Profiles of `var1` when `interpolation_type = linear` or `step`.

The actual profile of `var1` is illustrated in figure 1 depending on `interpolation_type = linear` or `step`. Symbols represent the keyframe data, lines represent the interpolated data. Values of `var1` are kept constant outside the `x`-range. For example, consider the following keyframe file:

```
3, 1
<interpolation_type>
!x                  var1
4.0                 2.5
6.0                 6.0
8.0                 4.0
```

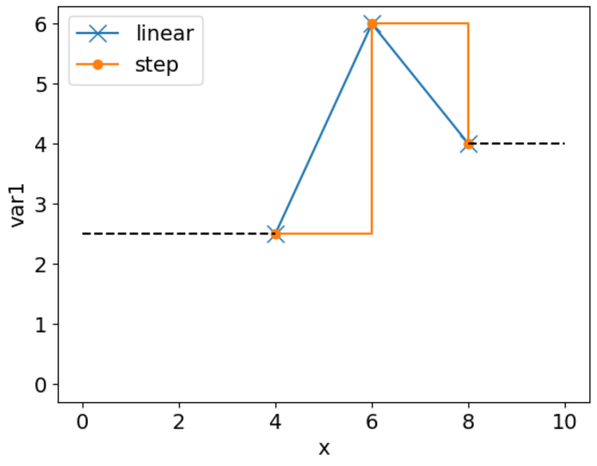


Figure 2: Value of `var1` outside the keyframe `x`-range is indicated with dashed line. Behaviour outside the specified keyframe `x`-range is independent of `interpolation_type`.

The full profile of `var1` in actual simulations is shown in figure 2. As seen, the first and last values of the keyframe values are extended outside the `x`-range (dashed line), both for `linear` and `step` interpolation schemes.

3 Problem setup

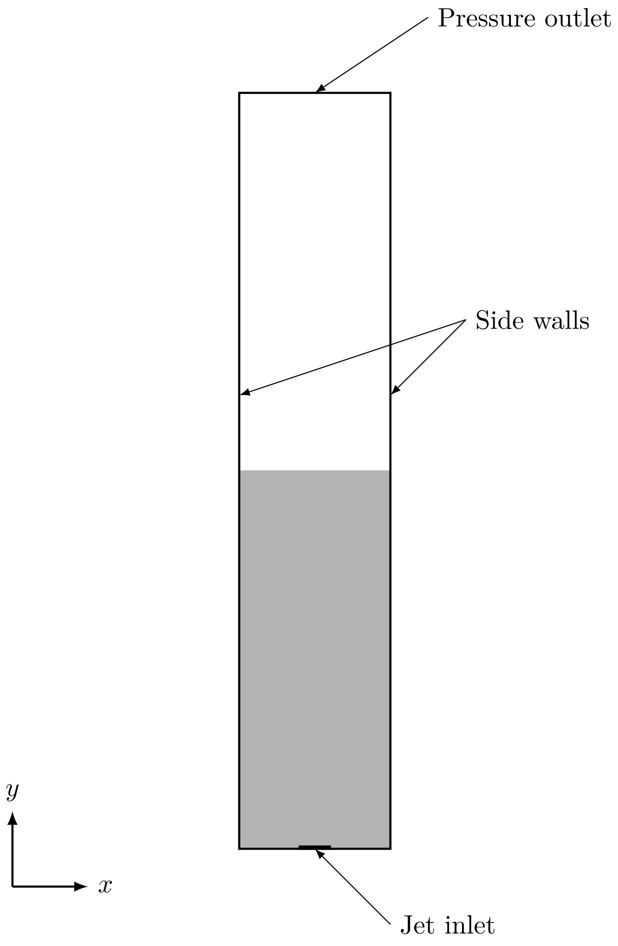


Figure 3: Schematic of the 2D fluidized bed. The domain width is 0.2 m and height is 1 m. The bottom inlet is of width 0.04 m. Bottom-half of the domain is initialized with particles (grey region).

This tutorial performs a two fluid model (TFM) simulation of fluidized bed with a pulsating jet inlet. An inlet boundary condition is located at the bottom. The inlet boundary condition is the boundary of interest for the keyframe function with its BCID=1. The time and the corresponding inlet velocities are given in `data_kf_0001.txt`. The functions compute the appropriate velocity for each timesteps and this is assigned to respective boundary cells for each timestep for the inlet. Below, we explain the function of each file in this tutorial:

- `usr_mod.f` - contains all the necessary functions for reading and interpolating keyframe data.
- `usr0.f` - calls the function to read keyframe files and allocates required variables.

- `usr1.f` - utilizes the interpolated data and assigns it to y-component of gas inlet velocity (by using `set_bc0_vel_inflow` subroutine).
- `usr_init_namelist.f` and `usrnlst.inc` - Both files let the solver and GUI known about the new keyword `read_kf` which is an addition in this tutorial. This is a Boolean flag used in the UDFs to read a given keyframe file.

Interested users can study the above files to understand how the implementation works, and adapt it to their specific needs. Below is the keyframe file (`data_kf_0001.txt`) used in this tutorial:

```
8, 1
step
! time    var1
0.00      2.00
0.25      0.00
0.50      2.00
0.75      0.00
1.00      2.00
1.25      0.00
1.50      2.00
2.00      0.00
```

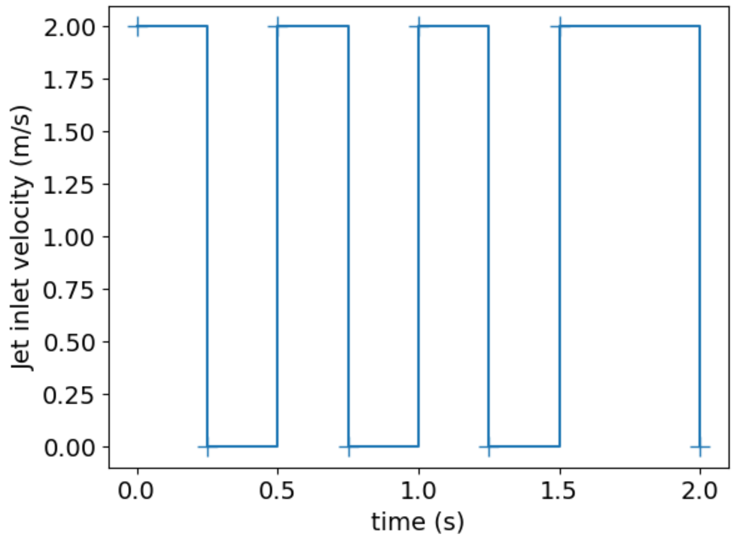
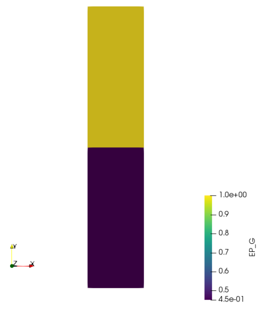


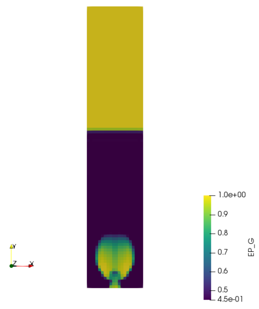
Figure 4: Inlet jet velocity for the pulsating fluidized bed. Markers (+) indicate the keyframe data and lines indicate the interpolated data (step function).

The plot of the keyframe file is shown in figure 4.

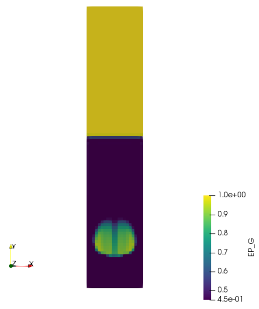
4 Results



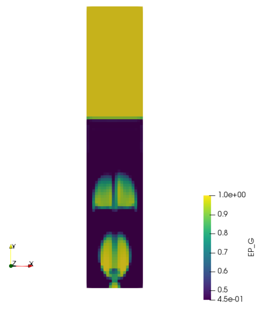
(a) $t = 0$s



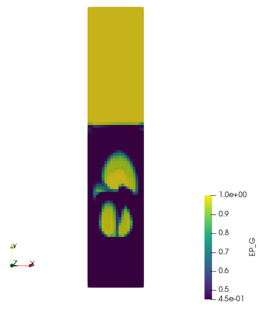
(b) $t = 0.25$s



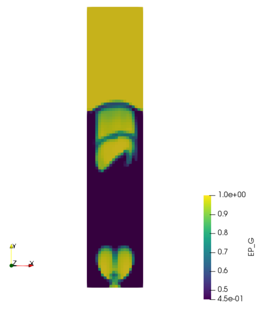
(c) $t = 0.5$s



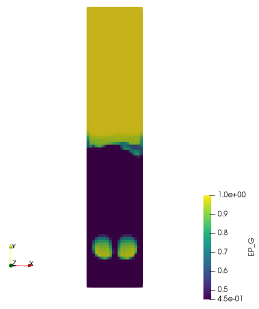
(d) $t = 0.75$s



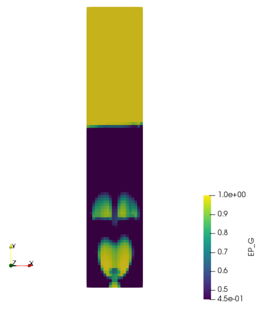
(e) $t = 1$s



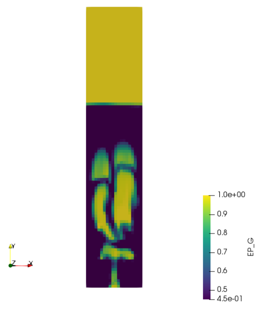
(f) $t = 1.25$s



(g) $t = 1.5$s



(h) $t = 1.75$s



(i) $t = 2$s

Figure 5: Snapshot of void fraction (ϵ_g) of the fluidized bed at different timesteps. The formation of the bubbles and its frequency matches with the data in figure 4.

The results of the pulsating fluidized bed simulations are plotted in figure 5. It can be observed that the pulsating behaviour matches well with our input data plotted in figure 4.

5 Notes

- Since this tutorial involves UDFs, the custom solver must be built prior to running the simulation.
- Once the custom solver is built, users can modify and save the keyframe files to adjust the jet inlet velocity. There is no need to build the custom solver when modifying keyframe data.
- The user-defined keyword `read_kf` is accessible through the GUI Advanced pane. To completely turn off the keyframe data for the inlet, set `read_kf(1)=False`. This will set the default zero velocity for the bottom jet inlet.