

Couette-Poiseuille Flow Computations

Aakash PATIL ¹

¹Master's Student in Turbulence
Ecole Centrale de Lille, FR 59650

aakash.patil@master.centraledelille.fr

1. Introduction and Theory

Couette flow is a classic example of shear-driven fluid motion. It can be described as the flow of a viscous fluid between two parallel surfaces(walls), one of which is moving tangentially relative to the other. The flow is driven by virtue of viscous drag force acting on the fluid, but may additionally be motivated by an applied pressure gradient in the flow direction which is then a case combined with the Poiseuille flow. Following figure represents a simplified geometry for these type of flows:

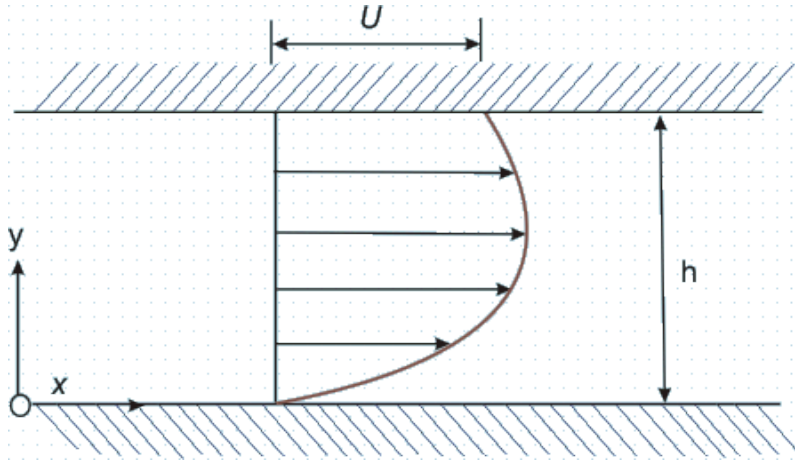


Figure 1. The representation of geometry.

The governing equations for starting the computations of the Couette-Poiseuille Flow could be derived from the Reynolds-averaged Navier-Stokes equations written as:

$$\frac{\partial \bar{U}_i}{\partial x_i} = 0 \quad (1)$$

$$\frac{\partial \bar{U}_i}{\partial t} + \bar{U}_j \frac{\partial \bar{U}_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{P}}{\partial x_i} + \nu \frac{\partial^2 \bar{U}_i}{\partial x_j^2} - \frac{\partial \overline{u_i u_j}}{\partial x_j} \quad (2)$$

where \bar{U}_i represents the i -th component of the average fluid velocity, u_i is the velocity fluctuation, $\overline{u_i u_j}$ are the components of Reynolds stresses, ν is the kinematic viscosity, ρ is density of the medium, and \bar{P} is the mean pressure. The subscript i is a free index which can take on the values 1, 2, and 3. Thus, both of the above equations in reality are three separate equations one for each dimension.

Our main assumption is that Couette-Poiseuille flow is fully developed, which justifies the use of following assumptions:

(a). The flow is steady, so the time derivatives are zero. (b). The top-wall moves in the x direction, so the significant mean velocity component will be in x direction. This means we can neglect the y and z components of mean velocity. (c). The flow is uniform in x direction, so the gradients of velocity in x direction are zero. Also the z dimension of the geometry under consideration is long enough to assume that the gradients along this direction are zero. (d). As we are assuming that mean flow in y and z direction negligible, the pressure gradients in those direction could be neglected. The only significant pressure gradient would be in x direction.

Using turbulent viscosity, ν_t , the turbulent shear stress is written as

$$\tau_{turb} = -\rho \overline{u_1 u_2} = -\rho \nu_t \frac{\partial \bar{U}_1}{\partial x_2} \quad (3)$$

So finally, we would obtain the required equation:

$$\frac{\partial}{\partial x_2} \left[(\nu + \nu_t) \frac{\partial \bar{U}_1}{\partial x_2} \right] = \frac{1}{\rho} \frac{\partial \bar{P}}{\partial x_1} \quad (4)$$

1.1. Discretizing the equations

The finite volume method (FVM) is a discretization technique for partial differential equations, especially those arising from the physical conservation laws. FVM uses a volume integral formulation of the problem with a finite partitioning set of volumes to discretize the equations. Using finite volume method we divide our physical domain into discrete number of volumes. Integrating the equation (4) for a one discrete volume from south-face(S) to north-face(N):

$$\int_S^N \frac{\partial}{\partial x_2} \left[(\nu + \nu_t) \frac{\partial \bar{U}_1}{\partial x_2} \right] dx_2 = \int_S^N \frac{1}{\rho} \frac{\partial \bar{P}}{\partial x_1} dx_2 \quad (5)$$

For turbulent viscosity:

$$\int_S^N \nu_t = \int_S^N l_m^2 \left| \frac{\partial \bar{U}_1}{\partial x_2} \right| \quad (6)$$

l_m is from the mixing length model, related to the size of most occurring eddies in flow, which could be considered as primarily responsible for mixing. It is expressed as:

$$l_m = l_0 \left[1 - \exp \left(-\frac{y}{A} \right) \right] \quad (7)$$

Where,

$$A = \frac{26\nu}{U_\tau} + 2 \frac{l_0}{H} = 0.21 - 0.43 \left[1 - 2 \left(\frac{y}{H} \right) \right]^4 + 0.22 \left[1 - 2 \left(\frac{y}{H} \right) \right]^6 \quad (8)$$

Here, U_τ is the friction velocity at wall, given by:

$$U_\tau = \sqrt{\nu \left(\frac{\partial \bar{U}_1}{\partial x_2} \right)} \quad (9)$$

Putting all the equations together and then rearranging them as the parts of a tridiagonal system, we obtain a discretized equation at node i , expressed as:

$$A(U_1)_{i-1} - B(U_1)_i + C(U_1)_{i+1} = D_i \quad (10)$$

where,

$$\frac{(\nu + \nu_t^S)}{(x_2)_i - (x_2)_{i-1}}(U_1)_{i-1} \implies \text{Lower Diagonal}(A) \quad (11)$$

$$\left[\frac{(\nu + \nu_t^N)}{(x_2)_{i+1} - (x_2)_i} + \frac{(\nu + \nu_t^S)}{(x_2)_i - (x_2)_{i-1}} \right] (U_1)_i \implies \text{Middle Diagonal}(B) \quad (12)$$

$$\frac{(\nu + \nu_t^N)}{(x_2)_{i+1} - (x_2)_i}(U_1)_{i+1} \implies \text{Upper Diagonal}(C) \quad (13)$$

$$\frac{1}{\rho} \frac{\partial P}{\partial x_1} \frac{(x_2)_{i+1} - (x_2)_{i-1}}{2} \implies \text{R.H.S. of Tridiagonal System}(D) \quad (14)$$

The boundary conditions would be implemented as follows:

$$(-A - B)(U_1)_1 + C(U_1)_2 = D_1 \quad (15)$$

$$A(U_1)_{N-1} - (B + C)(U_1)_N = D_N - C(2U_{wall}) \quad (16)$$

The relations in this section would be adopted for N number of such discrete volumes in the flow domain to solve a system of linear equations. This FVM formulation is then applied in a FORTRAN90 code to solve and obtaining various flow parameters.

2. Description of the Computation Code

Using the theory described in the previous section, a FORTRAN 90 code is written to compute the flow parameters for different input conditions. The main input parameters of the code are: the velocity of the moving wall ($V_{wall}=V_w=V_b$), pressure gradient ($\frac{\partial P}{\partial x}$), viscosity(ν), density (ρ) and height of the channel(H) for different grids from homogeneous ($\alpha = 0$) to stretched grid ($\alpha < 0.05$) with a specified number of grid points N . The parameters computed are: $u(y)$, $u_{\tau 1}$, $u_{\tau 2}$ and bulk velocity U_q .

Along with the main program *cpFlow.f90* to call all the routines, the code contains 4 modules:

1. *module_grid.f90* which generates a grid with stretching. This module was provided by Dr. Laval and I made a minor change in this module to accommodate the height of channel.

2. *module_parameters.f90* has all the required constant parameters as well as input conditions for simulating different flow cases.

3. *module_solver.f90* has a routine to implement all the theory discussed in this report. Depending on the case, this routine performs the required computation and stores the data for further analysis.

4. *module_triSolve.f90* has a properly functioning routine for solving a tridiagonal matrix using Thomas algorithm.

For automatic plotting of the data and proper organization of the code, the following directory structure is implemented:

```
cpFlow
├── src
│   ├── cpFlow.f90
│   ├── module_triSolve.f90
│   ├── module_solver.f90
│   ├── module_grid.f90
│   ├── module_parameters.f90
│   └── Makefile
├── ExperimentalData
│   └── <casewise experimental data files>
├── SimulatedData
│   ├── dataFiles_previousRun
│   │   └── <casewise simulation data files>
├── GeneratedPlots
│   ├── plots_previousRun
│   │   └── <plots generated for report>
└── README.md
```

The code can be compiled by using *Makefile*. If automatic plotting is enabled from the *parameters.f90*, Python2.7 with numpy and matplotlib has to be installed before.

3. Results and Discussion

In this section, the computed results are presented for 18 different cases and a comparison is made with the experimental results for the same 18 cases obtained by [Telbany and Reynolds 1980] and [Gilliot-Ottavy 1997].

The input parameters for case number 1 to 15 are from [Telbany and Reynolds 1980] whereas those for case number 16 to 18 are from [Gilliot-Ottavy 1997].

3.1. Velocity Profile

3.1.1. Varying Grid Stretching

The velocity profile was computed by varying the grid stretching parameter, alpha (α), applied to a uniform grid and keeping a fixed number of grid points (N). α decides how to stretch the grid in y-direction i.e. $\alpha = 0$ means a homogeneous grid. Figure (2) shows a comparison of velocity profiles for Case 2, 6, 15, and 18 computed for stretched grid ($\alpha = 1E-01$) and homogeneous grid ($\alpha = 1E-15$). It can be observed that the velocity profile is in close agreement with the experimental data for $\alpha = 1E-01$. It was also observed that upto $\alpha = 1E-06$, the experimental data and the simulated data was in a close agreement with the computations for majority of cases. But for a few cases, where convergence was not reached despite a significantly large convergence criteria, the velocity profile does not coincide with the experimental data. It can be concluded that good results are obtained for stretched grid as compared to the homogeneous grid. The reason for this could be attributed to the fact that higher gradients of velocity are present near the wall region and a stretched grid near the wall would take into account this effect.

3.1.2. Varying the Number of Grid Points

The velocity profile was computed by varying the number of grid points, N , in y-direction and by keeping a fixed value of $\alpha = 1E-6$. Figure (3) shows a comparison of velocity profiles for Case 2, 6, 15, and 18 computed for number of points $N = 10001$, $N = 1001$, and $N = 101$. It can be observed that the velocity profile is not much affected by the number points in a stretched grid and that the computations are in close agreement with the experimental data for most of the cases. It can be concluded that the velocity profile becomes smoother by increasing the number of grid points.

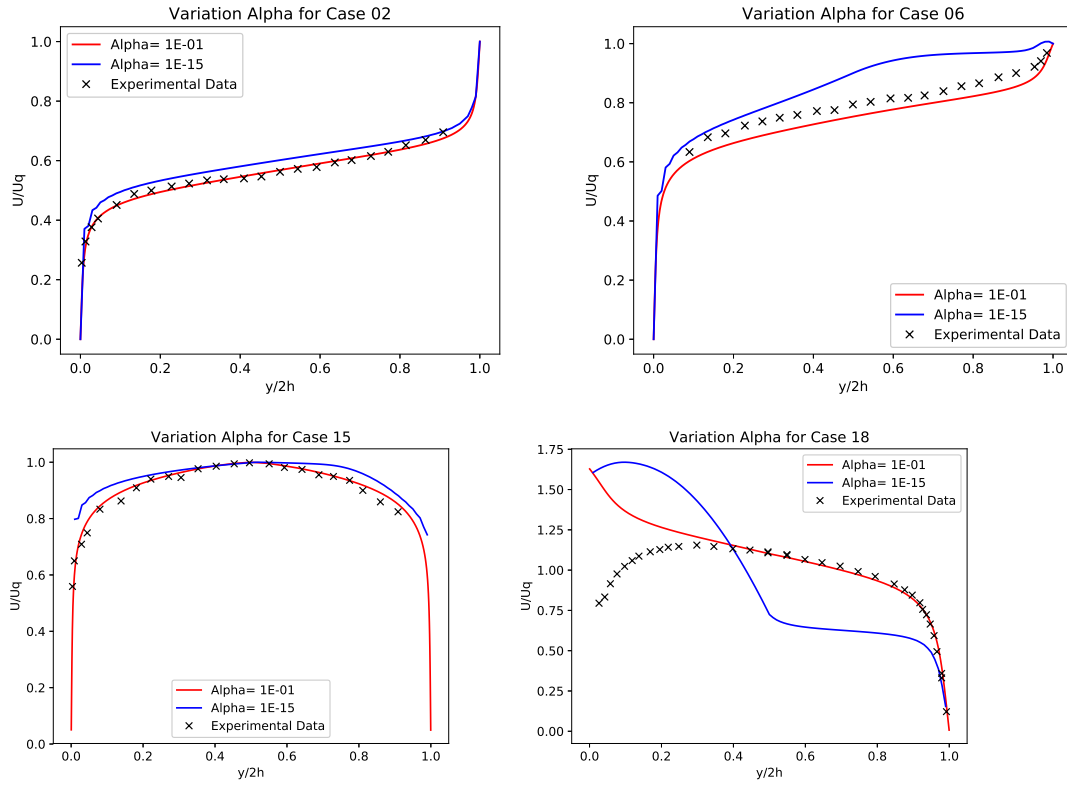


Figure 2. Comparison of velocity profiles for Case 2, 6, 15, and 18 by varying grid stretching(α)

3.2. Convergence

Total number of iterations taken by the computations to converge is compared in this subsection. A two convergence criteria were set, one with the maximum allowed error between successive computations and second with maximum number of iterations to converge. The computations were terminated if neither of the convergence criteria was reached. As the computation time for each case was very less, a significantly large convergence criteria was used. Figure (4) shows the comparison of number of iterations when the convergence was reached by varying the grid stretching parameter, α . It can be observed that almost all the cases, except case number 8, were converged for $\alpha = 1E - 01$ i.e. for a stretched grid.

Figure (5) shows the comparison of number of iterations when the convergence was reached by varying the number of grid points, N . It can be observed that most of the cases, were converged for $N = 101$.

3.3. Maximum Velocity and Mean Velocity

Maximum velocity and average velocity was computed for each case and compared with the experimental values. For some of the cases which were converged, the computed values of both the maximum velocities and average velocities are in agreements with their respective experimental values. Figure (6) shows the plot of casewise comparison of computed maximum velocity and mean velocity with their experimental values. The

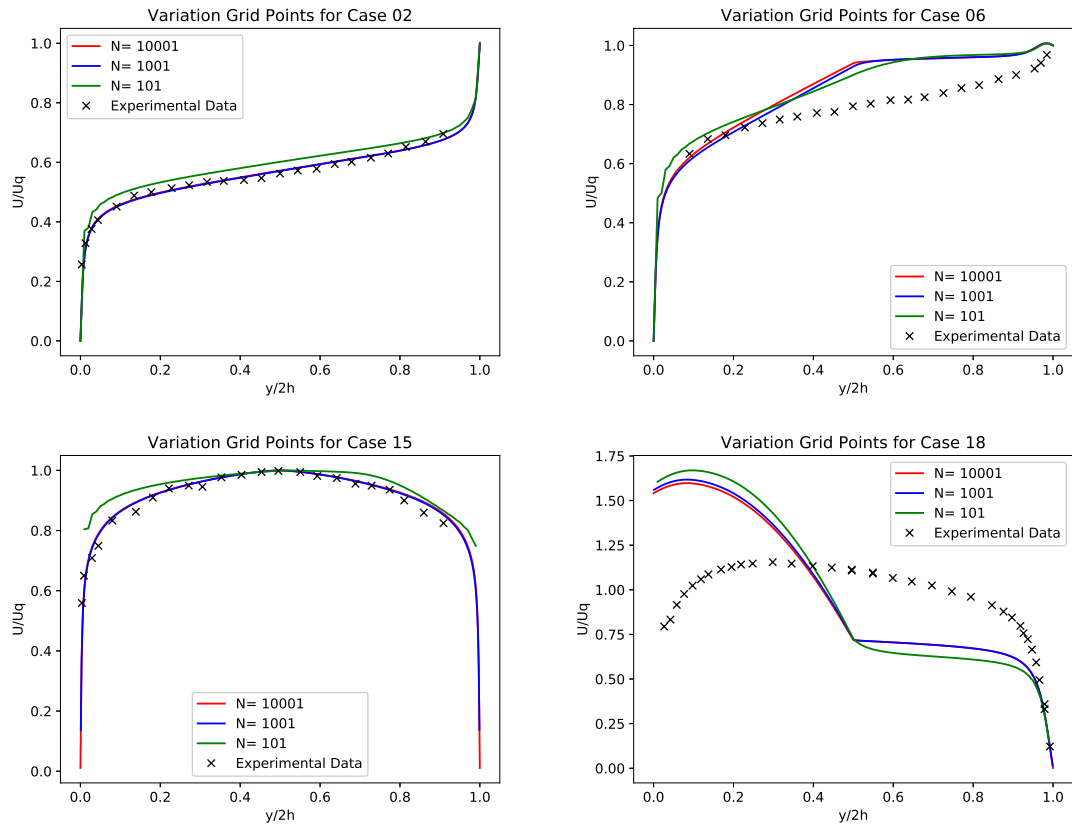


Figure 3. Comparison of velocity profiles for Case 2, 6, 15, and 18 by varying number of grid points (N)

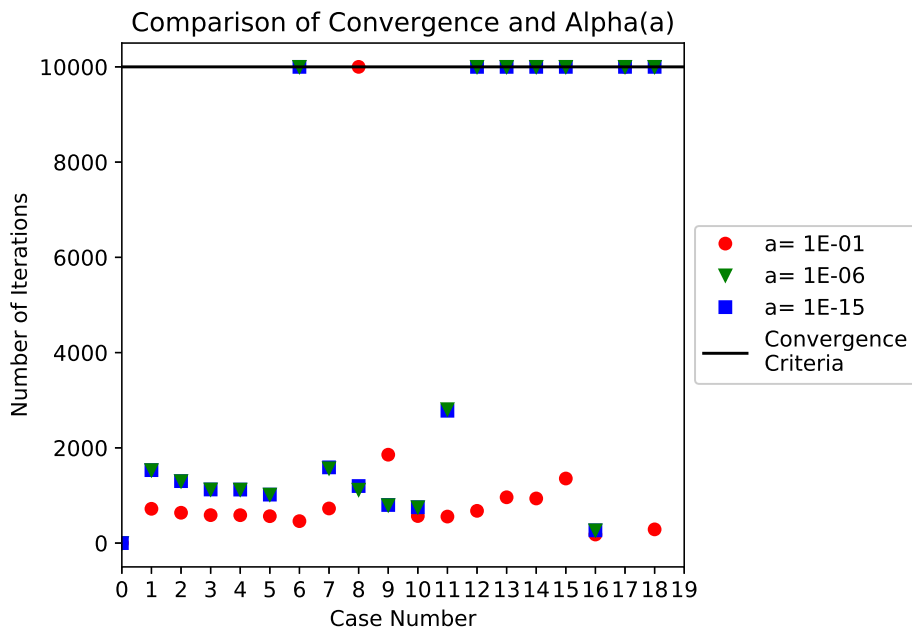


Figure 4. Comparison of number of iterations to reach convergence when the grid stretching(α) was varied.

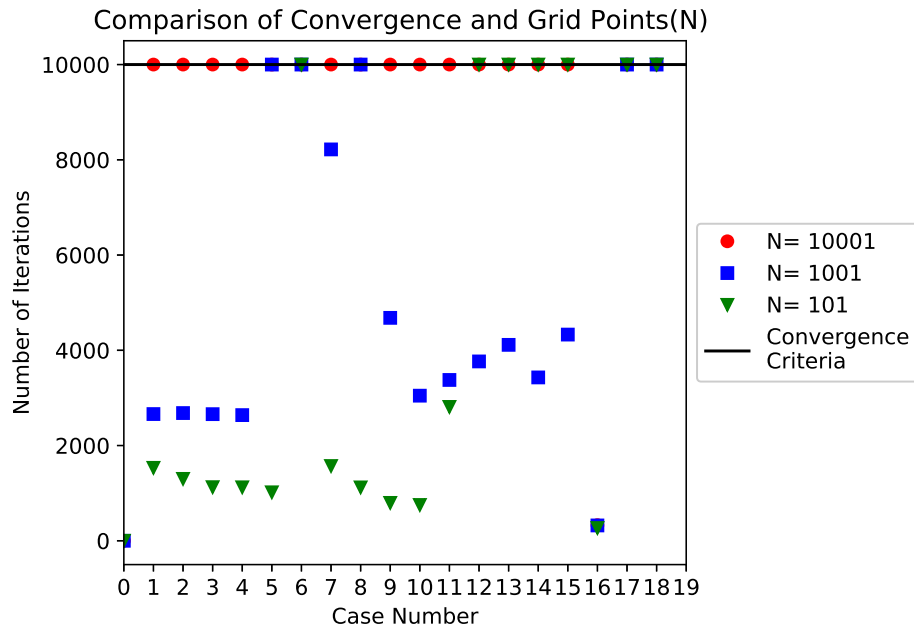


Figure 5. Comparison of number of iterations to reach convergence when the number of grid points (N) was varied.

maximum velocities and average velocities are written in a tabulated form which is in the appendix as Table(1).

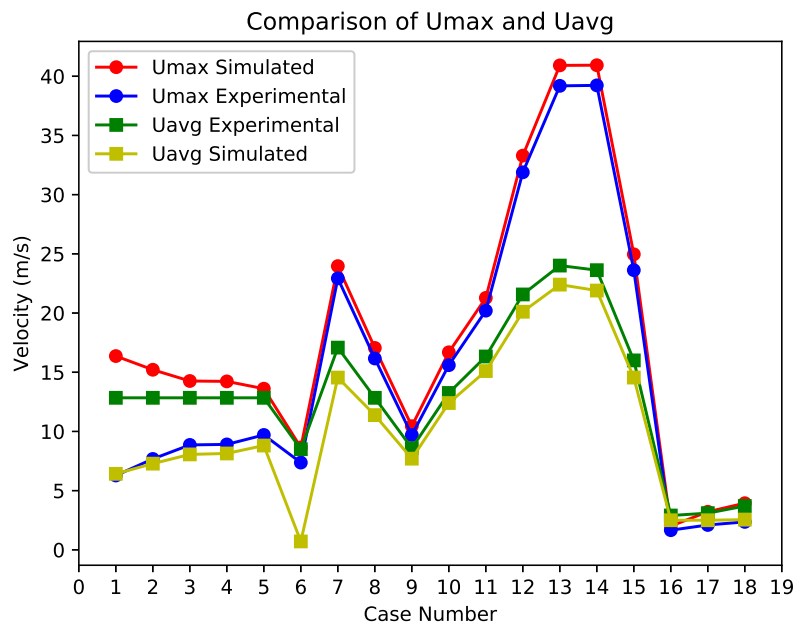


Figure 6. Casewise comparison of maximum velocity and mean velocity.

3.4. Friction Velocity

The friction velocity for the high-stress wall ($U_{\tau 1}$) and low-stress wall ($U_{\tau 2}$) was computed for each case and compared with the experimental values. For most of the cases, the computed values of $U_{\tau 1}$ are in agreement with their respective experimental values whereas a majority of $U_{\tau 2}$ values are not matching with their corresponding experimental values. Figure (7) shows the plot of casewise comparison of computed friction velocities for the high-stress wall and low-stress wall with their experimental values. These values are also written in a tabulated form, which is in the appendix as Table(2).

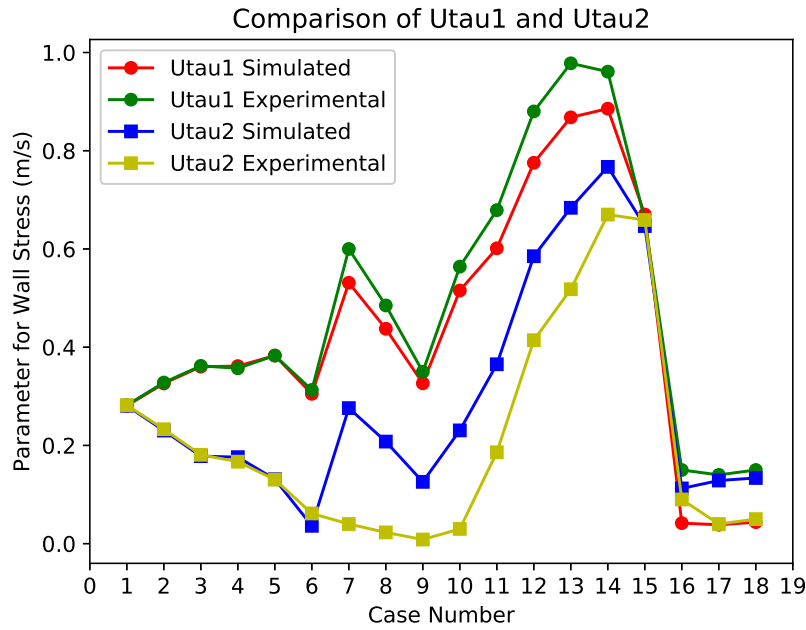


Figure 7. Casewise comparison of friction velocity parameter for high-stress and low-stress wall.

3.5. Dimensionless Wall Parameters

The velocity is normalized by $U_{\tau 2}$ to make it dimensionless, which is given as:

$$u^+ = \frac{U}{U_{\tau 2}} \quad (17)$$

Similarly, distance in y-direction is normalized by $\frac{U_{\tau 1}}{\nu}$ to make it dimensionless and given as:

$$y^+ = y \frac{U_{\tau 1}}{\nu} \quad (18)$$

These values of u^+ and y^+ are then plotted in log-scale to get an idea of spread of these dimensionless wall parameters. Figure (8) shows the plots of u^+ versus y^+ for different groups of cases as 1-5, 8-14, and 16-18. y^+ is used to describe how coarse or fine is the grid near wall for a particular flow pattern. A faster flow near the wall, i.e. for the higher values of u^+ , will produce higher values of y^+ , so the grid size near the wall must be adopted accordingly. These dimensionless parameters are important in turbulence modeling to determine the proper size of the cells near domain walls.

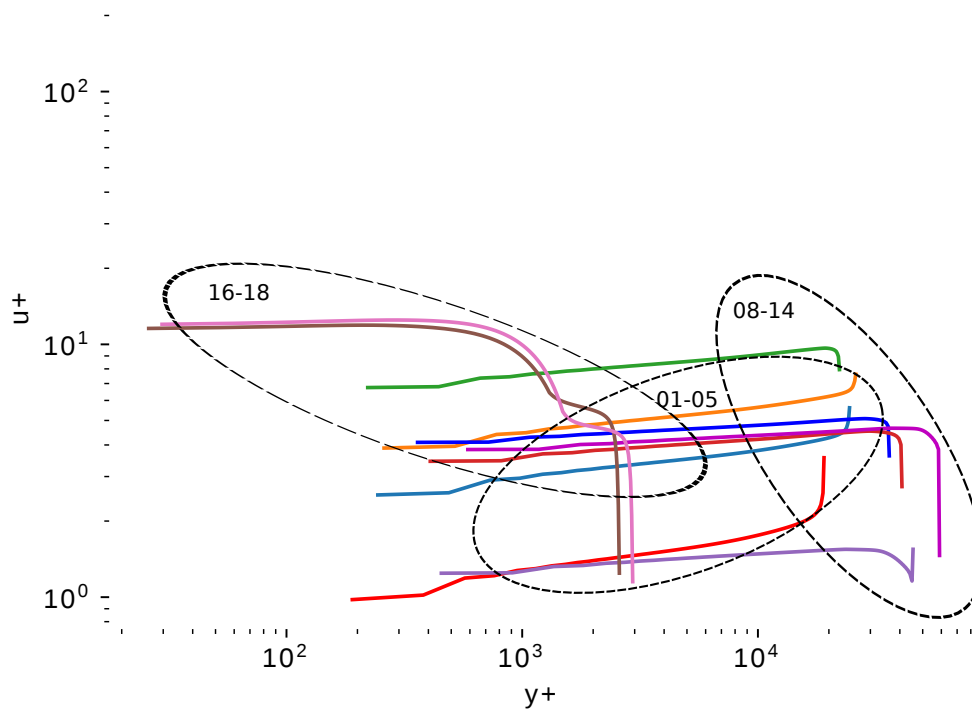


Figure 8. Comparison of dimensionless wall parameters for all the cases.

References

- Gilliot-Ottavy, A. (1997). *Caractérisation par anémométrie à fils chauds d'écoulements turbulents de Poiseuille et de Couette-poiseuille en vue de la validation de modèles de turbulence*. PhD thesis. Thèse de doctorat dirigée par Stanislas, Michel Mécanique Lille 1 1997.
- Telbany, M. M. M. E. and Reynolds, A. J. (1980). Velocity distributions in plane turbulent channel flows. *Journal of Fluid Mechanics*, 100(1):1–29.

4. Tabulated Values

Case	Umax	Umax Error	Uavg	Uavg Error
1	16.3660683	3.5260683	6.27981158	0.14018842
2	15.2103159	2.3703159	7.67768239	0.39768239
3	14.2600693	1.4200693	8.86249551	0.80249551
4	14.2268497	1.3868497	8.9047659	0.7647659
5	13.6106941	0.7706941	9.70389322	0.89389322
6	8.64901449	0.14901449	7.36891593	6.65891593
7	23.9639357	6.8839357	22.9314891	8.3814891
8	17.0545988	4.2145988	16.1613652	4.7813652
9	10.4443017	1.8543017	9.69784031	1.99784031
10	16.6854202	3.4354202	15.595926	3.195926
11	21.2802359	4.9502359	20.2013803	5.1013803
12	33.2949091	11.7249091	31.8879591	11.7779591
13	40.9164111	16.9064111	39.1886138	16.7886138
14	40.9313909	17.3113909	39.2285821	17.3285821
15	24.9619107	8.9619107	23.6211923	9.0711923
16	2.00743461	0.89256539	1.6601757	0.8398243
17	3.21470946	0.11470946	2.09924281	0.40075719
18	3.93965317	0.23965317	2.36001729	0.18998271

Table 1. Table of Computed Umax and Uavg values with their respective errors.

Case	$U_{\tau 1}$	$U_{\tau 1}$ Error	$U_{\tau 2}$	$U_{\tau 2}$ Error
1	0.280823379	0.001176621	0.280823379	0.001176621
2	0.326099612	0.001900388	0.230245425	0.002754575
3	0.360328113	0.001671887	0.178214347	0.002685653
4	0.361493267	0.004493267	0.176117528	0.009217528
5	0.382879677	0.000120323	0.131289166	0.000789166
6	0.305057567	0.007942433	0.0363301083	0.0251698917
7	0.531205808	0.068794192	0.275986253	0.235986253
8	0.437528365	0.047471635	0.207826691	0.184926691
9	0.326468296	0.023531704	0.125819144	0.117419144
10	0.515375051	0.048624949	0.230583066	0.200583066
11	0.601332148	0.077667852	0.365239162	0.179239162
12	0.775435905	0.104564095	0.585234282	0.171034282
13	0.867999255	0.110000745	0.683796232	0.165796232
14	0.885612716	0.075387284	0.767131057	0.097131057
15	0.670085202	0.011085202	0.646703827	0.012296173
16	0.041818926	0.108181074	0.11247745	0.02247745
17	0.0384256412	0.1015743588	0.128543652	0.088543652
18	0.0437743443	0.1062256557	0.133730352	0.083730352

Table 2. Table of Computed U_{τ} values and errors.