COBEM 2021 26th International Congress of Mechanical Engineering
ABCM Associação Brasileira de Engenharia e Ciências Mecânicas
November 22-26, 2021, Virtual Congress, Brazil

COB-2021-0845
A NEW DIRECT NUMERICAL SIMULATION CODE OF BOUNDARY LAYER FLOWS VERIFIED BY THE METHOD OF MANUFACTURED SOLUTIONS

Mateus Paranaíba Ribeiro
Lívia Souza Freire
Leandro Franco de Souza
Universidade de São Paulo, Instituto de Ciências Matemáticas e de Computação, Departamento de Matemática Aplicada e Estatística,
Av. Trabalhador são-carlense, 400, 13566-590, São Carlos, SP, Brasil
mateusparanaiba@gmail.com, liviagrion@gmail.com, lefraso@icmc.usp.br

Abstract. *The objective of this work is to develop an efficient Direct Numerical Simulation (DNS) code with great potential for simulating turbulence in three-dimensional and incompressible boundary layer flows. A velocity-vorticity formulation was chosen to solve the Navier-Stokes equations. In addition, periodic conditions were adopted in the longitudinal and transversal directions. Therefore, it was possible to use the spectral method for the calculation of spatial derivatives in these directions. In the vertical direction, a stretched grid method was used to obtain better precision in the region close to the wall, and a high-order compact finite difference method was used to calculate the spatial derivatives. The time derivative is integrated by a fourth order Runge-Kutta method and the code was verified by the method of manufactured solutions. A three-dimensional DNS code was developed and verified, which has potential to simulate turbulent and incompressible boundary layer flows.*

Keywords: *Boundary layer flow, Direct Numerical Simulation, High-order compact finite-difference scheme, Manufactured Solution Method, Spectral Method.*

1. INTRODUCTION

Turbulent flows have been widely studied due to their presence in many phenomena in nature and industry. Because of the lack of analytical solution of the governing equations, the study of turbulence depends on the development of numerical simulations among its main tools. There are currently several techniques for turbulent flow simulation, including Direct Numerical Simulation (DNS). DNS numerically solves the governing equations and captures all scales of the turbulent flow, which guarantees a high reliability of the results, but at a high computational cost (Lee *et al.*, 2013). It is evident that testing different methodologies and techniques is very important in the development of efficient DNS codes. This includes different formulations of the budget equations, the use of different techniques for approximating spatial and temporal derivatives, the organization of the grid, parallelization, among others.

In this context, a velocity-vorticity formulation was chosen to solve the Navier-Stokes equations, which allowed to eliminate the flow pressure term. Furthermore, periodic conditions were adopted in the longitudinal and transversal directions. Therefore, it was possible to use the spectral method (Gottlieb and Orszag, 1977) for the calculation of spatial derivatives in these directions, which provides exact solutions, reducing numerical errors, as it makes it possible to reduce the calculation of the approximations of the derivatives to a simple product in these directions. In the vertical direction, a stretched grid method was used to obtain better precision in the region close to the wall, since the idea is that, later, this code will be able to effectively simulate the boundary layer, which is a problem that needs high resolution at points close to the bottom wall, due to the large velocities gradient present in this type of problem.

As in Silva (2007), to ensure that a code performs numerical simulations with a good representation of the phenomenon under study, verification and validation tests must be performed (Roache, 1998). In Computational Fluid Dynamics, verification and validation have different meanings and represent different evaluation concepts. On one hand, verification is a study of the numerical code that consists in ensuring that the equations chosen for a given model are being solved correctly and quantify the numerical error of the solution (Steinberg and Roache, 1985). On the other hand, validation involves comparison between numerical solutions and experimental results, with the main objective of evaluating whether the equations used in the model are adequate to model the physical problem of interest. In this work, only code verification was performed. The manufactured solution method (MSM) (Roy *et al.*, 2004) was chosen for this task. This method has been established as one of the most used options for code verification, being also an important tool for quantifying the accuracy of numerical solutions (Roache, 1994), which was performed using MSM combined with a mesh convergence test.

2. FORMULATION AND NUMERICAL METHODS

The flow is assumed to be time-dependent, Newtonian, three-dimensional and incompressible. The governing equations are the continuity and Navier–Stokes equations, given by:

$$\frac{\partial u_i}{\partial t}+\frac{\partial u_i u_j}{\partial x_j} = -\frac{1}{\rho_0}\frac{\partial p}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\right], \qquad i=1,2,3, \tag{1}$$

$$\frac{\partial u_i}{\partial x_i} = 0\,, \tag{2}$$

where the velocity field $\boldsymbol{u}=(u_1,u_2,u_3)$, written in Cartesian coordinates $\boldsymbol{x}=(x_1,x_2,x_3)$, together with the static pressure p and time t, are the variables of the problem, and ρ_0 and ν are density and kinematic viscosity of the fluid (considered constant), respectively. We adimensionalize the variables by:

$$\boldsymbol{x}^*=\frac{\boldsymbol{x}}{L},\ \boldsymbol{u}^*=\frac{\boldsymbol{u}}{U},\ t^*=\frac{U}{L}t,\ p^*=\frac{p}{\rho_0 U^2}, \tag{3}$$

where L and U denote characteristic length and velocity scales, respectively. The symbol * will be omitted for convenience.

A vorticity-velocity formulation was adopted to solve the pressure-velocity coupling problem of Navier-Stokes equations. The flow vorticity is defined as the rotational of the velocity vector, i.e.,

$$\boldsymbol{\omega} = -\nabla\times\boldsymbol{u}\,. \tag{4}$$

After the adimensionlessization of Eq.(1), some manipulations, and applying the vorticity definition, we get

$$\frac{\partial \omega_i}{\partial t}+\varepsilon_{ijk}\frac{\partial a_k}{\partial x_j} = \frac{1}{Re}\nabla^2\omega_i\,, \qquad i,j,k=1,2,3, \tag{5}$$

where $a_i=-\varepsilon_{ijk}u_j\omega_k$ denotes the nonlinear terms, $Re=UL/\nu$ is the Reynolds number and ε_{ijk} is the permutation tensor. The vorticity-velocity formulation corresponds to the numerical solution of the vorticity transport equation (Eq. (5)), together with the Poisson's equations

$$\frac{\partial^2 u_i}{\partial x_j\partial x_j} = \varepsilon_{ijk}\frac{\partial \omega_k}{\partial x_j}\,, \qquad i,j,k=1,2,3, \tag{6}$$

resulting from the combination of the definition of vorticity with the continuity equation. The main advantage of this formulation is the elimination of the pressure term.

Periodicity conditions were adopted in the longitudinal and transversal directions, and therefore it was possible to work with the variables in Fourier space. For this, the discrete Fourier transform (DFT) is defined for a generic variable g in physical space, as

$$g(\boldsymbol{x},t)=\sum_{k_1=0}^{K_1}\sum_{k_3=0}^{K_3}G(k_1,x_2,k_3,t)\exp\Big(-i\beta_1-i\beta_3\Big), \tag{7}$$

where $i=\sqrt{-1}$, β_1 and β_3 are the wave numbers in the respective direction, defined by $\beta_1=2\pi k_1/\lambda_1$ and $\beta_3=2\pi k_3/\lambda_3$ in which λ_1, λ_3 are the fundamental Fourier mode wavelengths and k_1, k_3 are the Fourier modes. Note that k_1 and k_3 vary up to a finite number K_1 and K_3, which are the highest Fourier modes considered in each direction, the others are discarded, making $g(\boldsymbol{x},t)$ an approximation. The variables in Fourier space are denoted generically by capital letters.

Applying this concept to the vorticity transport equations (Eqs.(5)) and Poisson equations for velocity (Eqs.(6)) generates the equations that must be solved for each point k_1, x_2 e k_3:

$$\begin{aligned}
\frac{\partial \Omega_1}{\partial t} + \frac{\partial A_3}{\partial x_2} + i\beta_3 A_2 &= \frac{1}{Re}\nabla^2_{k_1,k_3}\Omega_1, && (8)\\
\frac{\partial \Omega_2}{\partial t} + i\beta_1 A_3 - i\beta_3 A_1 &= \frac{1}{Re}\nabla^2_{k_1,k_3}\Omega_2, && (9)\\
\frac{\partial \Omega_3}{\partial t} - \frac{\partial A_1}{\partial x_2} - i\beta_1 A_2 &= \frac{1}{Re}\nabla^2_{k_1,k_3}\Omega_3, && (10)\\
-\Big(\beta_1^2 + \beta_3^2\Big)U_1 &= i\beta_3\Omega_2 + i\beta_1\frac{\partial U_2}{\partial x_2}, && (11)\\
-\Big(\beta_1^2 + \beta_3^2\Big)U_2 + \frac{\partial^2 U_2}{\partial x_2^2} &= i\beta_1\Omega_3 - i\beta_3\Omega_1, && (12)\\
-\Big(\beta_1^2 + \beta_3^2\Big)U_3 &= - i\beta_1\Omega_2 + i\beta_3\frac{\partial U_2}{\partial x_2}, && (13)
\end{aligned}$$

where

$$\nabla^2_{k_1,k_3} = \left(-\beta_1^2 + \frac{\partial^2}{\partial x_2^2} - \beta_3^2 \right). \qquad (14)$$

In the normal direction, at the bottom wall, there are the no-slip and impermeability conditions. At the upper wall the flow is free. To numerically solve Eqs. (8)-(13), the spatial derivatives were approximated by a high-order compact finite difference scheme for stretching grids in normal direction as shown by Rogenski (2015) and by a spectral method (Gottlieb and Orszag, 1977) in the longitudinal and transversal directions. The time derivative was approximated by the classical fourth order Runge-Kutta method. The chosen method to verify this code was the manufactured solution method (MSM) (Roy *et al.*, 2004) which has already been used in several other works such as in Salari and Knupp (2000) and Petri *et al.* (2015). All the simulations were done in the domain $[0, 2\pi) \times [0, 1] \times [0, 2\pi)$, and the stretching factor sf was defined by

$$h_i = h_{2_0} sf^{(i)}, \qquad i = 0, \cdots, N_2 - 2 \qquad (15)$$

where h_{2_0} and N_2 are the first spacing and the number of points used in the normal direction, respectively. Note that in the case of equidistant points $sf = 1$.

3. MANUFACTURED SOLUTION METHOD

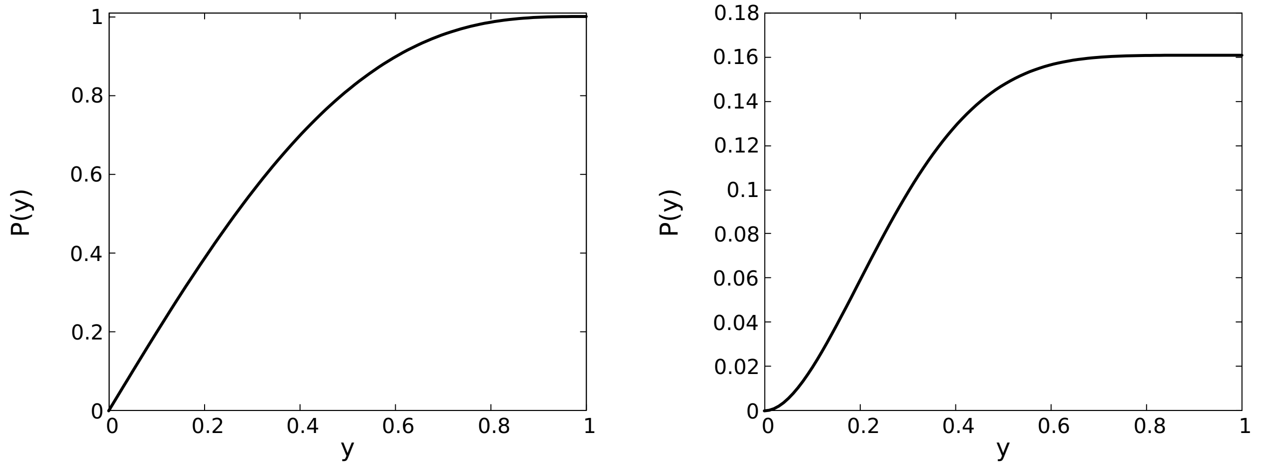
The code was verified by the MSM, which consists of choosing a solution to the problem and calculating what would be the boundary conditions and forcing terms necessary to reach it. For the present work, the following manufactured solutions were chosen:

$$\begin{aligned}
u_1 &= -\sin(x_1)\sin(x_3)P'(x_2),\\
u_2 &= \cos(x_1)\cos(x_3)P(x_2),\\
u_3 &= -\cos(x_1)P'(x_2)[\cos(x_3) + \sin(x_3)],\\
\omega_1 &= \cos(x_1)\{\sin(x_3) + \cos(x_3)P''(x_2)[P''(x_2) - P(x_2)]\},\\
\omega_2 &= \sin(x_1)P'(x_2)[2\cos(x_3) + \sin(x_3)],\\
\omega_3 &= \sin(x_1)[\cos(x_3)P(x_2) - \sin(x_3)P''(x_2)],
\end{aligned}$$

with the polynomial $P(x_2)$ given by

$$P(x_2) = -\frac{7}{8}x_2^8 + \frac{30}{7}x_2^7 - \frac{15}{2}x_2^6 + 4x_2^5 + \frac{15}{4}x_2^4 - 6x_2^3 + \frac{5}{2}x_2^2. \qquad (16)$$

The polynomial P, illustrated in Fig. 1(b), was inspired by the Pohlhausen polynomial (Fig 1(a)), which was made to approximate Blasius' solution. This adaptation was chosen because it has a value of 0 on the lower boundary, satisfying the no-slip and impermeability conditions. It also has a null derivative in both boundaries, which guarantees continuity. Furthermore, it is an 8^{th} order polynomial, preventing the high-order numerical methods used to omit truncation errors. The sine and cosine functions were chosen in the longitudinal and transversal directions as they are periodic. It is noteworthy that with the choice of these manufactured solutions, for the Navier-Stokes equations to be satisfied, it is necessary to add source terms in Eq. (5). These source terms can be found replacing the solution manufactured in Eq. (5) itself.



(a) Original Pohlhausen polynomial.

(b) Adaptation of the Pohlhausen polynomial.

Figure 1: Comparison between the Pohlhausen polynomial and its adaptation used in the code verification.

With the system of equations well defined and a way to exactly calculate the error made in the simulations, it is possible to define the average (E_{mean}) and maximum (E_{max}) errors as:

$$E_{mean} = \frac{1}{N}\sum_{n=1}^{N} |sm_n - sn_n|, \tag{17}$$

$$E_{max} = \max |sm_n - sn_n|, \tag{18}$$

where N is the total number of points compared in the mesh in question, sm_n and sn_n are, respectively, the manufactured and numerical solutions evaluated at the point n of the domain.

The choice of computational meshes used in the mesh convergence test must be done with caution. In order to perform a rigorous test, it is essential to calculate the errors at the same domain points in different meshes. That is a trivial task for evenly spaced meshes, however, in the case of stretched meshes it is necessary that the stretch factor is changed according to the number of points used in this direction. Following what was done by Petri *et al.* (2015), the meshes adopted are in the Tab. 1, in which N_2 represents the number of points used in the wall normal direction, h_2 is the spacing when there is no stretching and h_{2_0} the length of the first spacing in the stretch case. The number of points used in the longitudinal and transversal directions was, in all tests, $N_1 = N_3 = 32$, because in these directions the derivatives are calculated by the spectral method and therefore it is expected that the 32 points (or modes, in the Fourier space) are enough so that the error arising from the calculation of the derivatives in these directions is of the order of the rounding error. To validate this hypothesis, an intermediate stretched mesh was chosen ($N_2 = 65$ and $sf = 1.0201$), and the mean and maximum errors obtained in simulations with $N_1 = N_3 = 32$ and $N_1 = N_3 = 64$ were compared. Ideally the points for comparison should coincide, but in this case it is not possible, because in the periodic directions there is a need to use the DFT, and for this to work correctly, the number of points used in these directions must be of the form 2^k with $k \in \mathbb{N}$. Thus, the points chosen for comparison do not match exactly, but are close enough to obtain the expected result (differences of the order of 10^{-13}), which can be seen in Tab. 2. In both preliminary tests and the simulations for the mesh convergence tests, the manufactured solutions were placed as an initial condition, the total simulation time was 0.04, the Reynolds number adopted was $Re = 1$ and for the Courant-Friedrichs-Lewy condition to be satisfied, the step in time was defined by $\Delta t = 1 \times 10^{-3} h_2$.

Table 1: Meshes used in the verification of the code.

Mesh	N_2	h_2	sf	h_{2_0}
1	33	3.125×10^{-2}	1.0406	1.5777×10^{-2}
2	65	1.5625×10^{-2}	1.0201	7.8093×10^{-3}
3	129	7.8125×10^{-3}	1.01	3.8852×10^{-3}

4. RESULTS

To complete the code verification, one should make sure that the error decay behaves according to the order of the implemented numerical methods. In this case, a compact finite difference method of 6^{th} order in the interior points and

Table 2: Errors in meshes with 65 points in the vertical direction, 32 and 64 points in the longitudinal and transversal directions. The symbol * represents the cases with mesh stretch in the direction normal to the wall.

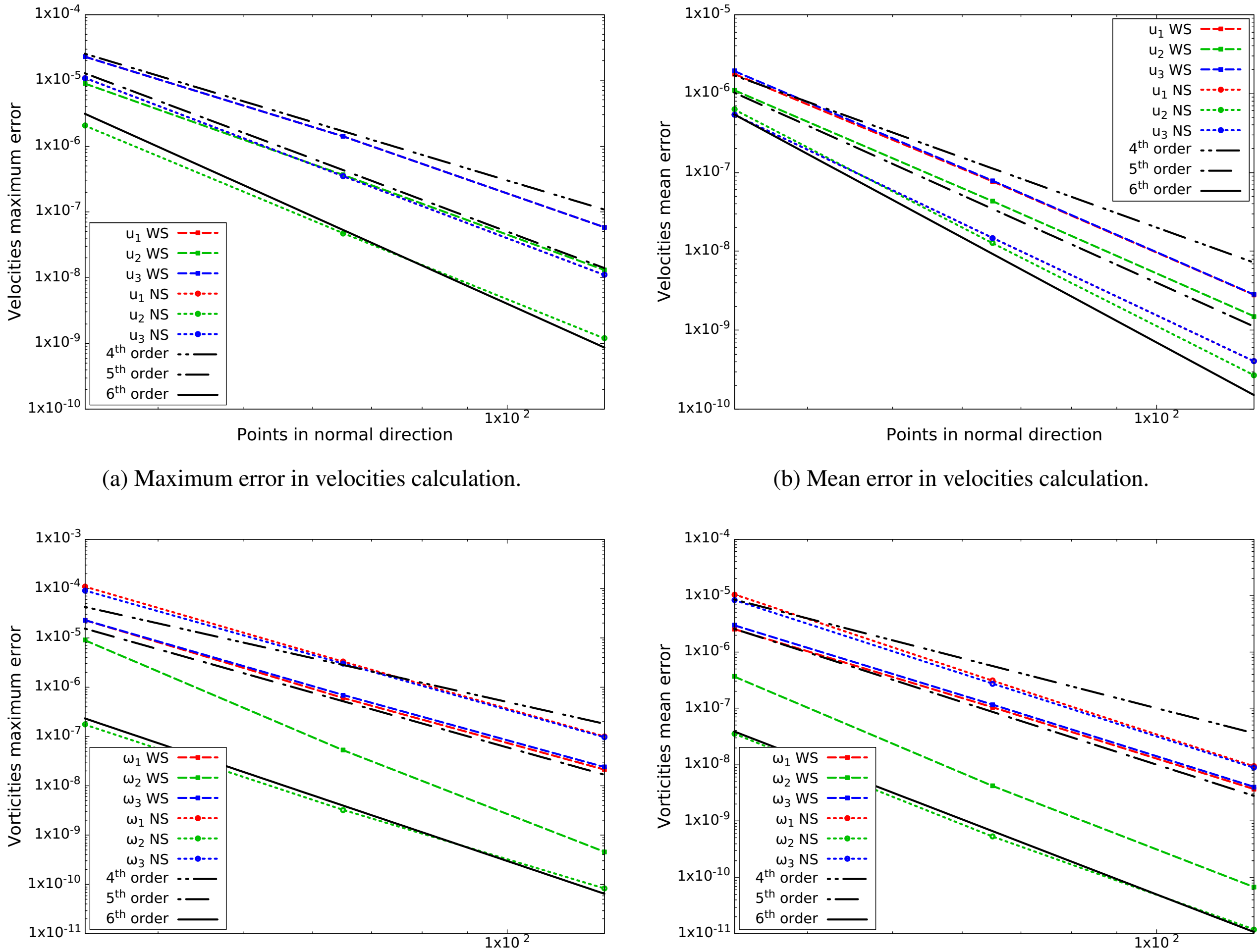
Variable	$N_1 = N_3$	E^*_{max}	E^*_{mean}
u_1	32	$1.40531373 \times 10^{-6}$	$7.67758102 \times 10^{-8}$
	64	$1.40531372 \times 10^{-6}$	$7.67758095 \times 10^{-8}$
u_2	32	$3.63257937 \times 10^{-7}$	$4.30745570 \times 10^{-8}$
	64	$3.63257934 \times 10^{-7}$	$4.30745574 \times 10^{-8}$
u_3	32	$1.40531422 \times 10^{-6}$	$7.83703744 \times 10^{-8}$
	64	$1.40531422 \times 10^{-6}$	$7.83703737 \times 10^{-8}$
ω_1	32	$6.02272131 \times 10^{-7}$	$1.04079820 \times 10^{-7}$
	64	$6.02272011 \times 10^{-7}$	$1.04079818 \times 10^{-7}$
ω_2	32	$5.26062571 \times 10^{-8}$	$4.18623130 \times 10^{-9}$
	64	$5.26062571 \times 10^{-8}$	$4.18623129 \times 10^{-9}$
ω_3	32	$6.92467812 \times 10^{-7}$	$1.16029110 \times 10^{-7}$
	64	$6.92467183 \times 10^{-7}$	$1.16029108 \times 10^{-7}$

of 5^{th} order in the extremes was used. This order reduction to the outermost points is a common practice, it's purpose is to maintain the stability of the method. When using methods of different orders in the same scheme, the expected order of convergence is that of the lowest order method. Knowing this, the results obtained were analysed using the Fig. 2. Note that on one hand in uniform meshes the velocities present superior results to those of stretched meshes. On the other hand, for the vorticities, in the longitudinal and transversal directions, the best results are found in the stretched meshes. Another interesting point is how much bigger the errors in these two directions are in relation to the error in the normal direction. This happens in all cases due to the need to calculate one more derivative by the finite difference method in the vorticity transport equation in these directions. Finally, in all cases the error had an adequate behavior, presenting a slope close to the expected.

The goal is to transform this code into an efficient one to solve boundary layer problems. It is known that higher-order methods tend to have a higher computational cost, but it also provides a better solution with less points in the domain. With this in mind, the magnitudes of the errors in each simulation were evaluated and are available in Tab. 3. With errors in the order of 10^{-5} in the thicker meshes, it is evident that the increase in computational cost when using a larger stencil, and having to solve a tridiagonal system for each derivative calculation, is justified for achieving this type of result with so few points in the domain.

Table 3: Magnitude of errors with the implicit method of 5^{th} order. The symbol * represents the cases with mesh stretch in the direction normal to the wall.

Variable	Mesh	E_{max}	E^*_{max}	E_{mean}	E^*_{mean}
u_1	1	1.0867×10^{-5}	2.2784×10^{-5}	5.3605×10^{-7}	1.7627×10^{-6}
	2	3.4871×10^{-7}	1.4053×10^{-6}	1.4773×10^{-8}	7.6776×10^{-8}
	3	1.1001×10^{-8}	5.8416×10^{-8}	4.0641×10^{-10}	2.8251×10^{-9}
u_2	1	2.0420×10^{-6}	8.9054×10^{-6}	6.2572×10^{-7}	1.0964×10^{-6}
	2	4.7442×10^{-8}	3.6326×10^{-7}	1.2667×10^{-8}	4.3075×10^{-8}
	3	1.1930×10^{-9}	1.3098×10^{-8}	2.7049×10^{-10}	1.5109×10^{-8}
u_3	1	1.0867×10^{-5}	2.2785×10^{-5}	5.3709×10^{-7}	1.9181×10^{-6}
	2	3.4872×10^{-7}	1.4053×10^{-6}	1.4805×10^{-8}	7.8370×10^{-8}
	3	1.1001×10^{-8}	5.8416×10^{-8}	4.0734×10^{-10}	2.8419×10^{-9}
ω_1	1	1.0871×10^{-4}	1.4700×10^{-5}	1.0440×10^{-5}	2.5326×10^{-6}
	2	3.2785×10^{-6}	6.0227×10^{-7}	3.0781×10^{-7}	1.0408×10^{-7}
	3	1.0018×10^{-7}	2.1357×10^{-8}	9.3637×10^{-9}	3.7027×10^{-9}
ω_2	1	1.7367×10^{-7}	5.8954×10^{-6}	3.5319×10^{-8}	3.6431×10^{-7}
	2	3.2513×10^{-9}	5.2606×10^{-8}	5.3203×10^{-10}	4.1862×10^{-9}
	3	8.3430×10^{-11}	4.6163×10^{-10}	1.2084×10^{-11}	6.7881×10^{-11}
ω_3	1	9.0856×10^{-5}	1.7867×10^{-5}	8.2728×10^{-6}	2.9507×10^{-6}
	2	2.9958×10^{-6}	6.9246×10^{-7}	2.7328×10^{-7}	1.1603×10^{-7}
	3	9.6522×10^{-8}	2.4060×10^{-8}	8.8432×10^{-9}	4.0584×10^{-9}



(a) Maximum error in velocities calculation.

(b) Mean error in velocities calculation.

(c) Maximum error in vorticities calculation.

(d) Mean error in vorticities calculation.

Figure 2: Error decay using the implicit method of 5^{th} order in logarithmic scale. The abbreviations WS and NS represent, respectively, the cases with and without stretching.

5. CONCLUSION

In this study, a DNS code for turbulent boundary layer simulation was developed and verified. This code is an extension of the original work of Petri *et al.* (2015), in which periodic conditions were assumed in the longitudinal direction (in addition to the periodicity in the transverse direction of the original code). This modification will allow turbulence to develop without the need to create an inflow condition with disturbance or a damping zone near the outflow boundary. Furthermore, in this study a different polynomial function for the MSM was chosen, which is closer to the Blasuis solution for laminar boundary-layer flows.

For all tests presented, the numerical solution did not diverge (indicating a code without errors). Moreover, the final error decreased as the mesh was refined, achieving order of precision close to the expected for the case without stretching (between 4.92 and 6.05) and for the case with stretching (between 4.02 and 6.83). The minimum order of convergence is a bit lower in the stretched meshes, but similar behavior was also observed in Petri *et al.* (2015), which obtained a much larger drop in precision order (between 1.47 and 3.18). This milder drop in the convergence order is probably related to the periodicity condition in the longitudinal direction, which does not limit the applicability of the code, since the physical problem of interest (atmospheric boundary layer) is better represented in this way. Therefore the implemented numerical methods have been verified and code development may move forward.

6. ACKNOWLEDGEMENTS

This study was partially financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - (Process Number 88887.509738/2020-00); L.S.F. was funded by the São Paulo Research Foundation (FAPESP, Brazil), Grants No. 2018/24284-1 and 2019/14371-7; Research carried out using the computational resource of the Center for Mathematical Applied to Industry (CeMEAI) funded by FAPESP (Process Number 2013/07375-0).

7. REFERENCES

Gottlieb, D. and Orszag, S.A., 1977. *Numerical analysis of spectral methods: theory and applications*. SIAM.

Lee, M., Malaya, N. and Moser, R.D., 2013. "Petascale direct numerical simulation of turbulent channel flow on up to 786k cores". In *SC'13: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. IEEE, pp. 1–11.

Petri, L.A., Sartori, P., Rogenski, J.K. and de Souza, L.F., 2015. "Verification and validation of a direct numerical simulation code". *Computer Methods in Applied Mechanics and Engineering*, Vol. 291, pp. 266–279.

Roache, P.J., 1994. "Response: To the comments by drs. w. shyy and m. sindir". *Journal of Fluids Engineering*, Vol. 116, pp. 198–199.

Roache, P.J., 1998. *Verification and validation in computational science and engineering*, Vol. 895. Hermosa Albuquerque, NM.

Rogenski, J.K., 2015. *Influência do gradiente de pressão na transição em escoamentos sobre superfícies côncavas*. Ph.D. thesis, Universidade de São Paulo.

Roy, C.J., Nelson, C., Smith, T. and Ober, C., 2004. "Verification of euler/navier–stokes codes using the method of manufactured solutions". *International Journal for Numerical Methods in Fluids*, Vol. 44, No. 6, pp. 599–620.

Salari, K. and Knupp, P., 2000. "Code verification by the method of manufactured solutions". Technical report, Sandia National Labs., Albuquerque, NM (US); Sandia National Labs ....

Silva, H.G.d., 2007. *Regime não-linear de trens de ondas modulados na direção transversal em um escoamento de Poiseuille plano*. Ph.D. thesis, Universidade de São Paulo.

Steinberg, S. and Roache, P.J., 1985. "Symbolic manipulation and computational fluid dynamics". *Journal of Computational Physics*, Vol. 57, No. 2, pp. 251–284.

8. RESPONSIBILITY NOTICE

The authors are solely responsible for the printed material included in this paper.