An Implicit Adaptive Non-Linear Frequency Domain Method (pNLFD) for Viscous Periodic Steady State Flows on Deformable Grids

Ali Mosahebi* and Siva K. Nadarajah†
McGill University, CFD Lab, Department of Mechanical Engineering, Montreal, Quebec, Canada

In the present study, an implicit and adaptive Nonlinear Frequency Domain method (pNLFD) has been implemented to the Navier-Stokes equations on deformable grids. Although the computational time for periodic flows is drastically reduced by using the NLFD approach over classical time marching schemes, implementing the pNLFD concept leads to an even faster numerical algorithm. Besides that, the need for a large amount of memory, which is the main disadvantage of the NLFD method, is resolved in the present pNLFD approach. Moreover, the concept of dynamic or moving/deformable grid, which is a need in many problems dealing with periodic flows, is extended to the pNLFD method. Finally, in order to accelerate the convergence, the nonlinear LU-SGS technique which is an implicit time marching method, is implemented. In the LU-SGS technique the cells are treated locally, hence its implementation is quite suitable for the pNLFD method, where different cells have different number of modes and therefore has to be treated individually. Results are presented for 2D stationary, oscillating and pitching cylinders and are compared with previous numerical results as well as experimental data.

I. Introduction

Periodic phenomena widely appear in physical fluid flow problems where the flow contains repeating patterns in time. For instance, fluid flow around helicopter rotor blades and wind turbines, and inside jet turbines are some of well-known cases associated with periodic compressible flows. In the incompressible regime, as an example, there is current interest in developing Micro Air Vehicles (MAV) that produce lift by a periodic flapping mechanism. Numerical methods based on Computational Fluid Dynamics (CFD) are widely used for the modeling and simulation of fluid flow problems. In the case of periodic problems, usually one is most interested in the periodic steady state solution when the initial transient behaviors vanish. However, because of the hyperbolic nature of the governing equations, simulation of the unsteady part of the solution is inevitable. Therefore, usual time marching schemes start solving the equations from an initial condition and march in time until the periodic steady state condition is achieved. The time step should be selected in such a way that all of the complicated flow physics involved in the transient part are appropriately captured. For many cases, the time associated with the initial transient portion is much larger than the time period. Therefore, most of the computational resources are spent on resolving the initial transient before reaching the desired periodic steady state.

Due to vast industrial demands for periodic flow solvers, new numerical techniques such as Linearized frequency domain, deterministic stress,¹ and harmonic balance methods have been devel-

---

*Ph.D. Student, AIAA Member
†Assistant Professor, AIAA Member
oped. Among these methods, only the harmonic balance method which was introduced by Hall et al.\textsuperscript{2} and further validated by McMullen et al.\textsuperscript{3-4} considers the nonlinearity of the governing equations and provide accurate results in the case of highly nonlinear problems. Both Hall and McMullen validated their methods against the Euler and Navier-Stokes equations for a number of unsteady periodic problems and their methods successfully accounted for strong nonlinearities in the problems.\textsuperscript{5-7} In the present study, the nonlinear LU-SGS technique together with local time-stepping are employed to accelerate the convergence to a periodic steady state solution.

Although a significant improvement in execution time can be obtained using the NLFD approach, this method has one primary disadvantage. The need for a large number of modes, places a severe restriction on the hardware resources as large amounts of memory are required. Reducing this large memory demand not only solves the memory restriction, but also increases the computational speed. In many physical problems, regions with high level of nonlinearities are restricted to the wake of an object in external flows, the boundary layer after separation, and around shock waves. Far from these regions, a small number of modes may be sufficient to accurately represent the flow field. It would be a waste of memory to allocate the same number of modes to all of the cells in the domain, and instead, the allocation should be done according to the level of local unsteadiness. The primary concept of the pNLFD approach is to augment the Fourier series in each control volume independent of the others based on the level of the spectral energy of the highest mode. An adaptive non-linear frequency domain approach was initially proposed by Maple et al.\textsuperscript{8-10} The authors demonstrated their approach on a quasi-one-dimensional flow and showed a significant improvement in memory usage.\textsuperscript{8} The extension of the adaptive concept to 2D viscous flows was performed by the present authors.\textsuperscript{11} It was shown that using the adaptive Nonlinear Frequency Domain method not only reduces the memory usage but also reduces the computational time proportionally.

Apart from the need to reduce the memory cost of the current NLFD approach, the scheme presently lacks the ability to model deformable grids which is essential for periodic flows with moving surface boundaries. In the present study, the pNLFD concept on deformable grids is developed for 2D viscous flows.

II. Governing Equations and their Discretization

The Navier-Stokes equations in time-dependent integral form for a moving and/or deforming control volume \( \Omega \) with surface element \( dS \) are as follows,

\[
\frac{\partial}{\partial t} \int_{\Omega} w d\Omega + \int_{\partial \Omega} (\vec{F}_c - \vec{F}_v) dS = 0 \tag{1}
\]

where the state vector \( \vec{w} \), inviscid flux vector \( \vec{F}_c \) and viscous flux vector \( \vec{F}_v \) are described respectively by

\[
\vec{w} = \begin{cases} 
\rho \\
\rho u \\
\rho v \\
\rho E 
\end{cases}, \quad \vec{F}_c = \begin{cases} 
\rho V_r \\
\rho u V_r + n_x p \\
\rho v V_r + n_y p \\
\rho H V_r + p V_t 
\end{cases}, \quad \text{and} \quad \vec{F}_v = \begin{cases} 
0 \\
n_x \tau_{xx} + n_y \tau_{xy} \\
n_x \tau_{yx} + n_y \tau_{yy} \\
n_x \theta_x + n_y \theta_y 
\end{cases} \tag{2}
\]
where

\[ \theta_x = u\tau_{xx} + v\tau_{xy} + k\frac{\partial T}{\partial x} \]

\[ \theta_y = u\tau_{yx} + v\tau_{yy} + k\frac{\partial T}{\partial y} \] \hspace{1cm} (3)

and \( \rho, u, v, T, E, \) and \( H \) denote the density, the Cartesian velocity components, the temperature, the total energy per unit mass, and the total or stagnation enthalpy respectively. The pressure is determined by the equation of state. \( n_x \) and \( n_y \) are the components of the outward facing unit normal vector of the surface \( \partial \Omega \). \( V_r \) and \( V_t \) are the contravariant velocity relative to the motion of grid and the contravariant velocity of the face of the control volume and are defined as:

\[ V_r = V - V_t \]

\[ V = n_x u + n_y v \] \hspace{1cm} (4)

\[ V_t = n_x \frac{\partial x}{\partial t} + n_y \frac{\partial y}{\partial t} \]

The viscous stresses may be written as;

\[ \tau_{xx} = 2\mu \frac{\partial u}{\partial x} + \lambda \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \]

\[ \tau_{yy} = 2\mu \frac{\partial v}{\partial y} + \lambda \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \]

\[ \tau_{xy} = \tau_{yx} = \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \] \hspace{1cm} (5)

where \( \mu \) and \( \lambda = -\frac{2}{3}\mu \) are the first and second coefficients of viscosity. These equations can be non-dimensionalized and be written in integral form for a control volume as follows:

\[ \frac{\partial (\Omega \vec{w})}{\partial t} + \sum_{\partial \Omega} (\vec{F}_c)S - \sum_{\partial \Omega} (\vec{F}_d)S - \frac{\sqrt{\gamma M_\infty}}{Re_\infty} \sum_{\partial \Omega} (\vec{F}_v)S = 0. \] \hspace{1cm} (6)

To eliminate odd-even decoupling of the solution and overshoots before and after shock waves, artificial viscosity \( (\vec{F}_d) \) is added to the convective terms. The artificial dissipation scheme used in the present study is a blended first- and third-order flux, first introduced by Jameson et al.\(^{13}\) Besides the conservation of mass, momentum and energy, the so-called Geometric Conservation Law (GCL) must be satisfied in order to avoid errors introduced by deformation of control volumes.\(^{12}\) This equation for each control volume may be written as;

\[ \frac{\partial \Omega}{\partial t} - \sum_{\partial \Omega} (V_t)S = 0. \] \hspace{1cm} (7)

A finite volume scheme is derived by applying the above equations directly to control volumes to give a set of ordinary differential equations of the form

\[ \frac{d(\Omega \vec{w})}{dt} + R(\vec{w}) = 0, \] \hspace{1cm} (8)

where \( \Omega \) is the cell volume, and the residual \( R(\vec{w}) \) is evaluated by summing the fluxes through the cell faces. The same relation for Eq. 7 can be expressed.
A. Spatial Discretization

The fluxes are represented in discrete form for each computational cell using a central second-order discretization. Equation (8) can then be written for each computational cell as;

$$\frac{d(\Omega_{i,j} w_{i,j})}{dt} + R(w)_{i,j} = 0. \quad (9)$$

The residual can be represented as;

$$R(w)_{i,j} = h_{i+\frac{1}{2},j} - h_{i-\frac{1}{2},j} + h_{i,j+\frac{1}{2}} - h_{i,j-\frac{1}{2}}, \quad (10)$$

where $h_{i\pm\frac{1}{2},j} = F_{c_{i\pm\frac{1}{2},j}} S_{i\pm\frac{1}{2},j} - F_{d_{i\pm\frac{1}{2},j}} - F_{v_{i\pm\frac{1}{2},j}} S_{i\pm\frac{1}{2},j}$. The $\pm\frac{1}{2}$ notation indicates that the quantity is calculated at the flux faces. The values of the flow variables are stored as conservative variables at the cell centers and can be regarded as cell averages. Accordingly, the convective flux at the cell face is computed by taking the average of the flux contributions from each cell across the cell face. The spectral radii of the flux Jacobian matrix are rescaled in each direction to better handle the high aspect ratio cells.

B. Temporal Discretization

The derivation of the NLFD method starts with equation (9) and the assumption that $\bar{\omega} = \Omega \omega$ and $R$ can be represented by separate Fourier series:

$$\bar{\omega} = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \hat{\omega}_k e^{i \frac{2\pi k}{T} t}, \quad R = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \hat{R}_k e^{i \frac{2\pi k}{T} t}, \quad (11)$$

where $i = \sqrt{-1}$. In the case of real problems, the Fourier coefficients for negative wave numbers are simply the complex conjugates of the coefficients for the positive wave numbers and can be eliminated in computation. The Fourier representations are then substituted into the semi-discrete form of the governing equations as described in equation (9) to yield,

$$\frac{d}{dt} \left[ \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \hat{\omega}_k e^{i \frac{2\pi k}{T} t} \right] + \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \hat{R}_k e^{i \frac{2\pi k}{T} t} = 0 \quad \text{in } \Omega. \quad (12)$$

As a result,

$$\hat{R}_k \ast = i \frac{2\pi k}{T} \hat{\omega}_k + \hat{R}_k = 0, \quad (13)$$

forms the new unsteady residual in the frequency domain for each wavenumber and must be solved iteratively using a pseudo time marching concept,

$$\frac{d\hat{\omega}_k}{dt \ast} + \hat{R}_k \ast = 0. \quad (14)$$

The solver attempts to find a solution, $\bar{\omega}$, that drives this system of equations to zero for all wavenumbers. The nonlinearity of the unsteady residual and, hence, bulk of our computing effort stems from the spatial operator, $R$. To calculate $R$ in the frequency domain, transformations between the physical and frequency domains are performed using FFT.

The pseudo-spectral approach begins by initializing the modified state vector, $\bar{\omega}(t)$, at all time instances. At the first iteration, $\bar{\omega}(t)$ will take on the value of the initial condition ($\Omega(0)\omega(0)$) and
for subsequent iterations, the values are based on the previous iterations. At each sub-iteration, as it will be presented shortly, the state vector \( w(t) \) can be obtained through the simple relation 

\[
 w(t) = \bar{w}(t) \Omega(t)
\]

At each of these time instances, the steady-state operator \( R(w(t)) \) can be computed by summing the convective, viscous, and artificial dissipation fluxes as mentioned in the previous subsection. FFT is then used to transform the state vector and spatial operator to the frequency domain where \( \hat{w}_k \) and \( \hat{R}_k \) are known for all wavenumbers. The unsteady residual \( \hat{R}^*_k \) can then be calculated by adding \( \hat{R}_k \) to the spectral representation of the temporal derivative, \( i \frac{2\pi k}{T} \hat{w}_k \). The iteration is advanced and \( \hat{w}_k \) is updated. Using an inverse FFT, \( \hat{w}_k \) is then transformed back to the physical space resulting in a new modified state vector \( \bar{w}(t) \) sampled at evenly distributed intervals over the time period. The wall and far-field boundary conditions are imposed within the time domain.

An unsteady residual exists for each wavenumber used in the solution and the pseudo-time derivative acts as a gradient to drive the absolute value of all of these components to zero simultaneously. A nonlinear LU-SGS method, originally developed by Sun et al.\(^{14}\) and was implemented for NLFD method by Cagnone and Nadarajah,\(^{15}\) is extended to the present pNLFD method. In this method, the real and imaginary parts of Eq. 14 are treated separately and set to be zero. From Eq.14 it is observed that,

\[
\frac{\partial \hat{w}_{Re,k}}{\partial \tau} + \hat{R}_{Re,k} - \frac{2\pi k}{T} \hat{w}_{Im,k} = 0,
\]

\[
\frac{\partial \hat{w}_{Im,k}}{\partial \tau} + \hat{R}_{Im,k} + \frac{2\pi k}{T} \hat{w}_{Re,k} = 0.
\]

(15)

These two equations are coupled through the source term \( i \frac{2\pi k}{T} \hat{w}_k \) and can be solved either coupled or staggered. For sake of simplicity, the staggered method is chosen. When the solution converges, there should be no difference between the two mentioned techniques. In this method the source term is updated explicitly through the sub-iterations. Therefore, for backward Euler discretization of the above equations for each mode and for either real or imaginary parts the following general formula may be used.

\[
\frac{\hat{w}_{Re,c}^{n+1} - \hat{w}_{Re,c}^n}{\Delta \tau} + \hat{R}_{Re,c}^{n+1} = -i \frac{2\pi k}{T} \hat{w}_{Re,c}^{n+1}.
\]

(16)

Although the source term is treated explicitly, in the above formula it is presented as a source term at the time instance \( n + 1 \). It is due to the fact that during sub-iterations between time instances \( n \) and \( n + 1 \), this source term is calculated at the previously updated time instance \( (n + 1, s) \). Therefore, mathematically, when the sub-iterations converge such that \( n + 1 = (n + 1, s + 1) = (n + 1, s) \) this source term is actually evaluated at the time instance \( n + 1 \).

Linearizing Eq. 16 results to,

\[
\frac{\hat{w}_{c}^{n+1} - \hat{w}_{c}^n}{\Delta \tau} + \frac{\partial \hat{R}_c}{\partial \hat{w}_c} \left[ \hat{w}_{c}^{n+1} - \hat{w}_{c}^n \right] + \sum_{nb \neq c} \frac{\partial \hat{R}_c}{\partial \hat{w}_{nb}} \left[ \hat{w}_{nb}^{n+1} - \hat{w}_{nb}^n \right] = -\hat{R}_c^{n} - i \frac{2\pi k}{T} \hat{w}_{c}^{n+1}.
\]

(17)

The direct solution of the above equation is computationally expensive due to the contribution of the neighbor cells. Therefore, a Gauss-Seidel approach which allows the transfer of the neighbors’ contribution to the right-hand side and then solves the above equation iteratively is employed. At
Further simplification can be achieved by the following approximation,

\[
\left( \frac{I}{\Delta \tau} + \frac{\partial \hat{R}_c}{\partial \hat{w}_c} \right) \left[ \hat{w}_{c}^{n+1,s+1} - \hat{w}_{c}^{n} \right] = - \sum_{nb \neq c} \frac{\partial \hat{R}_c}{\partial \hat{w}_{nb}} \left[ \hat{w}_{nb}^{n+1,s} - \hat{w}_{nb}^{n} \right] - \hat{R}_c \frac{2\pi k}{T} \hat{w}_{c}^{n+1,s}, \tag{18}
\]

Further simplification can be achieved by the following approximation,

\[
\hat{R}_c(\hat{w}_{c}^{n}, \hat{w}_{nb}^{n}) + \sum_{nb \neq c} \frac{\partial \hat{R}_c}{\partial \hat{w}_{nb}} \left[ \hat{w}_{nb}^{n+1,s} - \hat{w}_{nb}^{n} \right] \approx \hat{R}_c(\hat{w}_{c}^{n}, \hat{w}_{nb}^{n+1,s}) \approx \hat{R}_c(\hat{w}_{c}^{n+1,s}, \hat{w}_{nb}^{n+1,s}) - \frac{\partial \hat{R}_c}{\partial \hat{w}_c} \left[ \hat{w}_{c}^{n+1,s} - \hat{w}_{c}^{n} \right] = \hat{R}_c^{n+1,s} - \frac{\partial \hat{R}_c}{\partial \hat{w}_c} \left[ \hat{w}_{c}^{n+1,s} - \hat{w}_{c}^{n} \right]. \tag{19}
\]

Using the above approximation in Eq. 18 results to the final formula for updating the unknowns at each sub-iteration,

\[
\left( \frac{I}{\Delta \tau} + \frac{\partial \hat{R}_c}{\partial \hat{w}_c} \right) \left[ \hat{w}_{c}^{n+1,s+1} - \hat{w}_{c}^{n+1,s} \right] = - \left( \hat{R}_c^{n+1,s} + \frac{\hat{w}_{c}^{n+1,s} - \hat{w}_{c}^{n}}{\Delta \tau} + \frac{2\pi k}{T} \hat{w}_{c}^{n+1,s} \right). \tag{20}
\]

As it was previously mentioned, the source term \( \frac{2\pi k}{T} \hat{w}_{c}^{n+1,s} \) is updated through the latest updated values hence should be considered implicit rather than explicit. When the solution converges within sub-iterations such that \( \hat{w}_{c}^{n+1,s+1} \approx \hat{w}_{c}^{n+1,s} \) the right-hand side of Eq. 20 would be zero and the original Eq. 16 would be solved.

The left-hand side matrix in the above equation is stored in a factorized LU form for both the real and imaginary parts for all of the modes at each cell. Using a symmetric sweeping pattern, the above system of equations are solved iteratively. Since the steady state solution is only required, there is no need for high level of convergence in sub-iterations and 3 to 10 sub-iterations are sufficient. Moreover, for better computational efficiency, the Jacobian matrix is frozen and updated after a predetermined number of time steps. The Jacobian matrix \( \frac{\partial \hat{R}_c}{\partial \hat{w}_c} \) is obtained numerically by perturbing the state vector and calculating its effect on the residual using FFT and its inverse. Besides that, the Jacobian in the frequency domain may be related to the Jacobian in the time domain through the chain rule;

\[
\frac{\partial \hat{R}_c}{\partial \hat{w}_c} = \sum_{r=1}^{n} \frac{\partial \hat{R}_c}{\partial \hat{w}_r} \frac{\partial \hat{w}_r}{\partial \hat{w}_c} = \sum_{r=1}^{n} \frac{\partial \hat{R}_c}{\partial \hat{w}_r} \frac{\partial w_r}{\partial \hat{w}_c}. \tag{21}
\]

In the above equation, \( r \) denotes the time steps in real space, and \( \frac{\partial \hat{R}}{\partial \hat{w}} \) and \( \frac{\partial w}{\partial \hat{w}} \) are the coefficients forming the basis for the forward and inverse Fourier transforms.\(^{15}\)

### C. \( p \)NLFD Implementation

The primary concept of the \( p \)NLFD approach is to augment the Fourier series in each control volume independent of the others based on the level of the spectral energy of the highest mode. This
allows for a large reduction in the severe memory cost of the NLFD approach as well as allows for very high number of modes to be introduced in regions of the flow where a larger amount of unsteadiness is present.

The procedure to augment the Fourier series for the LU-SGS method is as follows. First, the initial solution is set to the freestream conditions and each control volume is assigned three time steps per period or the fundamental frequency. This represents the minimum number of modes that can be specified. As the solution progresses, one additional mode is added to each control volume if the ratio of the spectral energy of the highest mode is above a reference spectral energy. The reference spectral energy is a user specified threshold to maintain a minimum desired level of spectral energy in the highest mode. The Fourier coefficients of the new mode are initially set to zero and as the solution develops, the residual of this mode will begin to converge. Since it takes several iterations before the solution converges three or four orders of magnitude, the frequency augmentation procedure is only enforced every several hundred iterations. In the adaptation procedure, the ability to decrease the number of modes for the cells which have more modes than what they really need is implemented as well. During the flow development, some of the cells may initially need higher number of modes, while after the flow is developed, the less number of modes would express the solution with the desire level of accuracy. The adaptation procedure should have the capability to decrease the number of modes for these cells if it is needed. This ability result a unique and better mode distribution in the domain independent of the adaptation procedure. At the moment, for decreasing the number of modes, the one mode before the last mode is considered. If the SE of this mode is less than the RSE, the last mode would be dropped. It should be noted that this approach for increasing and decreasing the number of modes may result unwanted oscillations in the mode distribution, which means the number of modes in some few cells increases in an adaptation step, while in the next adaptation step it decreases. To eliminate these oscillations other criteria is considered, which determine these cells and eliminates their mode oscillations.

Apart from the frequency augmentation, another important modification to the code, is the evaluation of the fluxes across flux faces where the adjacent cells do not hold the same number of modes. Since the NLFD approach is a nonlinear technique, the fluxes and boundary conditions are computed and enforced in the time domain. Hence, at each of these flux faces, a special treatment of the flux is required. At each cell, an inverse FFT of the Fourier coefficients, results in the state vector being sampled at evenly distributed intervals over the time period in the time domain. For an instance, if the left cell has three time steps, the solution is available at the $0^\circ$, $120^\circ$, and $240^\circ$ phases, while the right cell might have five time steps per period, hence the solutions are available at the $0^\circ$, $72^\circ$, $144^\circ$, $216^\circ$, and $288^\circ$ phases. Apart from the solution at the $0^\circ$, there is a mismatch when computing the flux across the face. For this example, to compute the right flux of the left cell, the highest mode in the right cell is dropped and an inverse FFT is applied to recover the solution for a three time step case. The phases match and the corresponding fluxes are computed. The re-sampled solution for the right cell is temporary and is discarded once the fluxes are evaluated. A similar strategy is applied for the left flux of the right cell, where an additional mode with the Fourier coefficients set to zero is added to the left cell, followed by an inverse FFT to recover the solution for a five time step case. The fluxes are then computed for the left flux of the right cell. This allows for a simple approach to compute the fluxes across cell faces with non-matching temporal resolutions, however, it renders a non-conservative scheme.
III. Results and discussion

In this section, the results are presented for the laminar vortex shedding behind 2D stationary, vibrating, and pitching cylinders using an implicit adaptive NLFD method. Both rigid and deformable grid movements are considered and the results are compared. The results are presented for Re=100 and M=0.3.

In the case of the stationary cylinder, the shedding frequency should be obtained as part of solution. A gradient approach was developed by McMullen\textsuperscript{16} for the class of the problems where the time period of the unsteadiness is unknown a priori. In this method, the time period is modified in each iteration and converges to the exact one as part of the solution. The derivation of the method is explained in McMullen’s thesis\textsuperscript{16} and will not be repeated here. The final formula for modifying the time period is as follows;

\[ T_{n+1} = T_n - \Delta T \frac{\partial |\hat{R}_k^e|^2}{\partial T}, \quad (22) \]

where

\[ \frac{1}{2} \frac{\partial |\hat{R}_k^e|^2}{\partial T} = \frac{2\pi V}{T^2} k |\hat{w}_k \times \hat{R}_k^e|, \quad (23) \]

where \( \hat{R}_k^e \) is the unsteady residual and \( \hat{w}_k \) is the state vector in the frequency domain as defined previously.

All the numerical tests for the grid study, determination of the temporal accuracy of the NLFD and pNLFD methods, computation of the speed up factor obtained through using the pNLFD method in comparison with the NLFD method, and the speed up factor which is obtained by using the LU-SGS method instead of the Runge-Kutta method are performed for the case of flow over a stationary cylinder.

In order to determine the minimum grid size required to generate a grid independent solution, four grids of successive refinement were employed with the NLFD method. The results are presented in table 1 for six interested coefficients; the Strouhal number (\( St \)), stagnation pressure coefficient (\( C_{p_s} \)), base suction coefficient (\(-C_{p_b}\)), and drag coefficients (total (\( C_{D_t} \)), pressure (\( C_{D_p} \)), and viscous or friction (\( C_{D_f} \)) drag coefficients). To be sure about the independency of the solution from time step, 10 modes are considered in the simulations. The error percentage (I) is the error between the selected grid for the present study, which is highlighted, and the finest grid. This error is less than 1.5 percent for all of the coefficients. The error percentage (II) is the error between the finest grid and the selected experimental values, which are highlighted as well. It is observed that all of the coefficients get closer to the experimental and numerical values as the number of cells increases.

In table 2, the interested coefficients are presented using the NLFD method. For the temporal resolution surveys, the NLFD results are presented from 1 up to 13 Maximum Number of Modes (MNMs). It is observed from these results that as the number of modes increases, the accuracy of the results improves. For instance, if the convergence criteria is set to an accuracy of five significant digits for all of the mentioned coefficients, the NLFD method with 8 modes is required. Using this number of modes results into the minimum spectral energy of \(9.77 \times 10^{-5}\) for the highest mode and in the whole domain.

Table 3 presents the pNLFD results for the same case. As it was mentioned, in the pNLFD method, the RSE is given and ensures that the SE of the highest mode in each cell would be less than this.
Table 1. Grid study $Re = 100$

<table>
<thead>
<tr>
<th>Grid study</th>
<th>$St$</th>
<th>$C_{ps}$</th>
<th>$-C_{pb}$</th>
<th>$C_{Dt}$</th>
<th>$C_{Dp}$</th>
<th>$C_{Df}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>128 x 68</td>
<td>0.18984</td>
<td>1.08323</td>
<td>0.80008</td>
<td>1.44380</td>
<td>1.09164</td>
<td>0.35217</td>
</tr>
<tr>
<td>192 x 96</td>
<td>0.17046</td>
<td>1.08549</td>
<td>0.75530</td>
<td>1.40090</td>
<td>1.05620</td>
<td>0.34469</td>
</tr>
<tr>
<td><strong>256 x 128</strong></td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td>1.05360</td>
<td>0.34448</td>
</tr>
<tr>
<td>384 x 192</td>
<td>0.16455</td>
<td>1.08711</td>
<td>0.74338</td>
<td>1.39000</td>
<td>1.04682</td>
<td>0.34318</td>
</tr>
<tr>
<td>Experimental results</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Williamson$^{17}$</td>
<td>0.16434</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>McMullen$^{16}$</td>
<td>0.159</td>
<td>-0.697</td>
<td>1.323</td>
<td>0.985</td>
<td>0.342</td>
<td></td>
</tr>
<tr>
<td>Henderson$^{18}$</td>
<td>-</td>
<td>-0.73901</td>
<td>1.34500</td>
<td>1.00476</td>
<td>0.34524</td>
<td></td>
</tr>
<tr>
<td>Park et al.$^{19}$</td>
<td>0.165</td>
<td>1.04</td>
<td>0.73</td>
<td>1.33</td>
<td>0.99</td>
<td>0.34</td>
</tr>
<tr>
<td>Error (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(I)</td>
<td>1.51</td>
<td>0.06</td>
<td>1.25</td>
<td>0.58</td>
<td>0.64</td>
<td>0.37</td>
</tr>
<tr>
<td>(II)</td>
<td>0.13</td>
<td>-</td>
<td>0.59</td>
<td>3.35</td>
<td>4.19</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Table 2. Comparison of the NLFD results with 1 up to 13 modes for $Re = 100$

<table>
<thead>
<tr>
<th>MNM</th>
<th>SE</th>
<th>$St$</th>
<th>$C_{ps}$</th>
<th>$-C_{pb}$</th>
<th>$C_{Dt}$</th>
<th>$C_{Dp}$</th>
<th>$C_{Df}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1.54 \times 10^{-2}$</td>
<td>0.17585</td>
<td>1.08666</td>
<td>0.75329</td>
<td>1.39895</td>
<td>1.05418</td>
<td>0.34478</td>
</tr>
<tr>
<td>2</td>
<td>$6.75 \times 10^{-3}$</td>
<td>0.16756</td>
<td>1.08647</td>
<td>0.74355</td>
<td>1.39064</td>
<td>1.04758</td>
<td>0.34306</td>
</tr>
<tr>
<td>3</td>
<td>$2.73 \times 10^{-3}$</td>
<td>0.16642</td>
<td>1.08645</td>
<td>0.74781</td>
<td>1.39380</td>
<td>1.05016</td>
<td>0.34364</td>
</tr>
<tr>
<td>4</td>
<td>$1.60 \times 10^{-3}$</td>
<td>0.16680</td>
<td>1.08648</td>
<td>0.75193</td>
<td>1.39739</td>
<td>1.05305</td>
<td>0.34434</td>
</tr>
<tr>
<td>5</td>
<td>$5.04 \times 10^{-4}$</td>
<td>0.16699</td>
<td>1.08648</td>
<td>0.75259</td>
<td>1.39798</td>
<td>1.05352</td>
<td>0.34446</td>
</tr>
<tr>
<td>6</td>
<td>$2.76 \times 10^{-4}$</td>
<td>0.16703</td>
<td><strong>1.08649</strong></td>
<td>0.75266</td>
<td>1.39805</td>
<td>1.05358</td>
<td>0.34447</td>
</tr>
<tr>
<td>7</td>
<td>$1.35 \times 10^{-4}$</td>
<td>0.16704</td>
<td><strong>1.08649</strong></td>
<td>0.75268</td>
<td><strong>1.39807</strong></td>
<td>1.05359</td>
<td>0.34447</td>
</tr>
<tr>
<td>8</td>
<td>$9.77 \times 10^{-5}$</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td><strong>1.05360</strong></td>
<td><strong>0.34448</strong></td>
</tr>
<tr>
<td>9</td>
<td>$6.46 \times 10^{-5}$</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td><strong>1.05360</strong></td>
<td><strong>0.34448</strong></td>
</tr>
<tr>
<td>10</td>
<td>$4.04 \times 10^{-5}$</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td><strong>1.05360</strong></td>
<td><strong>0.34448</strong></td>
</tr>
<tr>
<td>11</td>
<td>$2.04 \times 10^{-5}$</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td><strong>1.05360</strong></td>
<td><strong>0.34448</strong></td>
</tr>
<tr>
<td>12</td>
<td>$1.09 \times 10^{-5}$</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td><strong>1.05360</strong></td>
<td><strong>0.34448</strong></td>
</tr>
<tr>
<td>13</td>
<td>$6.25 \times 10^{-6}$</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td><strong>1.05360</strong></td>
<td><strong>0.34448</strong></td>
</tr>
</tbody>
</table>
reference value. In order to set the RSE in the \(p\)NLFD method, the average SE value of the last mode of its corresponding NLFD method is used. The first column in table 2 shows the number of modes in the corresponding NLFD method and the third column shows the RSE which is set for the \(p\)NLFD method tests.

Table 3. Comparison of the \(p\)NLFD results with RSEs corresponding to NLFD results with 1 up to 13 modes for \(Re = 100\)

<table>
<thead>
<tr>
<th>ENM</th>
<th>MNM</th>
<th>RSE</th>
<th>(St)</th>
<th>(-C_{pb})</th>
<th>(C_{Dt})</th>
<th>(C_{Dp})</th>
<th>(C_{Df})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>(1.54 \times 10^{-2})</td>
<td>0.17585</td>
<td>1.08666</td>
<td>0.75329</td>
<td>1.39895</td>
<td>1.05148</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>(6.75 \times 10^{-3})</td>
<td>0.16249</td>
<td>1.08548</td>
<td>0.70743</td>
<td>1.35895</td>
<td>1.02147</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>(2.73 \times 10^{-3})</td>
<td>0.16626</td>
<td>1.08643</td>
<td>0.72126</td>
<td>1.37133</td>
<td>1.03196</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>(1.60 \times 10^{-3})</td>
<td>0.16751</td>
<td>1.08650</td>
<td>0.74541</td>
<td>1.39119</td>
<td>1.04798</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>(5.04 \times 10^{-4})</td>
<td>0.16653</td>
<td>1.08646</td>
<td>0.74821</td>
<td>1.39355</td>
<td>1.04996</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>(2.76 \times 10^{-4})</td>
<td>0.16680</td>
<td>1.08647</td>
<td>1.0864</td>
<td>1.39458</td>
<td>1.05151</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>(1.35 \times 10^{-4})</td>
<td>0.16699</td>
<td>1.08648</td>
<td>0.75126</td>
<td>1.39684</td>
<td>1.05206</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>(9.77 \times 10^{-5})</td>
<td>0.16701</td>
<td>1.08648</td>
<td>0.75170</td>
<td>1.39721</td>
<td>1.05290</td>
</tr>
<tr>
<td>9</td>
<td>11</td>
<td>(6.46 \times 10^{-5})</td>
<td>0.16701</td>
<td>1.08648</td>
<td>0.75217</td>
<td>1.39762</td>
<td>1.05323</td>
</tr>
<tr>
<td>10</td>
<td>12</td>
<td>(4.04 \times 10^{-5})</td>
<td>0.16703</td>
<td>1.08648</td>
<td>0.75240</td>
<td>1.39782</td>
<td>1.05339</td>
</tr>
<tr>
<td>11</td>
<td>13</td>
<td>(2.04 \times 10^{-5})</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75272</td>
<td>1.39809</td>
<td>1.05361</td>
</tr>
<tr>
<td>12</td>
<td>14</td>
<td>(1.09 \times 10^{-5})</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75270</td>
<td>1.39808</td>
<td>1.05361</td>
</tr>
<tr>
<td>13</td>
<td>14</td>
<td>(6.25 \times 10^{-6})</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td>1.05359</td>
</tr>
</tbody>
</table>

In table 4 the obtained coefficients are compared with previous numerical as well as experimental studies. The obtained Strouhal number is in good agreement with the experimental values reported by Williamson,\(^{17}\) while McMullen’s results \((St \approx 0.159)\) has a 3.5% difference.\(^{16}\) Other coefficients are compared with the numerical results of Henderson.\(^{18}\) Again the results are in good agreement except for the pressure drag coefficient and as a consequence total drag coefficient. The difference is due to the compressibility effects since the solution at \(M = 0.2\) has a smaller error for the difference of the pressure drag when compared to the experimental values.

Table 4. Comparison of the obtained coefficients with previous experimental/numerical results for \(Re = 100\)

<table>
<thead>
<tr>
<th></th>
<th>(St)</th>
<th>(-C_{pb})</th>
<th>(C_{Dt})</th>
<th>(C_{Dp})</th>
<th>(C_{Df})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Present study</td>
<td>0.16704</td>
<td>0.75268</td>
<td>1.39807</td>
<td>1.05360</td>
<td>0.34448</td>
</tr>
<tr>
<td>previous studies</td>
<td>(0.16434^a)</td>
<td>0.73901(^b)</td>
<td>1.34500(^b)</td>
<td>1.00476(^b)</td>
<td>0.34524(^b)</td>
</tr>
<tr>
<td>difference%</td>
<td>1.64%</td>
<td>1.84%</td>
<td>3.94%</td>
<td>4.86%</td>
<td>0.22%</td>
</tr>
</tbody>
</table>

a: Experimental results of Williamson\(^{20}\) b: Numerical results of Henderson\(^{18}\)

For convergence up to 5 significant digits, the \(p\)NLFD method requires more modes (MNM=14) when compared to the NLFD method (MNM=8). However, it is expected that for the same value of RSE the same MNM and accuracy would be obtained. It can be shown that satisfying a RSE criteria in the \(p\)NLFD method is not sufficient, but the average SEs of all modes should sufficiently converge to their minimum values. In the other word, the average value of SEs at all modes
should be stabilized. Although in the pNLFD, through the definition of the RSE, the SE of the highest mode is set to the average SE of the corresponding NLFD method, which means no cell may exceed this value, the average SEs of the modes will not be equal to the NLFD method. To simplify this expression, there are not enough cells in higher modes to establish the level of SE at those modes. It is illustrated that the extra modes from 9\textsuperscript{th} to 14\textsuperscript{th} do not contribute to the accuracy of the results in the pNLFD method. Therefore, if in this case, the allowable maximum number of modes forced to be 8 (the same as the NLFD method), the same accurate results would be obtained. Actually the low RSE acts to establish the correct average SE levels at all modes to their minimum possible amounts. The final average SEs of all modes for the pNLFD method with MNM=8 and RSE=6.25 \times 10^{-6} is the same as those values in the NLFD method with MNM=8.

Figure 1 presents the convergence of the Strouhal number versus iteration number. As it can be observed from this figure, the Strouhal number obtained from the pNLFD solver converges to the result of the NLFD solver as higher number of modes are incorporated through the adaptation cycles. It clearly shows how the Strouhal number converges from the NLFD results with low number of modes (the solid red line corresponding to the NLFD result with MNM=2) to the NLFD results with high number of modes (the solid blue line corresponding to the NLFD result with MNM=8).

The average value of the residual corresponding to each mode, \( \hat{R}^*_k \), is presented in Figure 2(a) for the pNLFD method with MNM=14 and in Figure 2(b) for the NLFD method with MNM=8 using the Runge-Kutta time marching algorithm. These two cases are compared because they both resulted to the desired coefficients at the same level of accuracy. In the pNLFD method, the adaptation is done at each 500 iterations. After each adaptation, one mode is added to the figure corresponding to the residual of the new mode. After 7000 iterations, when the 14\textsuperscript{th} mode is added, the mode refinement does not continue - part (I) of Figure 2(a) - instead the distribution of the modes is changed for 45 more adaptation steps, part (II) of Figure 2(a). From iteration 29500 up to the end, neither adaptation nor rearrangement of the modes are changed as illustrated in part (III) of Figure 2(a). The convergence criteria is convergence of all coefficients upto 5 significant digits, illustrated by the vertical blue lines. Both methods converged within 30,000 iterations. Finally, it should be mentioned that the jump which is observed after each adaptation in the pNLFD method is due to the fact that for the adapted cells, the Fourier coefficient of the new mode is initially set to zero. This value is modified after some iterations until convergence. The rate of convergence of these new Fourier coefficients are much faster than the total rate of convergence of the problem; therefore, the effects of residual jumps on the total convergence rate can be neglected.

As it was noted, the main advantage of the pNLFD method in comparison with the NLFD method is the reduction in memory usage and computational time. The ratio of the memory which is used in the pNLFD to NLFD approach is of great importance. These needed memories are almost proportional to the Degree of Freedom (DOF) of the problem which here can be defined as DOF = Number of cells × Number of time steps per cell. In the case of the NLFD method, this definition results in a simple relation for DOF because all of the cells have the same number of modes or equal time steps. However, in the case of the pNLFD approach, the distribution of the modes changes after each adaptation, which makes it difficult to determine the DOF in regions (I) and (II) of figure 2(a). However, when the adaptation of the modes have converged (part (III) of figure 2(a)), the distribution of the modes will no longer change and a fixed DOF can be determined.

Table 5 presents the total number of cells that have the same number of modes (M) from first up to 14\textsuperscript{th} mode. Therefore, the DOF for the NLFD and pNLFD approaches can be obtained as follows;
Table 5. Modes distribution (M) among the Nodes (N) for $p$NLFD case with ENM=13 at $Re = 100$

<table>
<thead>
<tr>
<th>M</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>1334</td>
<td>8725</td>
<td>15067</td>
<td>4882</td>
<td>743</td>
<td>263</td>
<td>220</td>
<td>217</td>
<td>263</td>
<td>159</td>
<td>192</td>
<td>141</td>
<td>44</td>
<td></td>
</tr>
</tbody>
</table>

$\text{DOF} = N \times (2M + 1) = 557056$ for NLFD,

$\text{DOF} = \sum_{i=1}^{14} N_i \times (2M_i + 1) = 242812$ for $p$NLFD.

In these equations, $2M + 1$ is the number of time instances, which the solution should be kept. The minimum value is three when dealing with just 1 mode and adding each extra mode increases this number by two. The memory saving factor can be defined as $\alpha = \frac{\text{DOF of } p\text{NLFD}}{\text{DOF of NLFD}} = 0.44$. Therefore, it is observed that the $p$NLFD method with RSE=6.25 $\times$ 10$^{-6}$ or ENM = 13 requires less than half of the memory which is needed by the corresponding NLFD method with MNM=8. Since the computational time almost linearly increases with increasing the number of modes, the same improvement in CPU time is expected (Speed up factor $\approx 2.3$). However, because of the time which is needed by the FFT and flux adjustment procedures, the real obtained speed up factor is 1.6.

A better comparison between the $p$NLFD and NLFD methods, and Runge-Kutta and LU-SGS time marching algorithms are presented in figure 3. In this figure, the residual of the mean flow component versus CPU time is presented for the NLFD method with MNM=8, the $p$NLFD method with EMN=13, and as described previously, the $p$NLFD method with MNM=8 and RSE =6.25 $\times$ 10$^{-6}$ which are solved using Runge-Kutta method and the NLFD method with MNM=8 and the $p$NLFD method with MNM=8 and RSE =6.25 $\times$ 10$^{-6}$ with the LU-SGS algorithm for the time marching. The errors are non-dimensionalized with respect to its maximum value; therefore, the convergence upto 5 significant digits for all of the cases will occur when these residuals go below 10$^{-9}$. According to this figure, it is clear that almost the same speed up factor is obtained for both LU-SGS and Runge-Kutta algorithms when the $p$NLFD method is implemented in comparison with the NLFD method (1.83 and 1.86 respectively). Moreover, for a same case (NLFD or $p$NLFD), the convergence in LU-SGS method is almost 4 times faster than the Runge-Kutta method.

In order to investigate the generality of the solver, the problem is solved for a range of Reynolds numbers form $Re = 60$ to $Re = 180$ with Ryenolds increments of 20. In Fig. 4, the resulted Strouhal number curve is presented. As it can be observed, decreasing the RSE decreases the difference between the $p$NLFD results and Williamson’s proposed correlation ($St = -3.3265/Re + 0.1516 + 1.6 \times 10^{-4} \times Re$). From $Re \approx 160$ onwards, the laminar 3D effects start and 2D modeling is not appropriate for the simulation of the problem. This is the reason for the divergence observed between the experimental values and 2D numerical results.

A. Deformable grid

The Forced vibrating/pitching cylinder is considered for the deformable grid cases. The grid movement is implemented and compared in two ways; rigid grid movement, where the entire grid rigidly
moves or rotates with the cylinder, and the deformable grid movement, where the grid position on the outer circle remains fixed while the displacements of the interior grid points are subject to the location of the cylinder at a particular time step. In the present cases, only small volume changes due to the small cylinder displacements were chosen to demonstrate the approach. For a more comprehensive study of the grid movement and deformation, the present method will be validated in the near future against other test cases involving large amounts of volume changes.

It should be mentioned that at the moment, the developed solver can handle only one dominant frequency in the domain; hence, the selection of the oscillation frequency and the vibration amplitude should be restricted in such a way that only one frequency remains in the domain. Numerous investigations show that to satisfy this criteria the oscillation of the cylinder should be limited to a region which is known as "lock-in". In this region the synchronization, or "lock-in", between the vortex shedding and the cylinder vibration frequencies occurs. When the wake is synchronized, the vortex-shedding frequency diverges from that corresponding to a fixed cylinder (natural or Strouhal frequency) and becomes equal to the frequency of the cylinder oscillation. Outside the "lock-in" region there are always extra frequencies in the domain. If a set of amplitude and frequency outside of the "lock-in" region are employed, then convergence to machine zero is not possible because of the need to resolve other frequencies.

Five cases are selected. For the vibration in the $y$-direction, $\frac{A}{D} = 0.1$ and $\frac{A}{D} = 0.2$ with $F = \frac{f}{St} = 0.16704 = 1$, and $\frac{A}{D} = 0.2$ with $F = 0.95$, for the pitching $\delta \theta = 10$ and $F = 1$, and for the vibration in the $x$-direction, $\frac{A}{D} = 0.2$ and $F = 1$ are selected. Except for the last case, all the other cases converged completely to the same results for both the rigid and deformable grids (the coefficients upto 5 digits remain the same). The results are summarized in table 6.

<table>
<thead>
<tr>
<th>Case</th>
<th>Vibration Amplitude</th>
<th>Vibration Frequency</th>
<th>$C_{ps}$</th>
<th>$-C_{pb}$</th>
<th>$C_{Dt}$</th>
<th>$C_{Dp}$</th>
<th>$C_{Df}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stationary Cylinder</td>
<td>-</td>
<td>0.16704</td>
<td>1.08649</td>
<td>0.75268</td>
<td>1.39807</td>
<td>1.05360</td>
<td>0.34448</td>
</tr>
<tr>
<td>Plunging Cylinder</td>
<td>0.1</td>
<td>0.16704</td>
<td>1.08598</td>
<td>0.89357</td>
<td>1.50993</td>
<td>1.14540</td>
<td>0.36453</td>
</tr>
<tr>
<td>$y$-dir</td>
<td>0.2</td>
<td>0.16704</td>
<td>1.07877</td>
<td>1.00645</td>
<td>1.61035</td>
<td>1.22992</td>
<td>0.38044</td>
</tr>
<tr>
<td></td>
<td>0.2</td>
<td>0.15869</td>
<td>1.08396</td>
<td>0.94758</td>
<td>1.56350</td>
<td>1.19044</td>
<td>0.37305</td>
</tr>
<tr>
<td>Pitching Cylinder</td>
<td>$10^\circ$</td>
<td>0.16704</td>
<td>1.03252</td>
<td>0.80253</td>
<td>1.47069</td>
<td>1.11386</td>
<td>0.35683</td>
</tr>
<tr>
<td>Transversing Cylinder</td>
<td>0.2</td>
<td>0.16704</td>
<td>1.11076</td>
<td>0.40480</td>
<td>1.45861</td>
<td>1.11392</td>
<td>0.34469</td>
</tr>
</tbody>
</table>

Table 6. Comparison of the results of the four mentioned cases for $Re = 100$

In the following figures, the start time ($t^* = 0$) is selected at the instance which the cylinder is in its minimum vibrating or pitching positions. Moreover, the oscillating frequency is non-dimensionalized with the frequency of the natural vortex shedding (Strouhal number) behind a stationary cylinder at $Re = 100$. In figure 6(a) the lift coefficients are presented in one complete cycle for the stationary cylinder. The drag coefficients for the same case is presented in figure 6(b). The influence of the oscillation amplitude on the total lift and drag coefficients are presented in figures 7(a) and 7(b). From this figure, it is clear that increasing the oscillation amplitude results
in increasing the mean drag coefficient as well as the drag and lift amplitudes. In figures 8(a) and 8(b), the influence of the frequency on the lift and drag coefficients are presented. It is clear that decreasing the frequency results in corresponding decrease in the mean and amplitude values of the mentioned coefficients. Finally, in figures 9(a) and 9(b) and 10(a) and 10(b), the effect of pitching and oscillation in the x-direction on the lift and drag coefficients are investigated. Again, these oscillation increases the mean value of the total drag coefficients and its amplitude as well as the total lift coefficient amplitude. It is observed that the average lift coefficients remain zero for a symmetric vibration or pitching, as expected.

Figures 5(a), 5(c), and 5(e) illustrate the grids, which are produced through using the deformable grid method, at three time instances, \( t^* = \frac{t}{T} = 0, \frac{1}{4}, \) and \( \frac{1}{2} \) for a plunging cylinder with \( \frac{A}{D} = 0.2 \). As it was mentioned, at \( t^* = 0 \), the cylinder is in its minimum vertical position \( (y_c = -0.2) \). The same results are presented for the pitching cylinder case at the same time instances in figures 5(b), 5(d), and 5(f).

Figure 11(a) presents the vorticity contours for the flow around the stationary cylinder at \( Re = 100 \). This figure shows how the positive and negative vorticies shed behind the cylinder. In figure 11(b) the mode distribution contour for the same case is presented. It shows that more modes are required for the cells which are at the wake of the cylinder while the rest of the domain needs less number of modes for the same level of accuracy. In figures 12 to 14 the same results are presented for the plunging cylinders (oscillating in y-direction). In figure 12 and 13 the non-dimensionalized frequency is \( F = 1 \) and the oscillation amplitudes are \( \frac{A}{D} = 0.1 \) and \( \frac{A}{D} = 0.2 \) respectively while in figure 14 \( \frac{A}{D} = 0.2 \) and \( F = 0.95 \). These results are presented for the pitching cylinder with \( \delta \theta = 10^\circ \) and \( F = 1 \) in figure 15 and for the x-direction vibrating cylinder at \( F = 1 \) and \( \frac{A}{D} = 0.2 \) in figure 16. Through these figures it is observed that in the case of plunging or vibrating cylinders, the region that requires higher number of modes is extended to the surface of the cylinder (even to the front of the cylinder in the case of vibrating in the x-direction), while in the case of stationary or pitching cylinders it starts in the wake and not on the cylinder wall. Moreover, it is obvious that along the stagnation line up to stagnation point, less number of modes is required, which shows low level of unsteadiness in these regions.

IV. Conclusion

In the present study an Implicit adaptive nonlinear frequency domain technique is developed for the deformable grids and implemented to the Navier-Stokes equations. For validating the \( pNLFD \) solver, the laminar vortex shedding behind 2D stationary, oscillating, and pitching cylinders are modeled. The results show good agreement with experimental investigations as well as other numerical results. Moreover, a significant reduction in memory usage as well as computational time was achieved through the implementation of the adaptive concept and LU-SGS method. The numerical results show that \( pNLFD \) method can improve both memory usage and CPU time up to 2 times in comparison with the corresponding NLFD method for the same convergence criteria. Moreover, using LU-SGS technique can increase the rate of convergence four times in comparison with the Runga-Kutta method. The obtained results for the deformable grid and the rigid grid movements are similar for these test cases but more validations are needed.
V. Acknowledgement

This research was supported by the McGill University Faculty of Engineering MEDA program and Fonds québécois de la recherche sur la nature et les technologies, Projet de recherche en quipe.

References


Figure 1. Convergence of Strouhal number vs. iteration number for the $p$NLFD and NLFD methods at $Re = 100$.

Figure 2. Decreasing of the average residual for each mode vs. iteration number at $Re = 100$: (a) $p$NLFD method with MNM=14 (b) NLFD method with MNM=8.
Figure 3. The average residual of the mean flow component vs. CPU time for the NLFD and $p$NLFD methods solved with the Runge-Kutta and LU-SGS techniques.

Figure 4. Strouhal number vs. Reynolds number for the $p$NLFD method with different values of the RSE.
Figure 5. The grid for deformable grid movement at different time instances; (a), (c), and (e) for plunging cylinder with oscillating amplitude of \( \frac{A}{D} = 0.2 \); (b), (d), and (f) for pitching cylinder with \( \delta \theta = 10^\circ \).
Figure 6. The pressure, friction, and total lift and drag coefficients for a stationary cylinder at \( Re = 100 \); (a) Lift coefficients (b) Drag coefficients.

Figure 7. The effect of oscillation amplitude (y direction) on the total lift and drag coefficients at \( Re = 100 \) and \( F = 1 \); (a) Total lift coefficient (b) Total drag coefficient.
Figure 8. The effect of oscillation frequency on the total lift and drag coefficients of a cylinder oscillating in y direction with oscillation amplitude of $\frac{A}{D} = 0.2$ at $Re = 100$; (a) Total lift coefficient (b) Total drag coefficient.

Figure 9. Total lift and drag coefficients for a pitching cylinder with $\delta \theta = 10^\circ$ at $Re = 100$ and $F = 1$ in comparison with a stationary cylinder; (a) Total lift coefficient (b) Total drag coefficient.
Figure 10. Total lift and drag coefficients for an oscillating cylinder (x-direction) with scillation amplitude of $A/D = 0.2$ at $Re = 100$ and $F = 1$; (a) Total lift coefficient (b) Total drag coefficient.

Figure 11. Flow around a stationary cylinder at $Re = 100$; (a) The vorticity contour (b) The mode distribution contour.
Figure 12. Flow around an oscillating cylinder ($y$-direction) with oscillation amplitude of $\frac{A}{D} = 0.1$ at $Re = 100$ and $F = 1$; (a) The vorticity contour (b) The mode distribution contour.

Figure 13. Flow around an oscillating cylinder ($y$-direction) with oscillation amplitude of $\frac{A}{D} = 0.2$ at $Re = 100$ and $F = 1$; (a) The vorticity contour (b) The mode distribution contour.
Figure 14. Flow around an oscillating cylinder (y-direction) with oscillation amplitude of $\frac{A}{D} = 0.2$ at $Re = 100$ and $F = 0.95$; (a) The vorticity contour (b) The mode distribution contour.

Figure 15. Flow around a pitching cylinder with $\delta \theta = 10$ at $Re = 100$ and $F = 1$; (a) The vorticity contour (b) The mode distribution contour.
Figure 16. Flow around an oscillating cylinder (x-direction) with oscillation amplitude of $\frac{A}{D} = 0.2$ at $Re = 100$ and $F = 1$; (a) The vorticity contour (b) The mode distribution contour.