Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 212 (2006) 637–661
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

The computational efficiency of non-linear frequency domain methods ☆

M.S. McMullen *, A. Jameson

Stanford University, Department of Aeronautics and Astronautics, Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035, United States

Received 14 December 2004; received in revised form 21 June 2005; accepted 15 July 2005
Available online 21 September 2005

Abstract

This paper demonstrates efficient solver technologies applied to the non-linear frequency domain (NLFD) method. The basis of the NLFD method is to assume the time period of the solution's oscillation and to transform both the solution and residual using a discrete Fourier transform. An unsteady residual is formed in the frequency domain and iteratively driven to a negligible value. This method is amenable to many of the convergence acceleration techniques used for steady state flows including pseudo-local time stepping, implicit residual averaging, coarse grid viscosity and multigrid. This paper will address the implementation of these techniques such that convergence rates of the modified unsteady solver are equivalent to those of the original steady-state techniques.
© 2005 Elsevier Inc. All rights reserved.

PACS: 47.11.+j

Keywords: Non-linear; Frequency; Harmonic; Balance; Unsteady; Fluid; Dynamics

1. Introduction

The calculation of unsteady flows continues to present a severe challenge to computational fluid dynamics (CFD). Depending on the requirements imposed by the end user, these unsteady flows can be divided into two categories. The first category contains flows, where the resolution of the complete

☆ Research supported by grants from the Accelerated Strategic Computing Initiative.
* Corresponding author. Tel.: +1 518 3877577.
E-mail addresses: mcmu@stanford.edu, mac_acl_2k3@yahoo.com (M.S. McMullen).
URL: aero-comlab.stanford.edu/mcmu/index.html (M.S. McMullen).

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2005.07.021

Nomenclature

English letters

c	wave speed
C_l	coefficient of lift
C_m	coefficient of moment
e	residual averaging coefficient
E_l	error in lift coefficient magnitude
E_m	error in moment coefficient magnitude
F	flux contained in spatial residual
I	unsteady residual
k	wave number
N	vector normal to surface or number of samples used in discrete Fourier transform
N_M	number of multigrid cycles required for a converged solution
N_N	number of solution instances required to resolve a solution oscillation
N_P	number of periods required to reach a periodic steady state
R	spatial residual
s	integer power of spectral viscosity term or arc length
t	time
U	continuous solution
V	volume
W	discrete solution
x	spatial dimension

Subscripts and superscripts

$\vec{(\,)}$	vector form of variable
$\hat{(\,)}_k$	Fourier coefficient of variable
$\bar{(\,)}$	averaged variable

Greek and roman symbols

ε_N	coefficient that scales course grid spectral viscosity
λ	eigenvalue
Λ	steady-state CFL
λ_t	eigenvalue of temporal discretization

time history of the solution is required by the user. Another category contains flows, where the user requires only the solution once it has reached a periodic steady state. The motivation behind this paper is to demonstrate numerical techniques that reduce the cost of computing this second category involving only periodic flows.

In the worst case, the timescale of the initial transient decay is much larger than the timescale of the periodic solution. For time accurate flow solvers, computing the decay of the initial transients represents the dominant computational cost in comparison with a single oscillation of the solution at its periodic steady state. An example of a physical problem with multiple timescales is the numerical simulation of the flow through an axial flow turbine. Yao et al. [20] computed an unsteady Reynolds averaged Navier–Stokes (RANS) calculation on a $1\frac{1}{2}$ stage turbine modeled after an experimental test rig. A

periodic steady state was reached after roughly 2500 time steps. The fundamental period of the oscillation of the solution at its converged state is resolved in about 80 time steps; a ratio of approximately 3.2%.

Traditionally scientists have attempted to mitigate these costs by employing Fourier series in the solution of the partial differential equations to solve directly for the solution at its periodic steady state. In fluid mechanics, acousticians and other researchers routinely linearize the conservation equations and assume that the unsteady perturbations are periodic in time. The procedure is to first solve for the mean flow components of the solution and then independently solve for any temporal frequency of the solution. The total solution can be reconstructed as the superposition of each frequency component given the linearization of the governing equations. The cost of this solution procedure is proportional to the product of the number of temporal modes calculated and the cost of the steady solution. The scheme does have drawbacks. Applications in turbomachinery and aeroelasticity often require transonic non-linear solutions, making this approach inapplicable due to the assumptions of the linearization.

Efficient periodic solutions to fully non-linear systems of equations were first proposed by Hall et al. [3,4] using the harmonic balance technique. This technique utilizes a pseudo-spectral approach to represent the non-linear residual in the temporal domain. McMullen et al. [14,15] proposed the non-linear frequency domain (NLFD) method which represents a form of the residual in the frequency domain. Regardless of the approach, iterative methods are then employed to drive this residual to a negligible value in a manner that is consistent with steady-state solvers. Both scientists have demonstrated the efficiency of this technique to represent complex non-linear flow solutions using a minimum number of modes. If all the modes of the solution converge as quickly as a similar steady-state calculation, then the cost of the calculation is the product of the cost of a steady solution and the number of instances used in the time series of the unsteady residual. This is consistent with linearized methods except that a slight penalty is incurred by the additional memory required to hold all the unsteady modes simultaneously.

The critical item in the discussion of the method's cost is the assumption that the convergence rate of the pseudo-spectral form of the residual is equivalent to that of the convergence rate of a steady-state code. One of the advantages of the pseudo-spectral approach is that it is amenable to parallel processing and classical convergence acceleration methods such as pseudo-local time stepping, multigrid, residual averaging, and coarse grid viscosity. It is the implementation details of these methods that are the focus of this paper. Ultimately, we will demonstrate a solver which has convergence properties consistent with representative solvers already developed for steady flows.

In later sections of this paper, we rigorously compare the cost of the NLFD solver to the cost of a representative unsteady solver, UFLO82. The comparison is based on operating the two codes (both with equivalent spatial discretizations) at equivalent error levels, and measuring the computational cost associated with each method. This paper will demonstrate the efficiency of the NLFD method over time accurate techniques for computations of a transonic pitching airfoil where the decay of the initial transients is relatively fast. In comparison to the turbomachinery example cited above, this ratio of cost spent computing the final period to that used computing the initial transient decay is between 14% and 33%. Although this quick decay naturally favors time accurate methods, we will demonstrate a clear performance benefit associated with NLFD methods. The relatively efficiency of NLFD methods will only improve for more complex problems like the turbomachinery example cited above.

Finally, we will demonstrate the sensitivity of the convergence rates of this solver to various physical parameters in the model problem. Specifically, we will vary the frequency of the airfoil's oscillation and its magnitude in an attempt to affect the level of non-linearities exhibited in the solution. These tests establish the robustness of the technique and demonstrate that the efficiency gains stated in previous sections were not the result of a model problem that was fortuitously selected.

2. Numerical method

2.1. Transforming the equations into the frequency domain

Detailed discussions of the development of the NLFD method for the Euler and Navier–Stokes equations have already been published [16]. However, the flexibility of the approach will allow its extension to many other similar systems of partial differential equations. As such this section will discuss the method using an arbitrary system of time dependant conservation equations that can be written for a generic control volume, Ω, as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega} U \,\mathrm{d}V + \oint_{\partial\Omega} \vec{F}\cdot\vec{N}\,\mathrm{d}s = 0, \tag{1}$$

where U is the continuous solution variable and $\vec{F}$ represents the transport of the conserved quantity across the control volume boundary. Assume that we can approximately represent the above equation using the following:

$$V\frac{\partial W}{\partial t} + R = 0, \tag{2}$$

where W and R are discrete approximations of the solution and flux integrals, respectively. If W an R are periodic in time then both can be represented by separate Fourier series:

$$\begin{aligned} W &= \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \widehat{W}_k \,\mathrm{e}^{\mathrm{i}kt}, \\ R &= \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \widehat{R}_k \,\mathrm{e}^{\mathrm{i}kt}, \end{aligned} \tag{3}$$

where

$$\mathrm{i} = \sqrt{-1}. \tag{4}$$

These discrete Fourier transforms can be substituted into the semi-discrete form of the governing equations provided by Eq. (2), and the time derivative of the state variable can be moved inside the series summation. Taking advantage of the orthogonality of the Fourier terms results in a separate equation for each wavenumber, k, in the solution

$$\mathrm{i}kV\widehat{W}_k + \widehat{R}_k = 0. \tag{5}$$

Here, however each coefficient $\widehat{R}_k$ of the transform of the residual depends on all the coefficients $\widehat{W}_k$, because $R(W(t))$ is a non-linear function of $W(t)$. Thus (5) represents a non-linear set of equations which must be iteratively solved. The solver attempts to find a solution, W, that drives this system of equations to zero for all wavenumbers, but at any iteration in the solution process the unsteady residual, $\widehat{I}_k$, will be finite:

$$\widehat{I}_k = \mathrm{i}kV\widehat{W}_k + \widehat{R}_k. \tag{6}$$

The nonlinearity of the unsteady residual stems from the spatial operator. There are two approaches to calculating the spatial operator expressed in the frequency domain. The first uses a complex series of convolution sums to calculate $\widehat{R}_k$ directly from $\widehat{W}_k$. Such an approach was discussed in Hall's introductory paper on Harmonic Balance techniques [3]. Hall justly discarded the approach due its massive complexity (con-

sidering artificial dissipation schemes and turbulence modeling) and cost that scales quadratically with the number of modes N.

The alternative proposed by Hall and modified by this research is to use a pseudo-spectral approach that relies on the computational efficiency of the fast Fourier transform (FFT). A diagram detailing the transformations used by the pseudo-spectral approach is provided in Fig. 1. The pseudo-spectral approach begins by assuming that $\widehat{W}_k$ is known for all wavenumbers. Using an inverse FFT, $\widehat{W}_k$ can be transformed back to the physical space resulting in a state vector $W(t)$ sampled at evenly distributed intervals over the time period. At each of these time instances the steady-state operator $R(W(t))$ can be computed. A FFT is then used to transform the spatial operator to the frequency domain, where $\widehat{R}_k$ is known for all wavenumbers. The unsteady residual $\widehat{I}_k$ can then be calculated by adding $\widehat{R}_k$ to the spectral representation of the temporal derivative $ikV\widehat{W}_k$.

The cost of the FFT is proportional to $N\ln(N)$. For most realistic values of N ($N = 1 \rightarrow 10$) the cost of the pseudo-spectral approach is dominated by the cost associated with calculating the spatial operator. Consequently the overall cost of the simulation scales by the product of the cost of evaluating a steady-state spatial operator and the number of time instances used to represent the solution N.

Instead of directly solving Eq. (5) a pseudo-time derivative can be added, and a time-stepping scheme can be employed to numerically integrate the resulting equations

$$V\frac{\partial \widehat{W}_k}{\partial \tau} + \widehat{I}_k = 0. \tag{7}$$

The application of the pseudo-time derivative is consistent with established convergence acceleration techniques used to solve steady-state problems. In the NLFD case, 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.

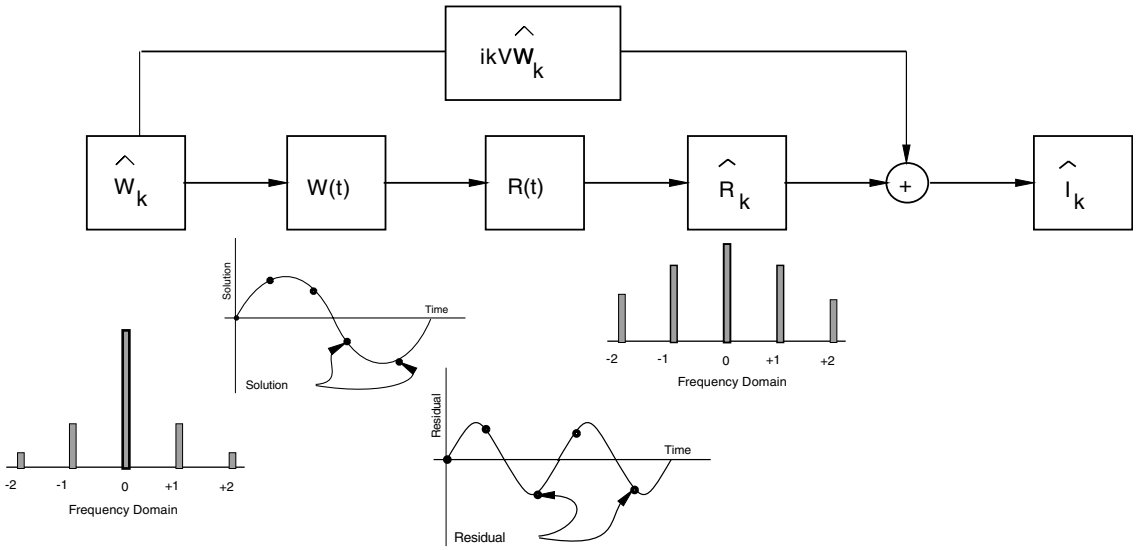

Fig. 1. Simplified dataflow diagram of the time advancement scheme illustrating the pseudo-spectral approach used in calculating the non-linear spatial operator R.

2.2. Local time stepping

A modified Runge–Kutta (RK) time-stepping scheme is used to advance the solution forward in pseudo-time [13]. Separate coefficients are used for the dissipative and convective components of the spatial operator at each stage of the RK scheme. These are based on a set of coefficients optimized to accelerate convergence for steady-state flows. Since the pseudo-time equation is being used to solve Eq. (6) in the frequency domain, the transformations between the frequency and the time domain illustrated in Fig. 1 are applied at every stage of the RK scheme.

For each wavenumber, a local time step dictated by numerical stability is used in each cell. Unlike an explicit time-accurate scheme where the global time step is determined by the minimum of the time steps for all the cells, the local time stepping scheme maximizes the correction each cell can make per time step. The pseudo-time evolution does not affect the temporal accuracy of the NLFD formulation, which is only a function of the number of modes used to represent the solution and residual.

Previous methods to solve unsteady flow problems have employed both explicit [10] and implicit [17] approaches in the treatment of the diagonal terms associated with the temporal derivatives. The explicit approach bases the calculation of the temporal derivative on known values of the solution, while the implicit approach uses a future value of the solution in this calculation. Following the work of Melson et al. [17] a stability analysis of both schemes has been performed and documented by McMullen [16]. Due to the imaginary character of the spectral representation of the temporal derivative, the implicit approach is flawed from a stability perspective. The NLFD code has been implemented with a explicit scheme where the temporal derivative is completely reevaluated at each step in the scheme. However, the set of coefficients associated with the viscous terms lag the evaluation of the spatial operator. This complicates the dataflow diagram for the pseudo-spectral approach shown in Fig. 1. A modified dataflow diagram including these lagged operations is included in Fig. 2. This diagram shows that the spatial operator, R, calculated in physical space is a function of the spatial operator at the previous stage and the spatial operator calculated using the current stage of the solution. The current iterate of the spatial operator is transformed back into the frequency domain using a FFT where it is added to frequency domain representations of the temporal derivative and the initial solution to produce a new stage of the solution.

2.3. Residual averaging

An implicit smoothing operator is applied to the unsteady residual in a process known as residual averaging. The coefficients of this implicit smoothing operator are chosen such that larger time steps can be used while maintaining the overall stability of the numerical scheme. The additional cost of applying the operator is offset by the improved convergence rates per multigrid cycle.

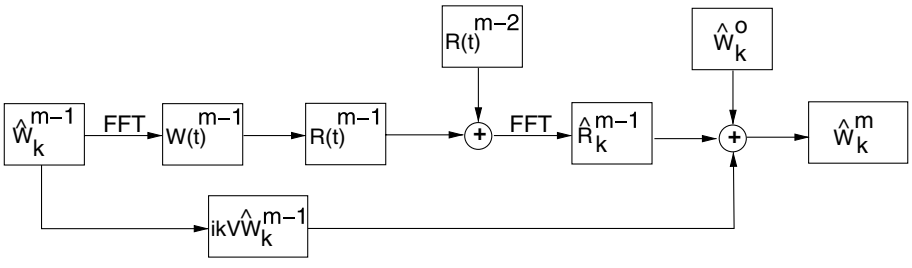

Fig. 2. Complete dataflow diagram of the time-stepping method.

This section proposes a residual averaging method which averages only in the spatial directions. Inspection of the unsteady residual presented in Eq. (7) shows that the temporal derivative term is proportional to the product of the temporal frequency and cell volume. For low frequency cases on a well refined grid these terms are second order and their impact on the stability of the numerical method can be neglected. However, the multigrid process transfers the solution to a very coarse mesh where these terms may dominate. If one does not adjust the time step accordingly, the instabilities created on the coarse mesh will cause instabilities on the fine mesh ultimately corrupting the solution.

The remainder of this section will use the advection equation as a model to derive stability estimates for the residual averaging operators. In space and time a continuous form of implicit residual averaging can be written as follows:

$$\left(1 - e\frac{\partial^2}{\partial x^2}\right)\Delta\tau\widehat{I} = \Delta\tau\left(\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x}\right). \tag{8}$$

For consistency with the NLFD method, assume that both the solution and residual can be represented by a Fourier series in time. Applying central difference operators to the spatial derivatives results in a discrete approximation to the continuous equation

$$-e\Delta\tau_k\widehat{\overline{I}}_{k_{i+1}} + (1+2e)\Delta\tau_k\widehat{\overline{I}}_k - e\Delta\tau_k\widehat{\overline{I}}_{k_{i-1}} = -\Delta\tau_k\left(\mathrm{i}k\hat{u}_k + c\frac{\hat{u}_{k_{i+1}} - \hat{u}_{k_{i-1}}}{2\Delta x}\right). \tag{9}$$

A von Neumann analysis of the resulting equation provides the spectral footprint of the time advancement scheme. In semi-discrete form the averaged residual can be written as the product of an eigenvalue λ and the solution

$$\begin{aligned} &\Delta\tau_k\widehat{\overline{I}}_k = \lambda\mathrm{i}\hat{u}_k, \\ &|\lambda| = \frac{k\Delta\tau_k + \Lambda\sin(\rho)}{1 + 2e(1-\cos(\rho))}, \\ &\Lambda = \frac{c\Delta\tau_k}{\Delta x}, \\ &\rho = \omega\Delta x. \end{aligned} \tag{10}$$

A thorough analysis of the effects of residual averaging on the stability of the NLFD method is provided by McMullen et al. [16]. Without the extensive algebra, a result can be obtained using asymptotic methods. If the temporal derivative is assumed large in comparison to the spatial residual then Eq. (10) is simplified to

$$|\lambda| = \frac{k\Delta\tau_k}{1 + 2e(1-\cos(\rho))}. \tag{11}$$

By inspection, the frequency ρ that maximizes the length of the residual is zero. Substituting this result back into the original equation results in an expression for the maximum permissible time step, which in essence is a CFL limiter that rescales the length of the eigenvalue to ensure stability

$$\Delta\tau_k < \frac{|\lambda|}{k} \quad \text{for } k > 0. \tag{12}$$

2.4. Coarse grid spectral viscosity

During the original tests of the multigrid solver, it was noticed that the convergence rate of the solver exhibited a dependence on the frequency of the unsteady solution; as this frequency increased the

convergence rate of the solver decreased. Noting Eq. (6), the effect of frequency on the temporal derivative is obvious. It can be argued that if the convergence rate of the steady problem ($k = 0$) is considered optimal, then any increase in frequency moves the set of equations away from the steady-state solution and negatively impacts convergence. In order to mitigate these effects, additional terms have been added to the coarse grid residual:

$$V\frac{\mathrm{d}\widehat{W}_k}{\mathrm{d}\tau} + \mathrm{i}kV\widehat{W}_k + \widehat{R}_k = \varepsilon_N V(\mathrm{i}k)^{2s}\widehat{W}_k. \tag{13}$$

Terms of this type were originally introduced in spectral viscosity methods proposed by Gelb and Tadmor [2,19]. The intent of these schemes was to add even powered derivatives of the solution to damp higher frequencies while maintaining spectral accuracy. However, in our approach these additional terms only affect the residual on coarser meshes in the multigrid process, and do not affect the converged answer on the fine mesh. By damping high frequency errors they improve the projected coarse mesh correction to the finest solution.

Given the stability region of a typical RK time advancement scheme, we can determine the gain/damping of the scheme as a function of the frequencies in the discretized solution. For the purposes of this analysis, we will assume that R is a linear operator of the following form:

$$R = c\frac{\delta W}{\delta x} - \mu\Delta x^3\frac{\delta^4 W}{\delta x^4}. \tag{14}$$

The Δx^3 term corresponds to the higher order term frequently seen in shock-capturing schemes that use a blend of low and higher order diffusive terms depending on their proximity to the shock [11,12]. Using central difference operators, and assuming reasonable values for the constants in Eq. (14), one could write the spectral footprint of the overall discrete operator as

$$\lambda\Delta\tau = \lambda_t\Delta\tau - \tfrac{3}{8}(1-\cos(\rho))^2 + 2.5\sin(\rho)\mathrm{i}, \tag{15}$$

where

$$\lambda_t = V(\mathrm{i}k - \varepsilon_N(\mathrm{i}k)^{2s}). \tag{16}$$

This spectral footprint is drawn on the stability diagrams for an explicit scheme in Fig. 3(a). The blue line is calculated using a $\lambda_t\Delta_\tau = \frac{\mathrm{i}}{2}$ which is representative of an NLFD scheme without spectral viscosity. The red line is calculated with $\lambda_t\Delta_\tau = -\frac{1}{2} + \frac{\mathrm{i}}{2}$ corresponding to an NLFD scheme with spectral viscosity. Obviously, the addition of spectral viscosity shifts the ellipses farther to the left in the complex $\lambda\Delta\tau$ plane. The gain along the path of the ellipse is provided in subplot (b) of Fig. 3. The data show that the shift in the spectral footprint induced by the spectral viscosity lowers the gain of the time-stepping scheme for the majority of frequencies in the solution. Lower gains increase the damping and improve the efficiency of the multigrid solver. However, the stability region of the time-advancement scheme limits the magnitude of this shift, which is equivalent to the magnitude of the spectral viscosity term.

To illustrate the effects of this method on a NLFD calculation of the unsteady Euler equations, a convergence study was performed using a test case which highlights the dependence between convergence and frequency. The case uses a 193×49 C-mesh around a 64A010 pitching airfoil with boundary conditions approximating Davis's experiment [1] in AGARD Report 702 (CT Case 6, Dynamic Index 55). The only deviation from the parameters defining this case are an increase in the dynamic angle of attack to $\pm 3.0°$. This was chosen to increase the nonlinearities in the flow field ultimately providing a more challenging test case.

Fig. 4 shows the magnitude of all components of the unsteady residual as a function of the multigrid cycle. The plots in this figure show the residual convergence for solvers with and without coarse grid

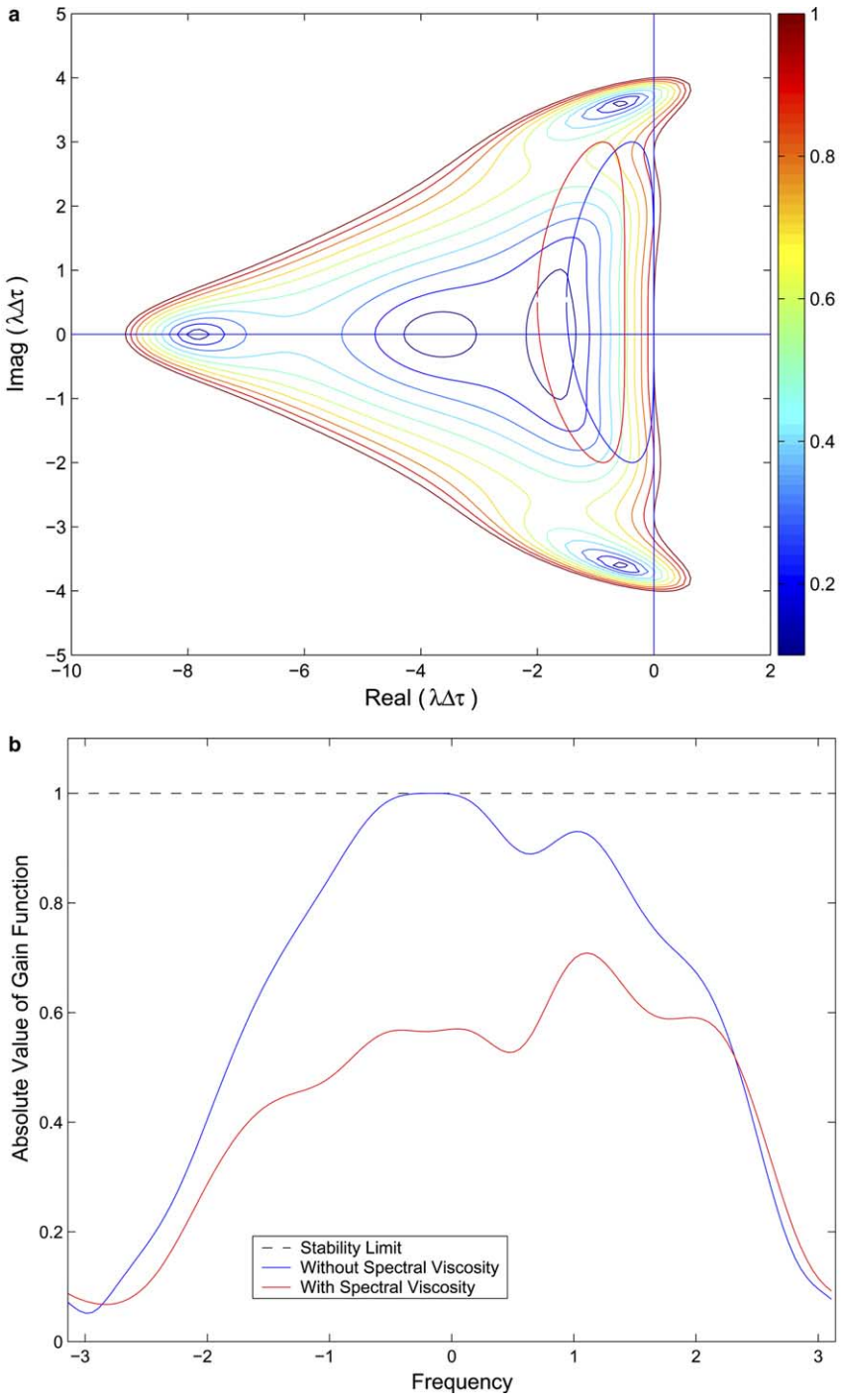

Fig. 3. (a) Stability diagram of the explicit time-stepping scheme. The red and blue ellipses are the spectral footprint of the discretization with and without spectral viscosity respectively, (b) The gain of the time-stepping scheme along the paths of the spectral footprint.

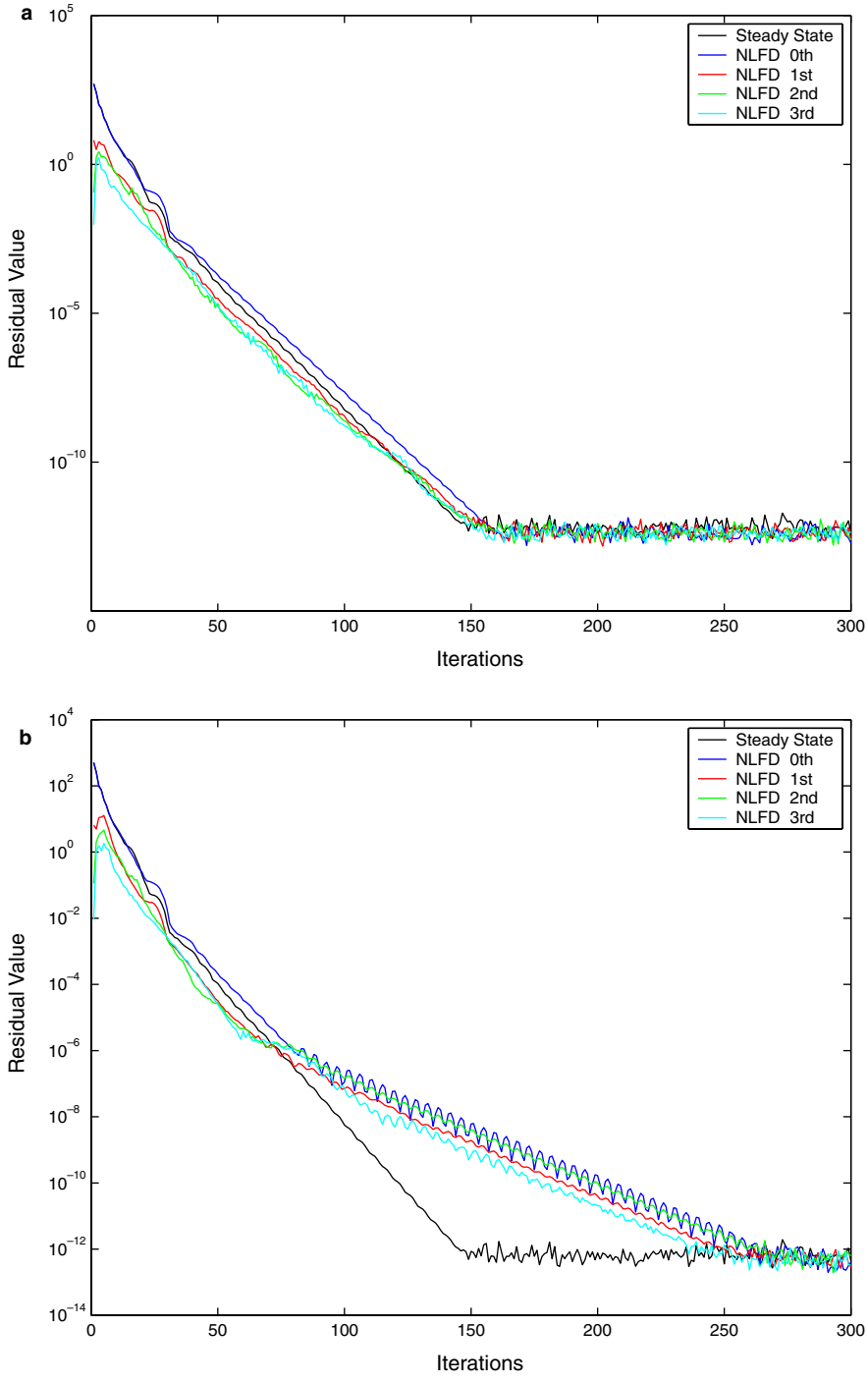

Fig. 4. Magnitude of all components of the unsteady density residual as a function of the multigrid cycle: (a) NLFD solver with coarse grid spectral viscosity and (b) NLFD solver without coarse grid spectral viscosity.

spectral viscosity. Over-plotted on these graphs is the convergence rate of the NLFD solver for the analogous steady problem lacking the pitching motion of the airfoil. Without spectral viscosity, the convergence rate of the unsteady NLFD solver is not equivalent to the steady-state case. However, Fig. 4(a) shows that the optimal convergence rate of the steady solver can be reclaimed for unsteady calculations through the use of spectral viscosity.

In the current formulation, the coarse grid spectral viscosity does not affect the fine grid solution. This is confirmed by subplot (a) of Fig. 5, which shows the difference between the ρe component of the solutions computed with and without spectral viscosity. The magnitude of the differences are consistent with the accuracy limitations of the floating point math used in the simulation. Although not shown, the other three components of the solution are similar; demonstrating that there is no effective difference between the solutions.

Given that this approach affects only the convergence and not the final answer, a number of terms of varying order with different coefficients $(s = 0, \frac{1}{4}, \frac{1}{2}, 1 (\varepsilon_N = \frac{0.1}{\frac{N}{2}-1} \rightarrow \frac{0.75}{\frac{N}{2}-1})$ and cutoff frequencies were tested on the pitching airfoil case. The cutoff frequency represents the lowest frequency to which the spectral viscosity term was applied; typically the fundamental harmonic. A variety of different test cases were attempted by varying the pitching frequency of the airfoil. Surprisingly the lower order dissipation schemes, including the zeroth order ($s = 0$), outperformed the higher order approaches. Regardless of the combination of coefficients, convergence performance continued to be a function of pitching frequency, and an approach that would automatically regain the optimal steady-state convergence rate over the variety of physical conditions was not found. However, using a coarse grid spectral term with $(\varepsilon_{\mathrm{N}} = \frac{0.75}{\frac{N}{2}-1})$ always improved the convergence performance over NLFD schemes not using this term. Given that this term is only calculated on the coarse grids, its impact on the computational cost can be considered negligible. The relative simplicity of implementing this term, its low cost, and consistent positive impact on convergence warrant the use of this approach in the future.

3. Efficiency comparison between the UFLO82 and NLFD solvers

The critical question is whether the non-linear frequency domain method offers a major reduction in computational cost over established time-accurate methods. This section addresses this question by presenting a comparison of a representative time-accurate code UFLO82 [10,18], with an NLFD code for Euler simulations of a pitching airfoil.

We begin by replacing the boundary conditions and spatial operators of the NLFD code with those of UFLO82, and establish that the two codes produce equivalent steady-state solutions. Next we conduct a convergence study to quantify the error in both codes as a function of temporal resolution. With the codes synchronized such that they produce solutions at equivalent error levels we can then compare each code's efficiency on the basis of computational cost. Since the spatial operators are equivalent, the comparison directly quantifies the efficiency of each method in representing the solution's temporal dimension.

3.1. Test case

Table 1 provides the boundary conditions employed by the pitching airfoil test case. Due to the mesh topology requirements of the UFLO82 code the grid used is an O-mesh. It was algebraically generated with 161 and 33 points in the circumferential and radial directions, respectively. The average farfield boundary distance for this grid is 129 airfoil chords.

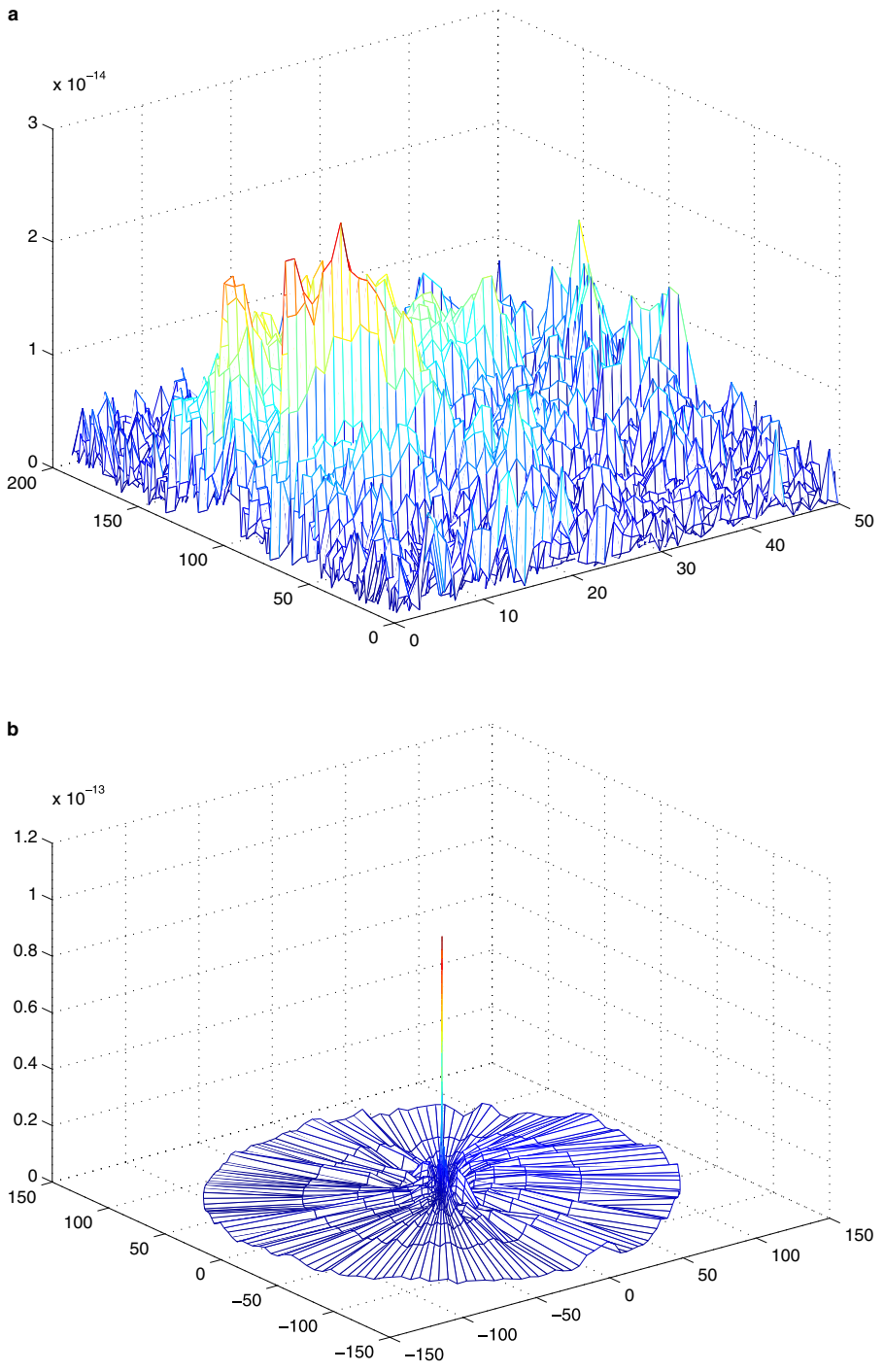

Fig. 5. Magnitude of the difference between the ρe component of two solutions: (a) two solutions computed with and without spectral viscosity shown at time $t = \frac{6}{7}T$ plotted in the computational space and (b) two steady-state solutions computed by UFLO82 and NLFD codes plotted in physical space.

Table 1
Description of the test case used in the comparison of UFLO82 and the NLFD codes

Parameter	Value
Airfoil	NACA 64A010
Mean angle of attack	0.0°
Angle of attack variation	±2.0°
Mach number	0.8
Reduced frequency	0.05

3.2. Comparison with UFLO82 For steady flow

We begin by verifying the equivalence of the spatial discretization and boundary conditions of the modified NLFD code with those contained in the UFLO82 code. Both UFLO82 and the modified NLFD code employ a cell centered finite volume scheme with JST artificial dissipation, a linear pressure extrapolation at the wall, and Riemann invariants in the farfield boundary condition [5–9]. Although both codes can compute unsteady flows, for this case they were run as steady-state codes with the dynamic angle of attack and the associated grid rotation terms set to zero. In order to eliminate the impacts of residual error on the solution, the number of multigrid cycles applied in the solution process was selected to drive the magnitude of all components of the residual to zero.

For verification, subplot (b) of Fig. 5 plots the difference between a component of the steady solutions produced by these two codes. Although not shown, all four components (ρ, ρu, ρv and ρe) of the two solutions match to machine accuracy, establishing the equivalence of the spatial operators and boundary conditions between the two codes.

3.3. Comparison with UFLO82 for unsteady flow

For unsteady flows, UFLO82 uses a dual time-stepping technique implementing a series of nested loops. The inner loop is a set of multigrid cycles used to drive the unsteady residual for that time step to a negligible value. This unsteady residual is written as the combination of an implicitly evaluated spatial residual and a second-order accurate A-stable discretization of the temporal derivative.

$$\frac{3W^{n+1} - 4W^n + W^{n-1}}{2\Delta t} + R(W^{n+1}) = 0. \tag{17}$$

Typically, the solution at the end of an inner loop is not converged to machine accuracy. In this case, the marginal change in the fluid properties from one additional multigrid cycle is not warranted due to the accuracy/stability requirements placed on the solution. For frequency domain solvers, the magnitude of the unsteady residual for all the wavenumbers is reduced to a negligible value by a series of multigrid cycles.

The outer loop of the dual time-stepping technique provides the time history of the discrete solution. The temporal accuracy of the solution is dictated by the time step employed by the solver which is directly related to the number of steps per temporal period in the solution. Analogously, the NLFD solver affects the temporal accuracy of its solution by specifying the number of modes used in the solution's representation. This directly affects the cost of the solver, by dictating the number of residual evaluations needed to be calculated in the pseudo-spectral approach.

The final parameter affecting the cost of a dual time-stepping solution is its proximity to a periodic steady state. Time-accurate solvers capture the decay of initial transients until the solution approaches a periodic steady state. Eventually the marginal decay of these transients produced from one additional time step is not warranted due to the accuracy requirements placed on the solution. The cost of resolving the

Table 2
Cost estimation for UFLO82 and NLFD solvers quantified in multigrid cycles

Variable	UFLO82 description	NLFD description
N_M	Number of multigrid cycles required for a solution at each time step	Number of multigrid cycles to solve the unsteady residual in the frequency domain
N_N	Number of time steps required to resolve a period of the solution	The ratio of the cost of an NLFD solution to a steady state solution. This is approximately the number of solution instances per time period
N_P	Number of time periods required to drive the initial transients to a negligible value, thereby reaching a periodic steady state	
Cost	$\approx N_M N_N N_P$	$\approx N_M N_N$

initial transients is eliminated in NLFD methods since these solvers admit only components of the solution which are periodic over a predefined length of time.

To summarize the previous discussion, Table 2 itemizes the costs associated with the dual time-stepping and NLFD methods. The comparison provided at the conclusion of this section will quantify each of these parameters and provide a computational cost for a solution in terms of the number of multigrid cycles used in its calculation.

3.3.1. Temporal resolution

To ensure that both codes were calculating solutions at equivalent error levels, a convergence study was performed to quantify error as a function of temporal resolution. This study will identify the number of steps per period used by UFLO82 that will produce a solution with equivalent error as the NLFD code using a given number of modes.

For the UFLO82 calculations, the residual at the end of each time step was driven to machine zero. Eight different temporal resolutions were calculated including 8, 16, 24, 32, 64, 128, 256 and 512 time steps per period. To eliminate any error associated with initial transients, 48 periods of the airfoil oscillation were simulated. As the solution evolved in time, Fourier transforms of the latest period of C_l and C_m data were computed. The magnitude of the fundamental harmonic of this statistic is used to quantify the error. The control solution was selected from the last oscillation of the most accurate solution (the UFLO82 calculation using 512 points per wave), and is subtracted from all other solutions to quantify the error due to the temporal resolution

$$\begin{aligned} E_l &= \left| |\widehat{C}_{l_l}| - |\widehat{C}_{l_{l_{512}}}| \right|, \\ E_m &= \left| |\widehat{C}_{m_l}| - |\widehat{C}_{m_{l_{512}}}| \right|. \end{aligned} \tag{18}$$

Similar runs were made with the NLFD code using one, two and three time varying modes. All components of the unsteady residual were driven to machine zero, and Fourier transforms of C_l and C_m were computed from the periodic steady-state solution. The magnitude of the fundamental harmonic was then subtracted from the control solution in a manner consistent with the UFLO82 results.

Fig. 6 shows the UFLO82 error for both lift and moment coefficients. The plots include the time history of error for the calculation producing the control solution; showing the difference in the figure of merit between the converged solution and the solution at any point in time. This is included to show that the error associated with the initial transients for the control calculation is significantly less than the overall error for any other calculation, and hence can be considered as insignificant.

Each solution approaches an error level representing the minimum error achievable in realistic environments. These minimum errors are replotted in subplot (a) of Figs. 7 and 8 as a function of the temporal

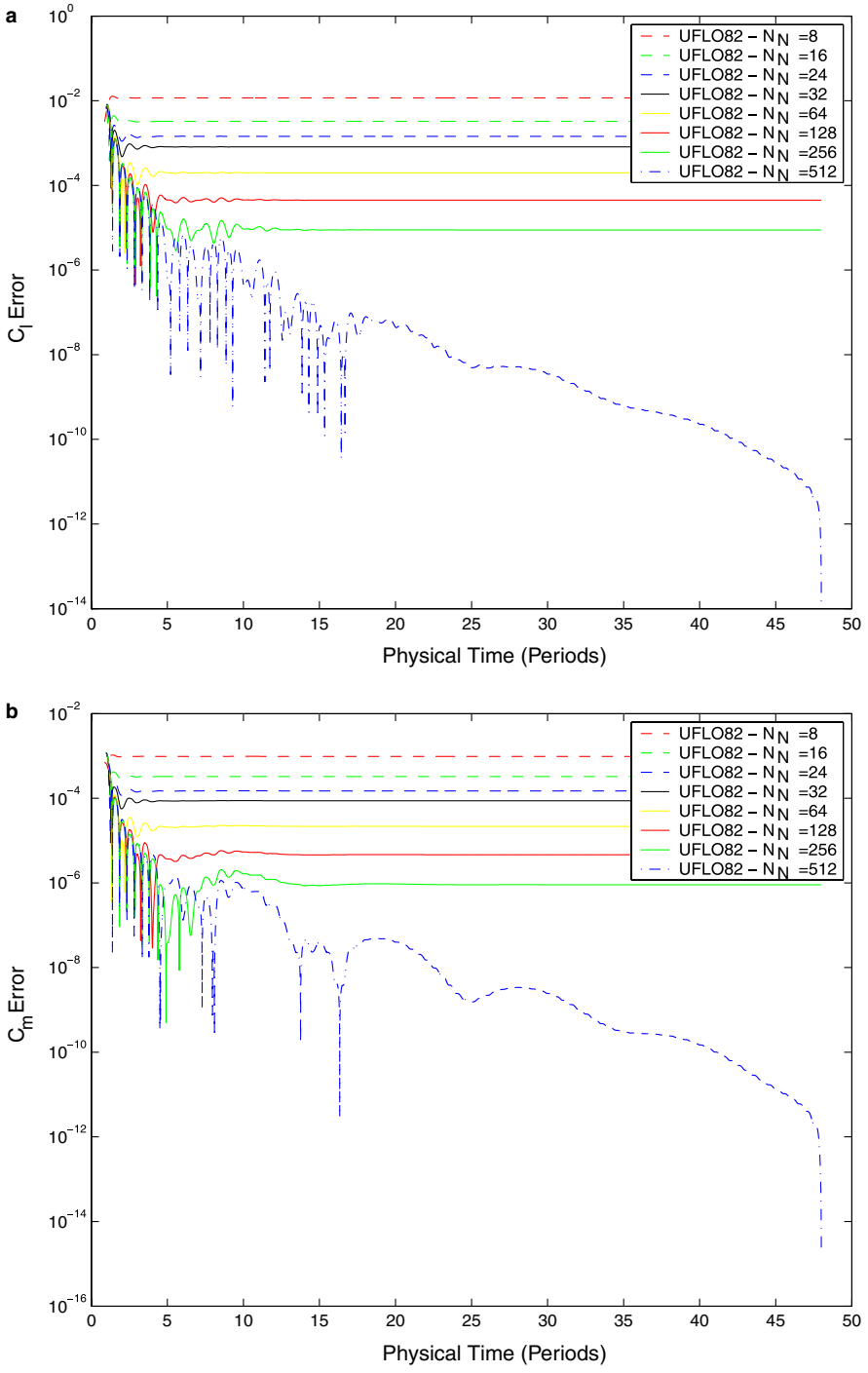

Fig. 6. Error as a function of physical time for various temporal resolutions of the UFLO82 code including the control solution: (a) lift and (b) moment.

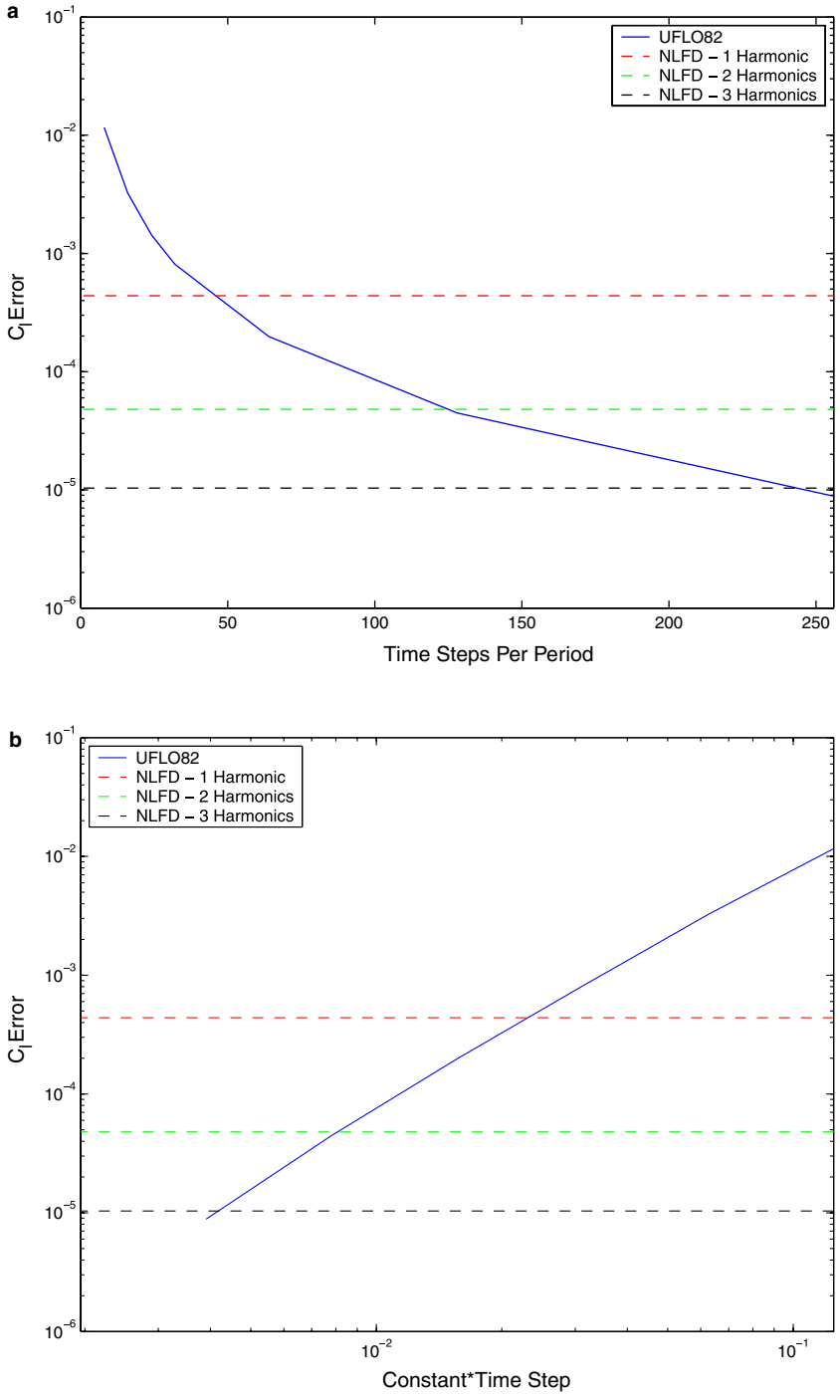

Fig. 7. Asymptotic error in lift computed at various temporal resolutions by the UFLO82 and NLFD codes: (a) semi-logarithmic scale and (b) full logarithmic scale.

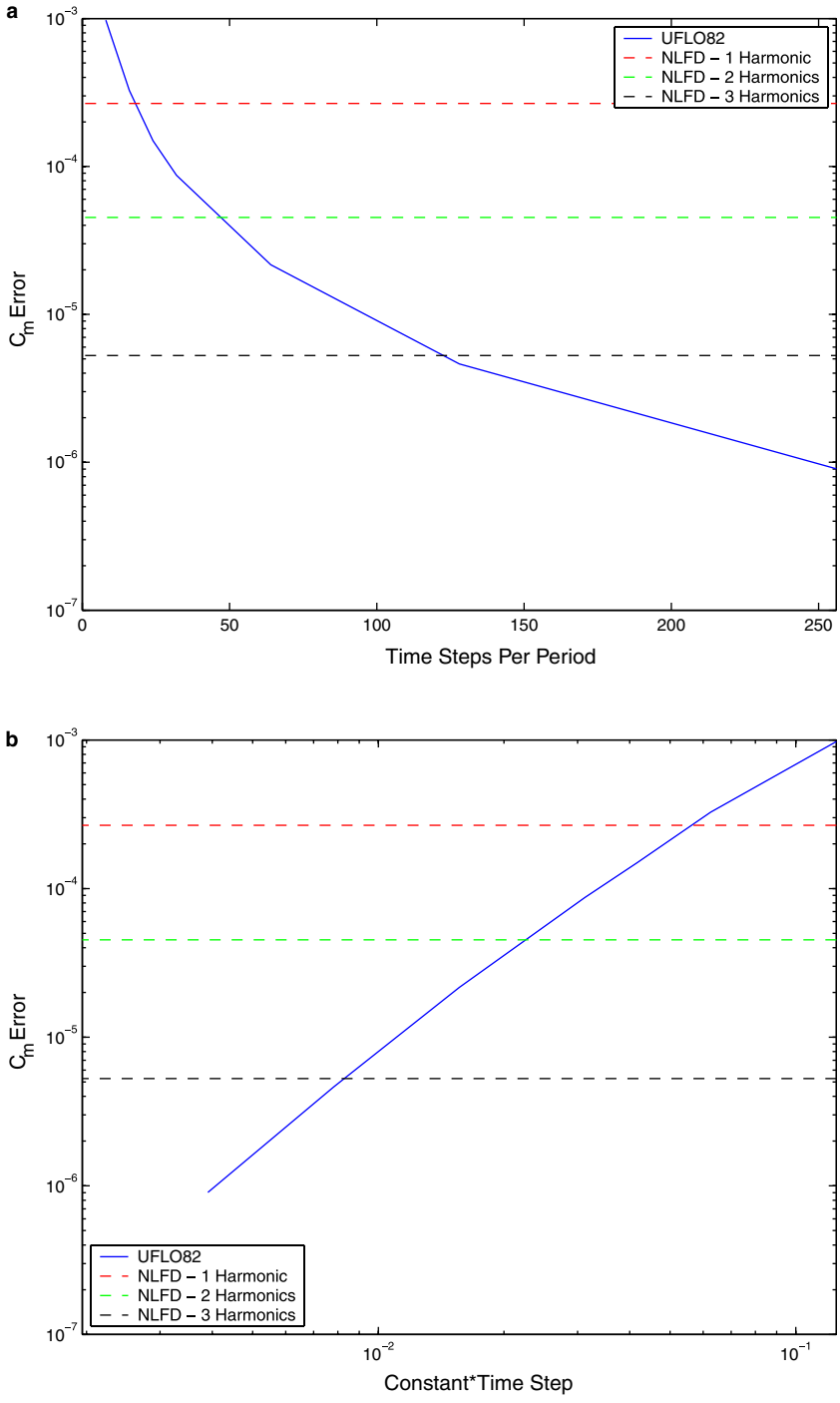

Fig. 8. Asymptotic error in moment computed at various temporal resolutions by the UFLO82 and NLFD codes: (a) semi-logarithmic scale and (b) full logarithmic scale.

Table 3
The number of time steps per period required by UFLO82 to reach error levels equivalent to results produced by an NLFD code using one, two, and thee time varying harmonics

NLFD modes	UFLO82 N_N	
	C_l	C_m
1	45	18
2	125	45
3	244	123

resolution used by each UFLO82 calculation. To check the accuracy of the UFLO82 code, subplot (b) provides the same data replotted on a full logarithmic scale showing that the UFLO82 error is proportional to Δt^2. Overplotted on all these figures are horizontal lines representing error levels in C_{l_1} and C_{m_1} based on NLFD calculations with varying numbers of temporal modes. The intersection of the NLFD lines with the UFLO82 curve shows the temporal resolutions each code must use to achieve equivalent error levels. These data are summarized in Table 3. Using C_l as a figure of merit, the NLFD code needs only 1 temporal mode to obtain similar error levels as a UFLO82 code using 45 time steps per period. Using C_m in the same capacity, the UFLO82 code needs only 18 points for one mode in the NLFD code. Regardless of the figure of merit, for every additional temporal mode added in the NLFD calculation, the temporal resolution of UFLO82 needs to be increased by roughly a factor of 2.5–3.

3.3.2. Multigrid cycles

In an attempt to quantify N_M for UFLO82, a convergence study was performed to identify the minimum number of multigrid cycles per time step required to reach the asymptotic error levels identified in Fig. 6. These error levels were functions of UFLO82 calculations where the residual was driven to machine zero at each time step. However, the residual error needs only to be small relative to the error due to the temporal discretization. Consequently an unbiased cost comparison of the methods should quantify the minimum number of multigrid cycles per time step required by UFLO82 to reach the equivalent error levels. Fig. 9 provides this data, for both C_l and C_m, for each temporal resolution identified in Table 3. In most cases, only six multigrid cycles per step were required.

A similar convergence study was performed for the NLFD code. Fig. 10 plots the error level as a function of the number of multigrid cycles. The intersection of the rightmost dashed line with the individual curves approximately identifies the minimum number of multigrid cycles required to reach the asymptotic error levels; these values are shown in Table 4.

3.3.3. Cost per multigrid cycle

The dominant cost within an NLFD solver, is the time associated with calculating instances of the spatial residual along the time period of the solution. Consequently the work per multigrid cycle scales with the number of time instances used to represent the solution. If N is the number of time varying modes, then the work per multigrid cycle should scale like $2N + 1$. Table 5 provides the measured execution time per NLFD multigrid cycle divided by the time required for a similar steady state cycle. As expected, the data approximates the growth rate cited above.

3.3.4. Decay of initial transients

The final factor in the cost of the UFLO82 solution is the time rate of decay of the initial transients. Fig. 11 plots the decay in C_{l_1} and C_{m_1} error over time calculated by UFLO82 for the 45 time steps per period case. Similar plots for the other temporal resolutions have been generated, but for the sake of brevity

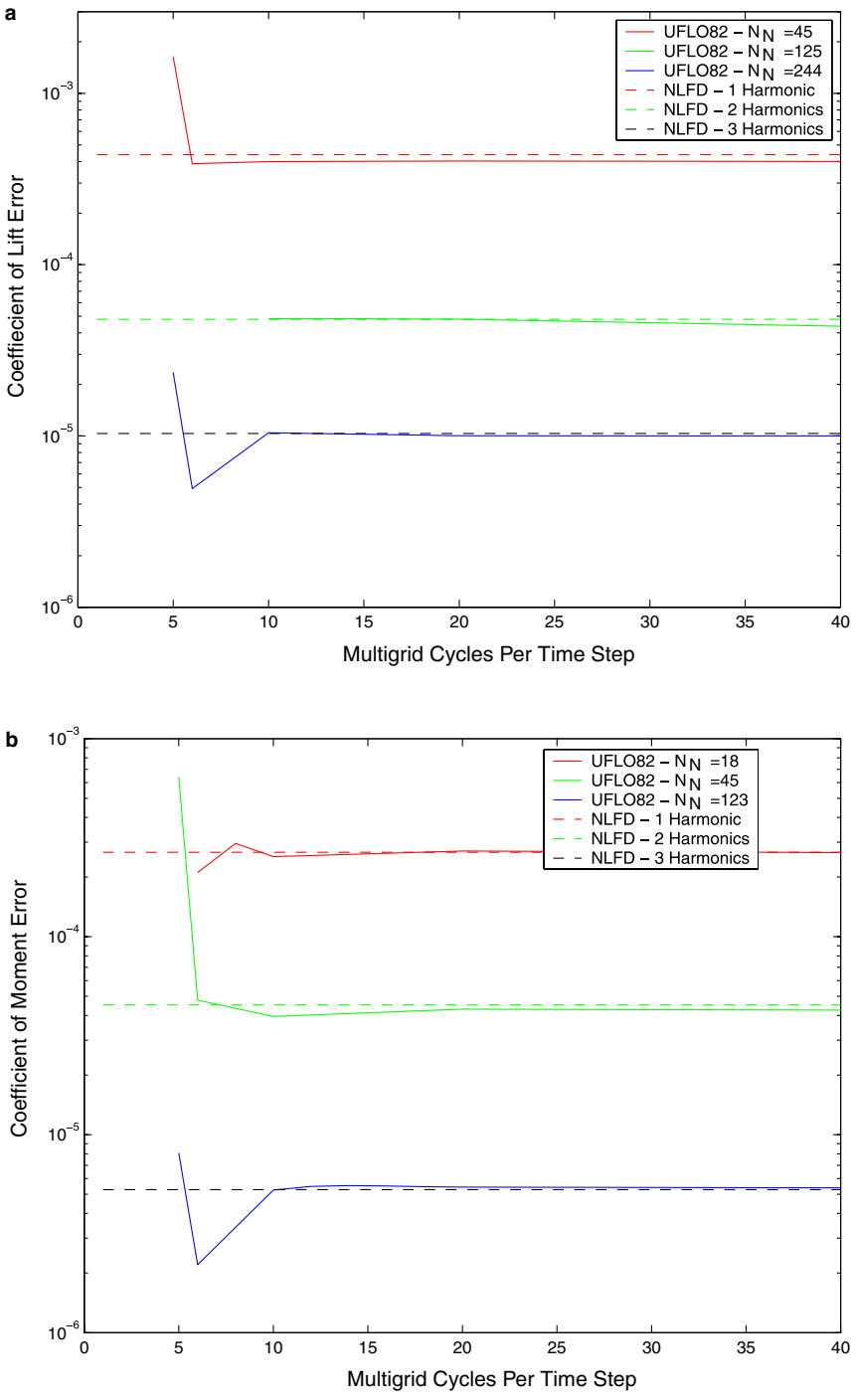

Fig. 9. The number of multigrid cycles required by the UFLO82 code to reach equivalent error levels: (a) lift and (b) moment.

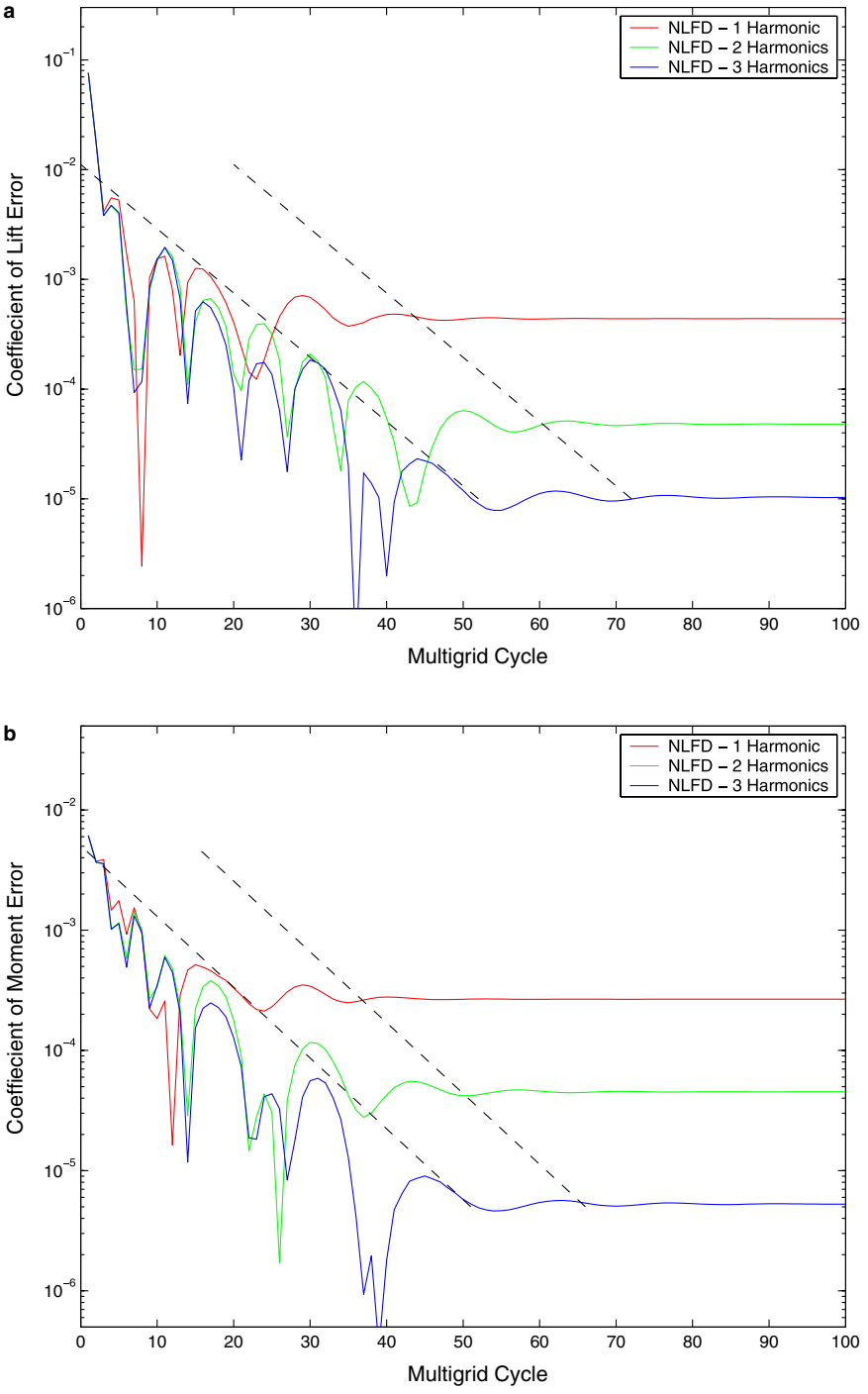

Fig. 10. The number of multigrid cycles required by the NLFD code to reach equivalent error levels: (a) lift and (b) moment.

Table 4
The number of multigrid cycles required by the NLFD solver to obtain a solution at equivalent error levels

NLFD modes	MG cycles	
	C_l	C_m
1	44	37
2	60	50
3	72	65

Table 5
The ratio of the execution time of an NLFD solution to a steady state solution, N_N, as a function of the number of unsteady modes used in the NLFD solution

Number of unsteady modes	0	1	2	3
$\frac{\text{NLFD cost}}{\text{Steady-state cost}}$	1.0	3.06	5.13	7.30

are not shown in this paper. Based on these data, Table 6 provides estimates of the number of periods required for the error to approach its asymptotic value. These numbers were selected conservatively to the benefit of the cost estimation for UFLO82.

3.3.5. Cost comparison

The cost of the two solvers can now be compared using the data provided in the previous sections. This comparison assumes that the work associated with a single multigrid cycle is equivalent between the two codes. This is an accurate approximation, given that UFLO82 and the NLFD code use the same pseudo-time advancement, residual averaging and multigrid aggregation and prolongation operators. This approach provides a comparison independent of the code implementation and compiler optimization algorithms.

Table 7 provides the relative cost data using C_l and C_m as the figures of merit. If the user requires equivalent error levels for both C_l and C_m data, then the lift results will drive the cost comparison. In this case, the NLFD code is a factor of 8–19 times faster than the UFLO82 code depending on the error level. In the worst case, using only C_m error as the basis for comparison, the NLFD code at the lowest temporal resolution is approximately 3 times faster the UFLO82 code. Not surprisingly, the marginal cost of the NLFD solver is also better than the UFLO82 code. For either figure of merit, the cost multiple between the two codes increases in favor of the NLFD method as the temporal accuracy is increased.

4. Sensitivity of the NLFD solver to flow conditions

4.1. Convergence versus reduced frequency

Consider the convergence rate of the steady residual an optimal performance condition. Obviously, increasing pitching frequency moves the NLFD residual away from its steady analogue. However, this section demonstrates that our implementation of the NLFD method will regain the optimal steady convergence rates; regardless of the temporal derivative's magnitude within the unsteady residual.

The test case uses the boundary conditions identified in Table 1 except that the reduced frequencies were varied over a range from 0 to 0.5. Solutions were calculated on a 193×49 C-mesh using three time varying

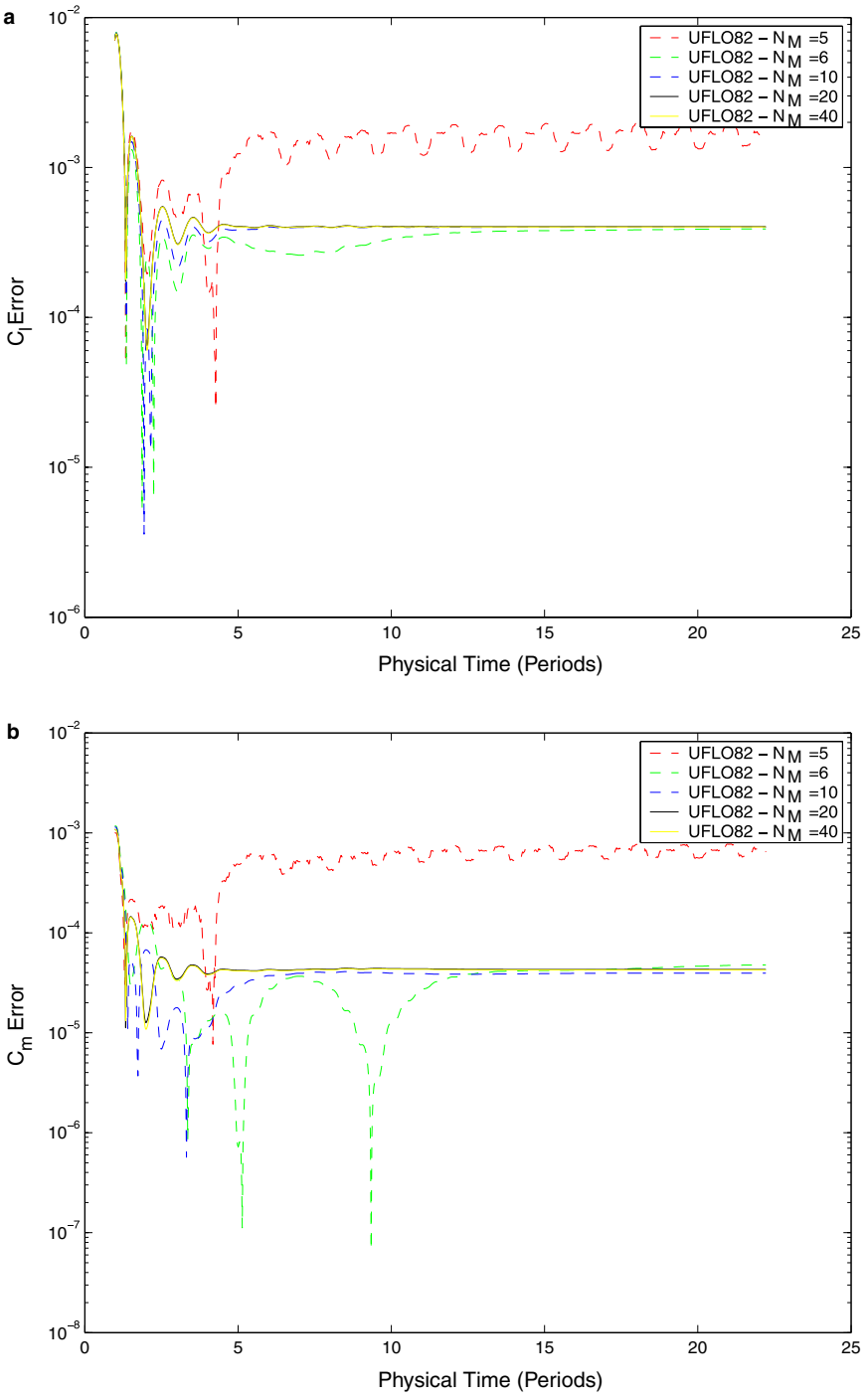

Fig. 11. Error as a function of physical time in units of periods for the UFLO82 solution calculated with 45 time steps per period: (a) lift and (b) moment.

Table 6
The number of time periods required by the UFLO82 solver to reach convergence for each temporal resolution

C_l		C_m	
N_N	Periods	N_N	Periods
45	4	18	3
125	5	45	4
244	7	123	6

Table 7
Cost comparison between UFLO82 and NLFD codes using both the errors in coefficient of lift and moment as the figure of merit

NLFD modes	Cost in multigrid cycles, $N_M \times N_N$	UFLO82, N_N	Cost in multigrid cycles, $N_M \times N_N \times N_P$
Error in C_{l_i} as figure of merit			
1	$44 \times 3.06 = 135$	45	$6 \times 45 \times 4 = 1080$
2	$60 \times 5.13 = 308$	125	$6 \times 125 \times 5 = 3750$
3	$72 \times 7.30 = 526$	244	$6 \times 244 \times 7 = 10248$
Error in C_{m_i} as figure of merit			
1	$37 \times 3.06 = 113$	18	$6 \times 18 \times 3 = 324$
2	$50 \times 5.13 = 257$	45	$6 \times 45 \times 4 = 1080$
3	$65 \times 7.30 = 475$	123	$6 \times 123 \times 6 = 4428$

modes such that the highest temporal component is oscillating at three times the fundamental frequency. All the calculations were performed holding constant any convergence acceleration parameters and artificial dissipation coefficients.

Subplot (a) within Fig. 12 shows the maximum over all the wavenumbers of the absolute value of the residual as a function of the multigrid cycle. The fastest converging solution was produced from a steady calculation (zero reduced frequency) at the mean angle of attack. The unsteady calculations exhibit similar convergence trends up to 100 multigrid cycles, or about a residual decay of about 10–11 orders of magnitude. At this point, the convergence rates begin to vary slightly until the residual reaches machine zero.

4.2. Convergence versus dynamic angle of attack

Unlike reduced frequency, the magnitude of the dynamic angle of attack terms are not directly included in the temporal derivative terms of the NLFD formulation. However, the effect of increasing the dynamic angle of attack is to increase the flow field nonlinearities and subsequent coupling between temporal modes. As such the sensitivity of the solver convergence rates with respect to this parameter should be investigated.

The test case uses the boundary conditions identified in Table 1 except that the dynamic angle of attack is varied from $\pm 0°$ to $\pm 2.8°$. Again, convergence acceleration parameters and artificial dissipation coefficients are held constant to ensure convergence rates are strict functions of dynamic angle of attack

Subplot (b) within Fig. 12 shows the maximum over all the wavenumbers of the absolute value of the residual as a function of the multigrid cycle. The fastest converging solution was predictably produced by the steady solver (dynamic angle of attack at $\pm 0°$). Overall, little variation is found over the range of dynamic angles of attack, with the poorest converging solution requiring only 10 additional multigrid cycles to reach machine zero.

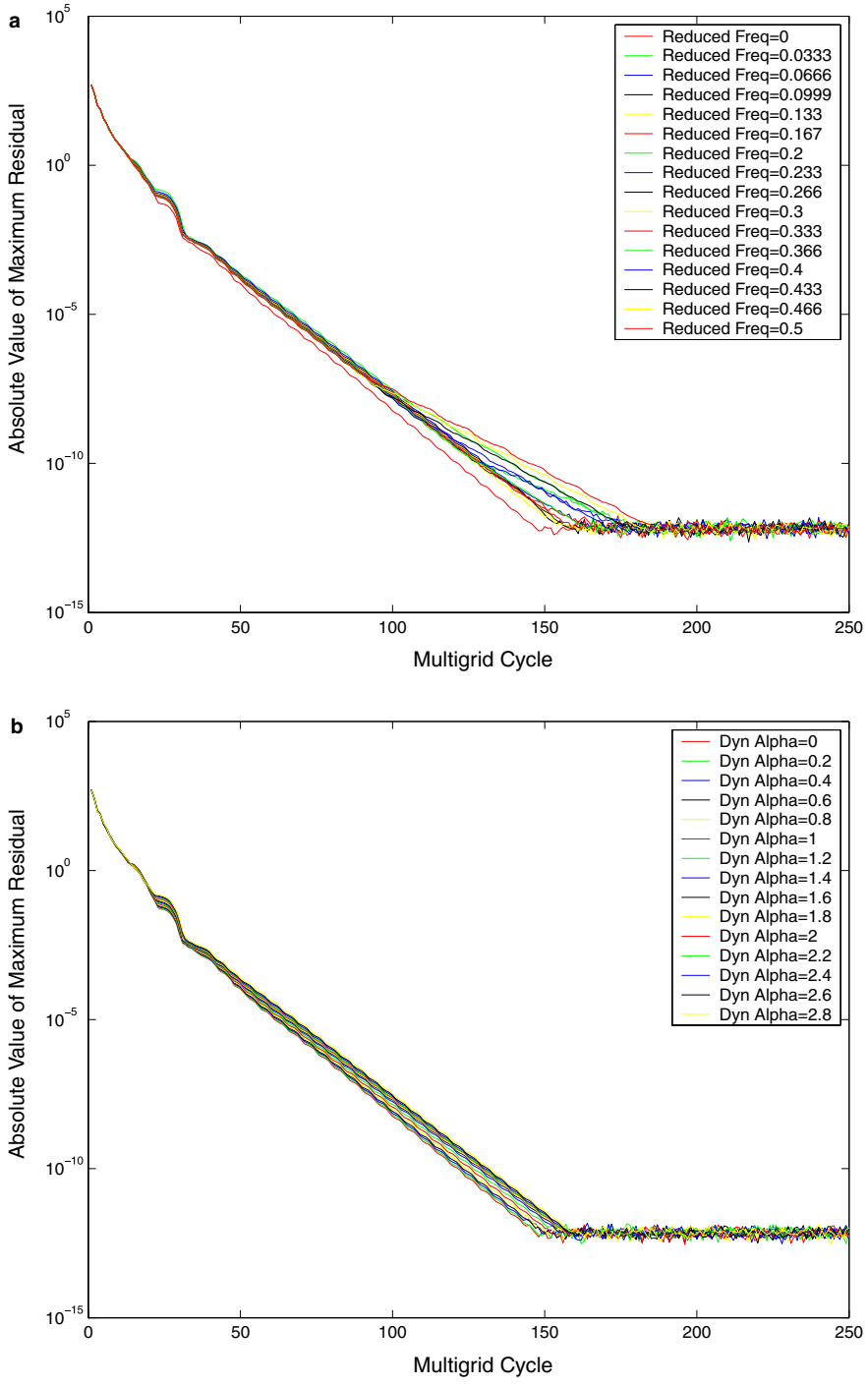

Fig. 12. The maximum of the NLFD residual as a function of the number of multigrid cycles completed: (a) over a sweep of reduced frequencies and (b) over a sweep of dynamic angles of attack.

5. Conclusion

The computational efficiency of the NLFD solver has been compared to that of UFLO82, a representative time accurate code. Unbiased comparisons were made between the methods by measuring the amount of work required by each solver to reach equivalent error levels. Using error in coefficient of lift as the figure of merit the NLFD code is 8–19 times faster than UFLO82. Using coefficient of moment as the basis for comparison, the NLFD method is roughly three to nine times faster than the time-accurate approach. The ratio of efficiency between the codes is provided as a range because the NLFD method gains efficiency at higher temporal resolutions. In addition, numerical experiments confirm that the NLFD solver retains its efficiency over a wide range of unsteady flow conditions.

References

[1] S.S. Davis, NACA 64A010 (NASA Ames Model) oscillatory pitching, AGARD Report 702, AGARD, January 1982, Dataset 2.
[2] A. Gelb, E. Tadmor, Enhanced spectral viscosity approximations for conservation laws, Applied Numerical Mathematics 33 (2000) 3–21.
[3] K.C. Hall, J.P. Thomas, W.S. Clark, Computation of unsteady nonlinear flows in cascades using a harmonic balance technique, Technical Report, 9th International Symposium on Unsteady Aerodynamics, Aeroacoustics and Aeroelasticity of Turbomachines, Lyon, France, September 2000.
[4] K.C. Hall, J.P. Thomas, W.S. Clark, Computation of unsteady nonlinear flows in cascades using a harmonic balance technique, AIAA Journal 40 (2002) 879–886.
[5] A. Jameson, Solution of the Euler equations for two dimensional transonic flow by a multigrid method, Applied Mathematics and Computation 13 (1983) 327–355.
[6] A. Jameson, Transonic flow calculations, Princeton University Report MAE 1651, Princeton University, 1984, in: F. Brezzi (Ed.), Numerical Methods in Fluid Dynamics. Lecture Notes in Mathematics, vol. 1127, Springer-Verlag, 1985, pp. 156–242.
[7] A. Jameson, Multigrid methods for compressible flow calculations, in: Proceedings of the Second European Conference on Multigrid MethodsLecture Notes in Mathematics, Springer, 1986, pp. 166–201.
[8] A. Jameson, Computational transonics, Communications on Pure Applied Mathematics 16 (1988) 507–549.
[9] A. Jameson, Computational aerodynamics for aircraft design, Science 245 (1989) 361–371.
[10] A. Jameson, Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings, Technical Report 91-1596, AIAA 10th Computational Fluid Dynamics Conference, June 1991.
[11] A. Jameson, Analysis and design of numerical schemes for gas dynamics I artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence, RIACS Technical Report 94.15, International Journal of Computational Fluid Dynamics 4 (1995) 171–218.
[12] A. Jameson, Analysis and design of numerical schemes for gas dynamics II artificial diffusion and discrete shock structure, RIACS Report No. 94.16, International Journal of Computational Fluid Dynamics 5 (1995) 1–38.
[13] L. Martinelli, A. Jameson, Calculations of viscous flows with a multigrid method, Dissertation, Princeton University, October 1987.
[14] M. McMullen, A. Jameson, J. Alonso, Acceleration of convergence to a periodic steady state in turbomachinery flows, AIAA paper 01-0152, AIAA 39th Aerospace Sciences Meeting, Reno, NV, January 2001.
[15] M. McMullen, A. Jameson, J. Alonso, Application of a non-linear frequency domain solver to the Euler and Navier–Stokes equations, AIAA paper 02-0120, AIAA 40th Aerospace Sciences Meeting, Reno, NV, January 2002.
[16] M.S. McMullen, A. Jameson, J.J. Alonso, The application of non-linear frequency domain methods to the Euler and Navier–Stokes methods, Dissertation, Stanford University, March 2003.
[17] N.D. Melson, M.D. Sanetrik, H.L. Atkins, Time-accurate Navier–Stokes calculations with multigrid acceleration, in: Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods, Copper Mountain, April 1993.
[18] N.A. Pierce, J.J. Alonso, Efficient computation of unsteady viscous flow by an implicit preconditioned multigrid method, AIAA Journal 36 (1998) 401–408.
[19] E. Tadmor, Convergence of spectral methods for nonlinear conservation laws, SIAM Journal on Numerical Analysis 26 (l) (1989) 30–44.
[20] J. Yao, R. Davis, J.J. Alonso, A. Jameson, Unsteady flow investigations in an axial turbine using the massively parallel flow solver TFLO, Technical Report 01-0529, AIAA 39th Aerospace Sciences Meeting and Exhibit, January 2001.