A Time-Spectral Hybridizable Discontinuous Galerkin Method for Periodic Flow Problems

by

Hemant Kumar Chaurasia
B.Sc. and B.E. (Honours), Monash University, 2008
S.M., Massachusetts Institute of Technology, 2010

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2014

Signature redacted

Author .....
Department of Aeronautics and Astronautics
May 22, 2014

Signature redacted

Certified by .....
Jaime Peraire
Department Head & H.N. Slater Professor of Aeronautics and Astronautics
Thesis Supervisor

Signature redacted

Certified by ...
Mark Drela
Terry J. Kohler Professor of Fluid Dynamics
Thesis Committee Member

Certified by ..... Signature redacted .....
Ngoc-Cuong Nguyen
Principal Research Scientist

Signature redacted

Thesis Committee Member

Accepted by .....
Paulo C. Lozano
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

A Time-Spectral Hybridizable Discontinuous Galerkin Method for Periodic Flow Problems

by

Hemant Kumar Chaurasia

Submitted to the Department of Aeronautics and Astronautics
on May 22, 2014, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Numerical simulations of time-periodic flows are an essential design tool for a wide range of engineered systems, including jet engines, wind turbines and flapping wings. Conventional solvers for time-periodic flows are limited in accuracy and efficiency by the low-order Finite Volume and time-marching methods they typically employ. These methods introduce significant numerical dissipation in the simulated flow, and can require hundreds of timesteps to describe a periodic flow with only a few harmonic modes. However, recent developments in high-order methods and Fourier-based time discretizations present an opportunity to greatly improve computational performance.

This thesis presents a novel Time-Spectral Hybridizable Discontinuous Galerkin (HDG) method for periodic flow problems, together with applications to flow through cascades and rotor/stator assemblies in aeronautical turbomachinery. The present work combines a Fourier-based Time-Spectral discretization in time with an HDG discretization in space, realizing the dual benefits of spectral accuracy in time and high-order accuracy in space. Low numerical dissipation and favorable stability properties are inherited from the high-order HDG method, together with a reduced number of globally coupled degrees of freedom compared to other DG methods. HDG provides a natural framework for treating boundary conditions, which is exploited in the development of a new high-order sliding mesh interface coupling technique for multiple-row turbomachinery problems. A regularization of the Spalart-Allmaras turbulence model is also employed to ensure numerical stability of unsteady flow solutions obtained with high-order methods.

Turning to the temporal discretization, the Time-Spectral method enables direct solution of a periodic flow state, bypasses initial transient behavior, and can often deliver substantial savings in computational cost compared to implicit time-marching. An important driver of computational efficiency is the ability to select and resolve only the most important frequencies of a periodic problem, such as the blade-passing frequencies in turbomachinery flows. To this end, the present work introduces an

adaptive frequency selection technique, using the Time-Spectral residual to form an inexpensive error indicator. Having selected a set of frequencies, the accuracy of the Time-Spectral solution is greatly improved by using optimally selected collocation points in time. For multi-domain problems such as turbomachinery flows, an anti-aliasing filter is also needed to avoid errors in the transfer of the solution across the sliding interface. All of these aspects contribute to the Adaptive Time-Spectral HDG method developed in this thesis.

Performance characteristics of the method are demonstrated through applications to periodic ordinary differential equations, a convection problem, laminar flow over a pitching airfoil, and turbulent flow through a range of single- and multiple-row turbomachinery configurations. For a 2:1 rotor/stator flow problem, the Adaptive Time-Spectral HDG method correctly identifies the relevant frequencies in each blade row. This leads to an accurate periodic flow solution with greatly reduced computational cost, when compared to sequentially selected frequencies or a time-marching solution. For comparable accuracy in prediction of rotor loading, the Adaptive Time-Spectral HDG method incurs 3 times lower computational cost (CPU time) than time-marching, and for prediction of only the 1st harmonic amplitude, these savings rise to a factor of 200. Finally, in three-row compressor flow simulations, a high-order HDG method is shown to achieve significantly greater accuracy than a lower-order method with the same computational cost. For example, considering error in the amplitude of the 1st harmonic mode of total rotor loading, a $p = 1$ computation results in 20% error, in contrast to only 1% error in a $p = 4$ solution with comparable cost. This highlights the benefits that can be obtained from higher-order methods in the context of turbomachinery flow problems.

Thesis Supervisor: Jaime Peraire
Title: Department Head & H.N. Slater Professor of Aeronautics and Astronautics

Acknowledgments

I am deeply grateful to all the people who helped make this thesis possible. First and foremost, I would like to thank my thesis advisor Prof. Jaime Peraire, for his central role in what has proven to be the opportunity of a lifetime. I have learned so much from him, not just in his capacity as a brilliant and passionate researcher, but also as an educator, a presenter, a writer, a caring mentor and a friend. His patient guidance and good humor have made this journey together a great one.

I would also like to thank Dr. Cuong Nguyen for all his efforts as a devoted mentor and collaborator, always willing to work with me to solve problems and explore new ideas. It has been a great pleasure to work with such a talented scholar, and to count him as a friend. I also thank Prof. Mark Drela for all of his contributions as a member of my Thesis Committee. I have gained much from his depth of expertise in aerodynamics and his passion for all that he does.

Thank you to my Thesis Readers, Prof. David Darmofal and Dr. Mani Sadeghi, whose careful review and constructive comments were invaluable in preparing this thesis. Further, I would like to thank Mani Sadeghi, Beth Lurie, Oliver Atassi and Huan-Min Shang at Pratt & Whitney for what has been an extraordinarily fruitful and enjoyable collaboration.

I am indebted to Dr. Xevi Roca for his academic mentorship, insightful discussions, and the great pleasure of his friendship. I could always count on him when I needed a fresh perspective on a research problem, or just a break to laugh a little bit, and his friendly encouragement kept me on the right track when I needed it most.

It has been a privilege to share these years with an amazing group of students in the ACDL, who have contributed in many ways to this work and to life outside the lab. Particular thanks go to Joel Saa and Ferran Vidal for not only being outstanding colleagues and labmates, but also for welcoming me to stay at their place during much of the preparation of this thesis; to David Moro for many helpful discussions, fun times and Spanish phrases; to Xun Huan for having my back in research, quals, ice

hockey, and the 100K; to Abby Men for her encouragement and intellectual jokes; to Masayuki Yano and Huafei Sun for their insights and advice; to Andrew March for runs around the river that helped my code to run too; to Chelsea He for sharing the fun of outreach; to David Lazzara, Eric Liu, TC, Eric Dow and Patrick Blonigan; to Marvin for being such a good guy; and to Alejandra and Mody for their support right from Day 1. I would also like to thank Jean, Joyce, Meghan, Ping and Beth for all their work to keep things running so smoothly.

Thanks also to many great friends who have supported me from near and far, including Carmen Guerra, Zach Bailey, Sydney Do, Whitney Lohmeyer, Risa Kawai, Caleb & Suzanne, James & Kirstie, Danny Li, Prof. Matthew Bailes, Prof. Mark Thompson, and friends from Monash, Kitzbuhel and the 100K.

I am so grateful to my parents Maneesh and Anu and my sister Ritu for their boundless love and support, the foundation from which everything else has followed. It is a great source of courage to know that you are always there for me, no matter the distance. And finally, I would like to thank my wife Dilani, my very best collaborator in all things. You have a way of bringing out the best in me, filling our world with joy, and turning extraordinary possibilities into real adventures that we share. Simply put, you make my dreams come true.

* * *

I gratefully acknowledge financial support for this work, which was provided in part by Pratt & Whitney, the Singapore-MIT Alliance, and the US Air Force Office of Scientific Research.

Contents

List of Figures

Chapter 1

Introduction

1.1 The Need for Better Periodic Flow Solvers

Time-periodic aerodynamic flows play a central role in a wide range of engineered systems, including jet engines, wind turbines, and flapping wings. With the ongoing exponential growth of computing power, engineers are increasingly relying on Computational Fluid Dynamics (CFD) software for cost-effective prediction of aerodynamic performance during the design process. However, existing CFD algorithms often struggle to achieve a high level of accuracy without incurring an impractically high computational cost. These difficulties arise from both the spatial and temporal discretizations typically employed:

- **Spatial:** Most high-fidelity CFD tools in use today are based on low-order Finite Volume discretizations, which are at most 2^{nd}-order accurate in space. *Grid-converged* solutions are often prohibitively expensive to obtain by these methods, forcing engineers to accept a significant degree of error in their simulations.

- **Temporal:** Time-marching methods (such as Runge-Kutta) are most common, though poorly suited to time-periodic problems. They do not exploit periodicity and they require time-accurate integration of initial transient behavior. As a result, several periods of integration and hundreds of timesteps may be required

to compute a periodic flow that is equivalently described by only a few Fourier modes.

These algorithmic limitations form a considerable barrier to the productive use of CFD in the design of jet engines and other periodic flow devices. New methods are needed that will improve the accuracy and efficiency of periodic flow simulations.

A Perspective From Industry

The following quotation further illustrates the need for advances in periodic flow solvers, from the perspective of an engineer in the turbomachinery industry:

> "High-cycle fatigue (HCF) is a major concern in turbomachinery design. This is especially true for aircraft engines which operate under a wide range of conditions. Requirements of low cost, low weight, small size, and high efficiency all have the tendency to increase susceptibility to aeromechanics, as cheaper or lighter materials are chosen, blade row gaps are reduced, blade aspect ratios are increased, and turbine blades are pushed toward thermal material limits. Computational design tools that predict HCF behavior, especially resonance response due to multi-row interaction, are employed to reduce the need for expensive testing. To be applicable to design optimization over a range of operating conditions in a manageable amount of time, these tools need to predict periodic flows with high accuracy and computational speed." [1]

[1] Mani Sadeghi, PhD, Staff Engineer, Pratt & Whitney (personal communication, May 6, 2014)

1.2 Thesis Objectives

Leading on from the motivation described in the previous section, the objective of this thesis is to develop and demonstrate a more accurate and efficient *grid-convergent* method for solving time-periodic aerodynamic flow problems, with a particular focus on flow through cascades and rotor/stator assemblies in aeronautical turbomachinery. The goal is to design a method that is both more *accurate* (able to achieve grid-converged solutions) and *efficient* (lower computational cost for a given level of accuracy) than the current state of the art.

To this end, contributions of this thesis are summarized as follows:

1. A Time-Spectral Hybridizable Discontinuous Galerkin (HDG) method for the efficient high-order solution of time-periodic flow problems

2. A high-order sliding interface coupling method for HDG, with an anti-aliasing filter for Time-Spectral discretizations

3. Application of the time-marching and Time-Spectral HDG methods to single- and multiple-row turbomachinery flow problems, with a regularized RANS-SA turbulence model suitable for higher-order methods

4. A Frequency Adaptive Time-Spectral method to solve a periodic problem using a non-sequential set of frequencies, and to provide an inexpensive error indicator to guide selection of frequencies

5. Quantification of the gains in accuracy and efficiency that can be obtained using the Time-Spectral HDG method, from an industrial perspective, comparing high-order to low-order and Time-Spectral to time-marching

1.3 Review of Methods for Periodic Flow Problems

Periodic flow problems arise in a broad range of engineering applications, and simulation techniques for such flows have a long history of development. This section provides a context for the present thesis by reviewing previous work in high-fidelity methods for time-periodic flow problems. The review is divided into two categories: (1) high-order spatial discretizations, and (2) time discretizations for periodic flows.

1.3.1 High-Order Spatial Discretizations

A central goal of this thesis is to demonstrate the benefits that can be obtained from using a high-order discretization in space, as opposed to the more commonly used low-order methods. High-order methods show promise for delivering more accurate solutions for a given computational cost, and the Hybridizable Discontinuous Galerkin (HDG) method stands out as a particularly attractive choice among high-order methods. However, before examining HDG in more detail, let us provide some context for HDG through a brief review of spatial discretization methods for flow simulation.

The first numerical methods for solving flow equations relied on Finite Difference spatial discretizations, such as those reviewed by Sod [92]. Strengths of these methods included their simplicity of implementation for simple differential operators, and their computational efficiency on simple structured meshes. However, as computational power grew and investigators sought to solve more advanced equations on complex geometries, Finite Difference methods proved increasingly inadequate.

These limitations were overcome by a new class of Finite Volume methods, which enabled a new level of geometric flexibility. Following early work on structured grids [59], Finite Volume methods were found to be particularly well-suited for solving conservation laws on arbitrary unstructured meshes [6]. Unlike Finite Difference methods, they guarantee numerical conservation of physically conserved quantities, and are easily applied to problems on arbitrarily complex geometries. Together with robust numerical stability properties, Finite Volume methods emerged as the method of

choice for Computational Fluid Dynamics (CFD) in industry, and to this day remain ubiquitous in commercial CFD packages.

However, the Finite Volume methods most commonly used in industry become very inefficient at high levels of accuracy. These methods can be described as *low-order* accurate in space, as they have a spatial truncation error which scales with mesh size h no faster than $\mathcal{O}(h^2)$. They have a long history of use in industry, and are very robust. However, grid-converged solutions are often too expensive to obtain, as solutions computed on successively finer grids do not improve in accuracy at a sufficient rate. This implies that users often have no choice but to accept a substantial degree of discretization error in their simulations, which can be manifested as numerical dissipation and dispersion.

High-order methods offer a solution to this problem. With a faster solution convergence rate, these methods open up the possibility of greater solution accuracy for a given computational cost. In this thesis we will refer to a method as *high-order* if it has a spatial truncation error that scales with mesh size faster than $\mathcal{O}(h^2)$. While high-order accurate Finite Volume methods have been developed [7, 51, 64], these use reconstruction techniques that can be complicated to implement for complex geometries, making them less common in industrial use. High-order Finite Difference methods also exist, but are only applicable to simple geometries with structured grids.

Finite Element methods, on the other hand, are readily formulated with high-order accuracy in space and are well equipped for complex geometries. In these methods, the discrete solution is represented on each element by a combination of high-degree polynomial basis functions. Finite Element methods have long been formulated this way, such as in the p-version of the Finite Element method [4], the Spectral Element method [80], and the streamline-upwind Petrov-Galerkin method (SUPG) [57, 58]. As demonstrated for hp-FEM by Babuska [3], increasing the polynomial order p can result in much faster convergence than refining the mesh at a constant p, when compared in terms of the total number of degrees of freedom. High-order Finite Element methods also accommodate high-order representation of curved boundary geometries, leading

to further enhancement of solution accuracy.

Discontinuous Galerkin (DG) methods are a particular type of high-order Finite Element method that has attracted growing interest in recent years. DG methods feature both the guaranteed local conservation of Finite Volume methods and the high-order accuracy of Finite Element methods. In contrast to the continuous Finite Element methods, DG numerical solutions are discontinuous between elements. These discontinuities introduce a larger number of total degrees of freedom, but also provide a richer approximation space and a natural means to stabilize the numerical scheme through terms in the fluxes between elements.

Looking back at key milestones in the development of DG methods, Reed & Hill were the first to introduce a DG method, in their application to the neutron transport equation in 1973 [85]. However, DG received little further attention until the next decade, when Johnson & Pitkaranta [60] and Richter [86] provided sharp *a priori* error estimates. In the years that followed, Chavent & Cockburn [15] applied slope-limiting, and Cockburn & Shu [29, 31] applied Runge-Kutta timestepping to the DG method (RKDG). DG methods were generalized for convection-diffusion systems by Bassi & Rebay [8], further generalized in the Local DG (LDG) method by Cockburn & Shu [30], and made more efficient through the Compact DG (CDG) method developed by Peraire & Persson [82].

DG methods have several important advantages over Finite Difference and Finite Volume methods, including that they work well on arbitrary meshes, result in stable high-order accurate discretizations of the convective and diffusive operators, allow for a simple and unambiguous imposition of boundary conditions, and are very flexible to parallelization and adaptivity [75]. However, they are also very computationally expensive due to the large number of degrees of freedom they employ, arising from the duplication of nodes at element interfaces to represent discontinuous solutions.

Recently, a successful approach for overcoming the computational expense of DG methods has been introduced in the Hybridizable Discontinuous Galerkin (HDG) method. HDG methods feature a significantly reduced number of globally-coupled

degrees of freedom, compared to previous DG methods. HDG was first introduced for elliptic problems by Cockburn *et al.* [20, 22, 24, 26–28], and developed for convection-diffusion problems by Cockburn *et al.* and Nguyen, Peraire & Cockburn [21, 72, 73]. Several authors then extended HDG methods to linear elasticity [93], Maxwell's equations [76], incompressible flow [23, 25, 74, 77], and the Euler and compressible Navier-Stokes equations [77, 81].

The key innovation of HDG is to parametrize the high-order solution on each element in terms of the *approximate trace* of the solution, which is defined only on the borders of each element (also known as the skeleton of the mesh). This decouples the system into a smaller global problem for the approximate trace, and a set of local problems on each element that can be solved independently once the approximate trace is known. This approach has substantially fewer globally-coupled degrees of freedom than earlier DG methods, offering significant savings in computation time and memory usage. In terms of convergence rate, HDG surpasses earlier DG methods by providing optimal convergence of all variables including the approximate gradient, for convection-diffusion problems. This can be exploited by a local postprocessing procedure to obtain even more accurate solutions. HDG also provides a framework for treating boundary conditions in a natural and systematic manner, through the definition of appropriate numerical fluxes on the domain boundaries. Finally, HDG is highly amenable to adaptation and parallelization [75], as demonstrated in recent work on scalable parallelization of HDG for compressible flow by Roca, Nguyen & Peraire [87].

HDG methods have shown great promise for the efficient computation of high-order accurate solutions, and the present thesis extends HDG to time-periodic flow problems and multi-row turbomachinery applications for the first time. A major focus of this work is to combine HDG with a time discretization that is well-suited for turbomachinery problems. The following section provides a brief review of time discretization methods for periodic problems.

1.3.2 Time Discretizations for Periodic Flows

Turbomachinery design and flutter analysis are among the many important applications that have motivated development of better algorithms for time-periodic flow problems over the last several decades. After the earliest work on analytical and semi-analytical methods for simple problems, more realistic flows were first solved using *time-marching* methods. For example, Giles [40] applied a time-marching method to an Euler analysis of rotor/wake interaction, and He & Denton [55] applied a four-stage explicit Runge-Kutta scheme to solve viscous unsteady flow through turbomachinery. Implicit time-marching based on Backward Differentiation Formulae (BDF) or Diagonally Implicit Runge-Kutta (DIRK [1]) schemes have also been applied to time-periodic problems in turbomachinery and beyond [83, 84]. Time-marching methods have the advantage that they are applicable to any arbitrary form of time variation, periodic or not, and are thus essential for resolving transient non-periodic flow behavior. However, when the main goal is to compute a periodic flow state (such as around a pitching airfoil), the inability of time-marching methods to exploit periodicity can make them highly inefficient. For instance, time-marching methods require time-accurate resolution of the initial transient behavior of the flow, even when this initial transient is of no interest to the engineer. As a result, several periods of integration are often required before initial transients subside and a repeating periodic flow state can be reached. Furthermore, explicit time-marching methods are restricted to very small timestep sizes by numerical stability constraints. While implicit time-marching methods are not constrained by stability, they are still required to resolve the time variation of the flow. As a result, even implicit time-marching methods commonly require hundreds of timesteps per period, over several periods of time integration.

Seeking more efficient alternatives to time-marching, much work has been done on formulating time discretizations schemes in the frequency domain. Comprehensive reviews of these methods are provided by Ekici & Hall [34], He [53] and Hall *et al.* [48]. The first frequency domain methods were *time-linearized* methods for the potential flow equations [103–105], the Euler equations [47] and the Reynolds-averaged Navier-

Stokes equations [19], mainly applied to cascade flutter calculations. In all of these methods, the unsteady flow equations were linearized about a steady mean flow field, leaving a set of linear, variable-coefficient equations for the unsteady harmonics that could be solved very efficiently. However, this time-linearized approach is only valid for small disturbance flows and cannot be applied to large-amplitude or nonlinear unsteadiness.

The first frequency domain method for larger-amplitude nonlinear unsteadiness was developed by He & Ning [56]. In their *nonlinear harmonic* approach, they decomposed the full unsteady flow problem into a set of equations for the time-average, coupled to another set of first-order accurate equations for harmonic disturbances. Unlike earlier time-linearized methods, harmonics influenced the time-averaged flow equations through deterministic stress terms. Solving these coupled equations simultaneously, He & Ning and other investigators were able to capture nonlinearities in turbomachinery flows quite accurately [16, 54, 106], although the treatment of harmonics was inexact and their initial demonstration was only implemented for one harmonic.

A more general method for large amplitude periodic unsteadiness was proposed by Hall *et al.* [46, 49, 50]. In their *Harmonic Balance* method, the unsteady flow conservation variables are represented by a Fourier series in time with spatially varying coefficients. Time derivatives are evaluated using a spectral operator, and the user can decide how many harmonic modes they wish to resolve. Consequently, this approach allows arbitrarily high accuracy in time for periodic unsteadiness of any amplitude and spectrum. Ekici and Hall [33, 35, 36] and Ekici *et al.* [38] later extended the Harmonic Balance method to model unsteady flows in turbomachinery with multiple excitation frequencies. Several other investigators successfully applied the Harmonic Balance approach to a range of problems, including flow around a cylinder and a pitching airfoil [67, 68], adjoint-based shape optimization in unsteady flow [70, 95], turbomachinery forced response calculations [100], rotor/stator interactions [90], unsteady aerodynamics of helicopter rotors [13, 18, 37, 63, 111, 112], sound propagation and radiation of lined ducts [11]; pulsating synthetic jets in quiescent cross-flows [108]; and nonlinear

flutter and limit cycle oscillation [96–99]. Implicit versions of the Harmonic Balance method have also been developed [32, 88, 91, 95, 107, 110, 111].

An important improvement to the Harmonic Balance method was to formulate this frequency domain approach solely in terms of time-domain quantities, for ease of implementation around existing time-marching solvers. Gopinath & Jameson [42] made this a key feature of their *Time-Spectral* method, in which the Fourier series of the solution is provided not in terms of spatially-varying coefficients, but in terms of discrete "snapshots" equispaced in time. This transformation into the time domain results in a system of equations where all snapshots are coupled to one another through a spectral time derivative operator. This formulation has the advantage that many parts of a conventional time-marching code can be re-used in the implementation of a Time-Spectral solver with the same spatial discretization.

Gopinath & Jameson successfully applied their Time-Spectral method to a range of periodic flow problems, including pitching airfoils and wings [42], vortex shedding [43], and turbomachinery flows with sliding mesh interfaces between blade rows [41, 44]. As with the work by Hall *et al.*, Gopinath & Jameson used low-order Finite Volume spatial discretizations. High-order spatial discretizations had not been used at all for Time-Spectral computations until very recent work by Knapke *et al.* [62] where a Discontinuous Galerkin discretization was combined with the Time-Spectral method for single cascade flows. However, a high-order HDG discretization has several advantages, including fewer globally coupled degrees of freedom than other DG methods. The HDG method's natural framework for boundary conditions also facilitates the formulation of a high-order sliding interface coupling technique, essential for turbomachinery applications.

In all of the time discretizations mentioned so far, it is assumed that the user will specify a set of desired frequencies in advance which define the numerical solution. However, it is not always obvious *a priori* which frequencies are the most important. Towards addressing this issue, one contribution of this thesis is a new adaptive method for automatically selecting frequencies that are likely to be most relevant in a given

time-periodic problem.

1.4 Thesis Overview

The remainder of this thesis is organized as follows. In Chapter 2, details of the temporal and spatial discretizations of the Time-Spectral HDG method are presented, including a combined formulation and demonstration of the method for convection problems and viscous flow over a pitching airfoil. Following this, Chapter 3 describes the formulation of a high-order sliding interface coupling technique for HDG, together with an anti-aliasing filter for use in Time-Spectral HDG simulations. Supporting results are provided from a convection-diffusion problem on a static grid and a viscous flow problem on a sliding grid. Chapter 4 presents applications of the Time-Spectral HDG method to turbomachinery flow problems. This begins with details of regularizations applied to the Spalart-Allmaras (SA) model, designed to assist in the numerical stability of high-order simulations of unsteady turbulent flows. Chapter 4 then presents a series of turbomachinery flow simulations computed using the Time-Spectral HDG method. Flow solutions for a rotor cascade, rotor/stator assemblies and a three-row compressor are presented using different combinations of high and low order in space, time-marching and Time-Spectral. Highlighted by these comparisons, performance characteristics of the Time-Spectral HDG method are discussed. In Chapter 5, a Frequency-Adaptive Time-Spectral method is introduced for periodic problems, and demonstrations are presented for linear and nonlinear ordinary differential equations and a rotor/stator flow problem. Finally, Chapter 6 presents a summary of findings from this thesis and suggests a few salient directions for future work.

Chapter 2

Time-Spectral Hybridizable Discontinuous Galerkin (HDG) Method

This chapter outlines the major features of the Time-Spectral Hybridizable Discontinuous Galerkin (HDG) method for solving time-periodic flows. This method is the first to combine a Time-Spectral discretization in time with an HDG discretization in space. The following sections describe the formulation of the temporal and spatial discretizations of Time-Spectral HDG, and provide a set of results for convection problems and pitching airfoil flows to demonstrate performance characteristics of the method.

2.1 Time-Spectral Method for Ordinary Differential Equations

The present thesis employs a Time-Spectral discretization in time, following earlier work in a Finite Volume setting by Gopinath & Jameson [41, 42]. The Time-Spectral method offers spectral accuracy in time and direct solution of time-periodic flows

without the need to resolve initial transients. When the periodic unsteadiness of a flow can be well represented by a small number of Fourier modes, the Time-Spectral method can achieve an order of magnitude improvement in computational efficiency compared to more conventional time-marching approaches [41].

To describe the Time-Spectral method, let us begin by considering a conservation law written in semi-discrete form:

$$\boldsymbol{M}\frac{du}{dt} + R(u) = 0 \tag{2.1}$$

Here, $u = u(t)$ is the numerical solution vector for a discretized spatial domain, defined for continuous time t. The term $\boldsymbol{M}$ represents a mass matrix, and $R(u)$ is a nonlinear residual vector that is a function of u. (The form of $\boldsymbol{M}$, R and u depends on the chosen spatial discretization scheme.)

The discrete Fourier transform of u for a given time period T is:

$$\hat{u}_k = \frac{1}{N}\sum_{n=0}^{N-1} u^n e^{-ikn\frac{2\pi}{T}\Delta t} \tag{2.2}$$

and its inverse transform is:

$$u^n = \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} \hat{u}_k e^{ikn\frac{2\pi}{T}\Delta t} \tag{2.3}$$

Here, $u^n = u(t_n)$ are N discrete *snapshots* in time of the solution u, equispaced by intervals $\Delta t = T/N$ over the period T. These N snapshots describe $K = (N-1)/2$ harmonic frequencies, which we will refer to in this thesis as *modes*. (For example, $N = 3$ snapshots describe exactly 1 mode – or, a constant value and a single harmonic with an amplitude and phase. Together these are 3 variables that can be mapped from the $N = 3$ snapshots.) The quantity $\hat{u}_k$ is the complex coefficient of the k^{th} Fourier basis function. The amplitude and phase of the k^{th} mode (or harmonic frequency, as we have defined it) is related to both $\hat{u}_k$ and $\hat{u}_{-k}$, which are in turn related to each

other by $\hat{u}_k = -\hat{u}_{-k}$ for real-valued u^n.

Now, by differentiating Equation 2.3, the discrete time derivative operator can be expressed in terms of frequency domain quantities as:

$$D_t u^n = \frac{2\pi}{T} \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} ik\hat{u}_k e^{ikn\frac{2\pi}{T}\Delta t} \tag{2.4}$$

Substituting the discrete Fourier transform (Equation 2.2) for $\hat{u}_k$, we obtain an expression for the spectral time-derivative of u exclusively in terms of time-domain quantities:

$$\begin{aligned} D_t u^n &= \frac{2\pi}{T} \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} ik \left(\frac{1}{N} \sum_{j=0}^{N-1} u^j e^{-ikj\frac{2\pi}{T}\Delta t} \right) e^{ikn\frac{2\pi}{T}\Delta t} \\ &= \sum_{j=0}^{N-1} \left(\frac{1}{N}\frac{2\pi}{T} \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} ike^{-ikj\frac{2\pi}{T}\Delta t} e^{ikn\frac{2\pi}{T}\Delta t} \right) u^j \\ &= \sum_{j=0}^{N-1} d_n^j u^j \end{aligned} \tag{2.5}$$

where d_n^j are constant coefficients which couple all the snapshots in time, u^j. For odd N, the form of these coefficients can be simplified using trigonometric identities [41, 43]:

$$d_n^j = \begin{cases} \frac{\pi}{T}(-1)^{n-j}\text{cosec}\left(\frac{\pi(n-j)}{N}\right), & \text{if } n \neq j \\ 0, & \text{if } n = j. \end{cases} \tag{2.6}$$

Together, Equations 2.5 & 2.6 define a Time-Spectral discretization of the time derivative $\partial u/\partial t$, which can be applied to the semi-discrete form of the governing equation for any chosen spatial discretization (2.1). The semi-discrete form will become:

$$\underbrace{\begin{pmatrix} d_0^0 & \cdots & d_0^{N-1} \\ d_1^0 & \cdots & d_1^{N-1} \\ \vdots & \ddots & \vdots \\ d_{N-1}^0 & \cdots & d_{N-1}^{N-1} \end{pmatrix}}_{\text{Matrix } \boldsymbol{D}} \begin{pmatrix} Mu_0 \\ Mu_1 \\ \vdots \\ Mu_{N-1} \end{pmatrix} + \begin{pmatrix} R(u_0) \\ R(u_1) \\ \vdots \\ R(u_{N-1}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{2.7}$$

or, more compactly:

$$(\boldsymbol{D} \otimes M)\boldsymbol{u} + \boldsymbol{R}(\boldsymbol{u}) = 0 \tag{2.8}$$

Here $\boldsymbol{u} = [u_0, \ldots, u_{N-1}]^T$ is the vector of N solution snapshots equispaced over the period $[0, T]$, $\boldsymbol{R}(\boldsymbol{u}) = [R(u_0), \ldots, R(u_{N-1})]$ is the vector of N residuals, and $\boldsymbol{D}$ is the $N \times N$ Time-Spectral coupling matrix that acts as a spectral time-derivative operator. Note that this matrix $\boldsymbol{D}$ is dense and skew-symmetric (i.e. $d_i^j = -d_j^i$ and $d_i^i = 0$). The Time-Spectral approach therefore couples all snapshots of the solution, requiring that the entire periodic flow solution be solved simultaneously. Once this system has been solved, the N snapshots can be interpolated in a Fourier sense (using FFT) to obtain the solution at any time in the period $[0, T]$.

A convenient feature of the Time-Spectral formulation is that all quantities to be computed (residuals and unknowns) are computed in the time-domain at discrete snapshots. This means that many subroutines from a conventional time-marching code can be reused in the implementation of a Time-Spectral solver, as opposed to other methods that are formulated in the frequency domain.

2.2 Hybridizable Discontinuous Galerkin (HDG) Finite Element Method

For spatial discretization, the present thesis employs a high-order Hybridizable Discontinuous Galerkin (HDG) method, in contrast to the low-order Finite Volume schemes seen in earlier work on Time-Spectral methods [42, 50]. HDG methods offer high-order accuracy and low numerical dissipation with a reduced number of globally-coupled

degrees of freedom compared to other DG methods, leading to substantial savings in computation time and memory usage [75]. In this section we provide a brief review of the formulation of the HDG method, summarizing the approach presented by Nguyen, Peraire & Cockburn [72, 73, 75].

To illustrate the method, we will consider a time-dependent convection-diffusion equation as a model problem. This can be written as a system of first-order equations:

$$\begin{aligned} \boldsymbol{q} + \kappa \nabla u &= 0, && \text{in } \Omega \times (0,T], \\ \frac{\partial u}{\partial t} + \nabla \cdot (\boldsymbol{c}u + \boldsymbol{q}) &= f, && \text{in } \Omega \times (0,T], \\ u &= g_D, && \text{on } \Gamma_D \times (0,T], \\ (\boldsymbol{q} + \boldsymbol{c}u) \cdot \boldsymbol{n} &= g_N, && \text{on } \Gamma_N \times (0,T], \\ u &= u_0, && \text{in } \Omega \text{ for } t = 0. \end{aligned} \tag{2.9}$$

Here, u is the scalar solution variable, $\boldsymbol{q}$ is its gradient, $\Omega \in \mathbb{R}^d$ is the physical domain with boundary $\partial\Omega$, $f \in L^2(\Omega)$ is a source term, $\kappa \in L^\infty(\Omega)$ is a positive diffusivity coefficient, $\boldsymbol{c} \in (L^\infty(\Omega))^d$ is a velocity vector field, g_D is the Dirichlet boundary condition, g_N is the Neumann boundary condition, and u_0 is the initial condition for u. Also, $L^2(D)$ is the space of square integrable functions on a domain D.

Let $\mathcal{T}_h$ be a collection of disjoint elements that partition Ω. We denote by $\partial\mathcal{T}_h$ the set $\{\partial K : K \in \mathcal{T}_h\}$. For an element K of the collection $\mathcal{T}_h$, $e = \partial K \cap \partial\Omega$ is a boundary face if the $d-1$ Lebesgue measure of e is nonzero. For two elements K^+ and K^- of the collection $\mathcal{T}_h$, $e = \partial K^+ \cap \partial K^-$ is the interior face between K^+ and K^- if the $d-1$ Lebesgue measure of e is nonzero. Let $\mathcal{E}_h^o$ and $\mathcal{E}_h^\partial$ denote the set of interior and boundary faces, respectively. We denote by $\mathcal{E}_h$ the union of $\mathcal{E}_h^o$ and $\mathcal{E}_h^\partial$.

Let $\mathcal{P}^p(D)$ denote the set of polynomials of degree at most p on a domain D. For any element K of the collection $\mathcal{T}_h$ we denote $W^p(K) \equiv \mathcal{P}^p(K)$ and $\boldsymbol{V}^p(K) \equiv (\mathcal{P}^p(K))^d$. We can now introduce discontinuous finite element spaces:

$$W_h^p = \{w \in L^2(\Omega) \ : \ w|_K \in W^p(K) \ \forall K \in \mathcal{T}_h\},$$
$$\boldsymbol{V}_h^p = \{\boldsymbol{v} \in (L^2(\Omega))^d \ : \ \boldsymbol{v}|_K \in \boldsymbol{V}^p(K) \ \forall K \in \mathcal{T}_h\}.$$

In addition, we introduce a traced finite element space of special importance to HDG:

$$M_h^p = \{\mu \in L^2(\mathcal{E}_h) \ : \ \mu|_e \in \mathcal{P}^p(e), \forall e \in \mathcal{E}_h\}.$$

We also set $M_h^p(g_D) = \{\mu \in M_h^p \ : \ \mu = \mathsf{P}g_D \text{ on } \Gamma_D\}$, where P denotes the L^2-projection into the space $\{\mu|_{\partial\Omega} \, \forall \mu \in M_h^p\}$. Note that M_h^p consists of functions which are continuous inside the faces (or edges) $e \in \mathcal{E}_h$ and discontinuous at their borders.

A few more definitions of notation will aid our description of the HDG method. For functions $\boldsymbol{w}$ and $\boldsymbol{v}$ in $(L^2(D))^d$, we denote $(\boldsymbol{w}, \boldsymbol{v})_D = \int_D \boldsymbol{w} \cdot \boldsymbol{v}$. For functions u and v in $L^2(D)$, we denote $(u, v)_D = \int_D uv$ if D is a domain in $\mathbb{R}^d$ and $\langle u, v \rangle_D = \int_D uv$ if D is a domain in $\mathbb{R}^{d-1}$. We finally introduce

$$(w, v)_{\mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} (w, v)_K, \quad \langle \zeta, \rho \rangle_{\partial\mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} \langle w, v \rangle_{\partial K}, \quad \langle \mu, \eta \rangle_{\mathcal{E}_h} = \sum_{e \in \mathcal{E}_h} \langle \mu, \eta \rangle_e,$$

for functions w, v defined on $\mathcal{T}_h$, ζ, ρ defined on $\partial\mathcal{T}_h$, and μ, η defined on $\mathcal{E}_h$.

Now that we have introduced all the function spaces and notation, we can present the HDG form of the governing equations (Equation 3.8). For the sake of illustration, we will discretize the time derivative using a Backward-Euler scheme. The HDG method then seeks an approximation $(\boldsymbol{q}_h^k, u_h^k, \lambda_h^k) \in \boldsymbol{V}_h^p \times W_h^p \times M_h^p(g_D)$ such that:

$$\begin{aligned} (\kappa^{-1}\boldsymbol{q}_h^k, \boldsymbol{v})_{\mathcal{T}_h} - (u_h^k, \nabla \cdot \boldsymbol{v})_{\mathcal{T}_h} + \left\langle \lambda_h^k, \boldsymbol{v} \cdot \boldsymbol{n} \right\rangle_{\partial\mathcal{T}_h} &= 0, \\ \frac{1}{\Delta t^k}\left(u_h^k, w\right)_{\mathcal{T}_h} - (\boldsymbol{c}u_h^k + \boldsymbol{q}_h^k, \nabla w)_{\mathcal{T}_h} + \left\langle (\widehat{\boldsymbol{c}u}_h^k + \widehat{\boldsymbol{q}}_h^k) \cdot \boldsymbol{n}, w \right\rangle_{\partial\mathcal{T}_h} &= (f, w)_{\mathcal{T}_h} + \frac{1}{\Delta t^k}\left(u_h^{k-1}, w\right)_{\mathcal{T}_h}, \\ \left\langle (\widehat{\boldsymbol{c}u}_h^k + \widehat{\boldsymbol{q}}_h^k) \cdot \boldsymbol{n}, \mu \right\rangle_{\partial\mathcal{T}_h} &= \langle g_N, \mu \rangle_{\Gamma_N}, \end{aligned} \tag{2.10}$$

for all $(\boldsymbol{v}, w, \mu) \in \boldsymbol{V}_h^p \times W_h^p \times M_h^p(0)$, where the *numerical flux* is defined by:

$$\widehat{\boldsymbol{c}u}_h^k + \widehat{\boldsymbol{q}}_h^k = \boldsymbol{c}\,\widehat{u}_h^k + \boldsymbol{q}_h^k + \tau(u_h^k - \lambda_h^k)\boldsymbol{n}, \quad \text{on } \partial K.$$

Here we have denoted $u_h^k = u_h(t^k)$ and $\boldsymbol{q}_h^k = \boldsymbol{q}_h(t^k)$, and u_h^0 is the L^2 projection of u_0 into W_h^p. Note that τ is the so-called stabilization parameter, chosen as $\tau = \|\boldsymbol{c}\cdot\boldsymbol{n}\| + \kappa/\ell$

for some characteristic length scale ℓ.

While this example has used a Backward-Euler scheme, many other discretization methods can be applied to approximate the time derivative term. For example, higher-order Backward Differentiation Formula (BDF) methods, implicit Runge-Kutta methods, or the Time-Spectral method that is employed in the present thesis. We also note that while we have illustrated the HDG formulation for a linear convection-diffusion problem, extension of the method to systems of nonlinear conservation laws is rather straightforward, and thus omitted here. Applications of the HDG method can be found in prior work on nonlinear convection-diffusion problems [73], the compressible Euler and Navier-Stokes equations [81], and the Reynolds-Averaged Navier-Stokes equations [69].

2.3 Combined Formulation of a Time-Spectral HDG Method

We now describe the combination of a Time-Spectral discretization in time (Section 2.1) with an HDG discretization in space (Section 2.2), forming the Time-Spectral HDG method which is the main subject of this thesis. Consider a general scalar conservation law for a time-periodic problem:

$$\frac{\partial u}{\partial t} + \nabla \cdot F(u, -\nabla u) = f, \quad \text{in } \Omega \times [0, T) \tag{2.11}$$

where u is the primal solution variable, $F = F(u, -\nabla u)$ are the nonlinear fluxes, $f \in L^2(\Omega)$ is a source term, $T \in \mathbb{R}$ is the fundamental time period, and $\Omega \in \mathbb{R}^d$ is the spatial domain.

The corresponding system of first-order equations will be:

$$\begin{aligned} \vec{q} + \nabla u &= 0, \quad && \text{in } \Omega \times (0, T], \\ \frac{\partial u}{\partial t} + \nabla \cdot F(u, \vec{q}) &= f, \quad && \text{in } \Omega \times (0, T], \end{aligned} \tag{2.12}$$

and we can apply the Time-Spectral method to approximate the time-derivative term:

$$\begin{aligned} \boldsymbol{q} + \nabla \boldsymbol{u} &= 0, \qquad \text{in } \Omega, \\ \boldsymbol{D}\boldsymbol{u} + \nabla \cdot \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{q}) &= \boldsymbol{f}, \qquad \text{in } \Omega, \end{aligned} \tag{2.13}$$

where the vector $\boldsymbol{u} = [u_0, \ldots, u_{N-1}]^T$ contains N equispaced snapshots of the solution over the period $[0, T]$, $\boldsymbol{q} = [\vec{q}_0, \ldots, \vec{q}_{N-1}]^T$, and $\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{q}) = [F(u_0, \vec{q}_0), \ldots, F(u_{N-1}, \vec{q}_{N-1})]^T$. The operator $\boldsymbol{D}$ is the Time-Spectral coupling matrix of constants d_n^j (Equations 2.6 & 2.7).

The Time-Spectral HDG method involves extended discontinuous finite element spaces:

$$\begin{aligned} \boldsymbol{W}_h^p &= \{\boldsymbol{w} \in (L^2(\Omega))^N \ : \ \boldsymbol{w}|_K \in (W^p(K))^N \ \forall K \in \mathcal{T}_h\}, \\ \boldsymbol{Q}_h^p &= \{\boldsymbol{v} \in (L^2(\Omega))^{N \times d} \ : \ \boldsymbol{v}|_K \in (\boldsymbol{V}^p(K))^N \ \forall K \in \mathcal{T}_h\}, \\ \boldsymbol{M}_h^p &= \{\boldsymbol{\mu} \in (L^2(\mathcal{E}_h))^N \ : \ \boldsymbol{\mu}|_e \in (\mathcal{P}^p(e))^N, \forall e \in \mathcal{E}_h\}. \end{aligned}$$

and we also set $\boldsymbol{M}_h^p(\boldsymbol{g}_D) = \{\boldsymbol{\mu} \in \boldsymbol{M}_h^p : \boldsymbol{\mu} = \mathsf{P}\boldsymbol{g}_D \text{ on } \Gamma_D\}$.

The Time-Spectral HDG method then seeks an approximation $(\boldsymbol{q}_h, \boldsymbol{u}_h, \widehat{\boldsymbol{u}}_h) \in \boldsymbol{Q}_h^p \times \boldsymbol{W}_h^p \times \boldsymbol{M}_h^p(\boldsymbol{g}_D)$ such that

$$\begin{aligned} (\boldsymbol{q}_h, \boldsymbol{v})_{\mathcal{T}_h} - (\boldsymbol{u}_h, \nabla \cdot \boldsymbol{v})_{\mathcal{T}_h} + \langle \widehat{\boldsymbol{u}}_h, \boldsymbol{v} \cdot \boldsymbol{n} \rangle_{\partial \mathcal{T}_h} &= 0, \\ (\boldsymbol{D}\boldsymbol{u}_h, \boldsymbol{w})_{\mathcal{T}_h} - (\boldsymbol{F}(\boldsymbol{u}_h, \boldsymbol{q}_h), \nabla \boldsymbol{w})_{\mathcal{T}_h} + \left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_h, \boldsymbol{q}_h, \widehat{\boldsymbol{u}}_h) \cdot \boldsymbol{n}, \boldsymbol{w} \right\rangle_{\partial \mathcal{T}_h} &= (\boldsymbol{f}, \boldsymbol{w})_{\mathcal{T}_h} \\ \left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_h, \boldsymbol{q}_h, \widehat{\boldsymbol{u}}_h) \cdot \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h \setminus \partial \Omega} + \left\langle \widehat{\boldsymbol{B}}_h, \mu \right\rangle_{\partial \Omega} &= 0 \end{aligned} \tag{2.14}$$

for all $(\boldsymbol{v}, \boldsymbol{w}, \boldsymbol{\mu}) \in \boldsymbol{Q}_h^p \times \boldsymbol{W}_h^p \times \boldsymbol{M}_h^p(0)$, where the numerical flux is defined:

$$\widehat{\boldsymbol{F}}(\boldsymbol{u}_h, \boldsymbol{q}_h, \widehat{\boldsymbol{u}}_h) \cdot \boldsymbol{n} = \boldsymbol{F}(\boldsymbol{u}_h, \boldsymbol{q}_h) \cdot \boldsymbol{n} + \tau(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \quad \text{on } \partial K. \tag{2.15}$$

and the boundary flux term $\widehat{\boldsymbol{B}}_h$ is defined on the domain boundaries, depending on the particular boundary conditions employed (see [81]).

Equations 2.14 and 2.15 define the nonlinear system of equations for the Time-

Spectral HDG solution of the governing conservation law, on the spatial domain and time period specified. Solving these equations involves a Newton-Raphson procedure, essentially the same as the procedure used in the HDG solution of a steady nonlinear problem [73, 81]. To briefly summarize, after linearizing the system of equations, one will obtain a matrix system of the form:

$$\begin{bmatrix} A & B & E \\ C & D & L \\ M & N & P \end{bmatrix} \begin{bmatrix} \delta Q \\ \delta U \\ \delta \widehat{U} \end{bmatrix} = \begin{bmatrix} H \\ F \\ G \end{bmatrix} \tag{2.16}$$

where $(\delta Q, \delta U, \delta \widehat{U})$ are the Newton update vectors associated with the Time-Spectral HDG solution quantities $(\boldsymbol{q}_h, \boldsymbol{u}_h, \widehat{\boldsymbol{u}}_h)$, respectively. As in previous applications of the HDG method, we can now exploit the block diagonal structure of the block $[A\ B;\ C\ D]$ and apply static condensation to form a much smaller system of equations solely for the global degrees of freedom $\widehat{\boldsymbol{u}}_h$:

$$\mathbb{K}\, \delta \widehat{U} = \mathbb{F} \tag{2.17}$$

where the form of $\mathbb{K}$ and $\mathbb{F}$ are as presented in [81]. Finally, after solving the global system of equations (Eq. 2.17), the local quantities $\boldsymbol{u}_h$ and $\boldsymbol{q}_h$ can be obtained efficiently in a parallel fashion over all elements.

An important feature of the HDG method is that the global matrix $\mathbb{K}$ is sparse and block-structured, with each dense block corresponding to the trace degrees of freedom on a particular face of the spatial mesh. For a Time-Spectral HDG discretization with spatial polynomial order p, number of conserved quantities m, total number of faces n_f, and number of snapshots N, each matrix block will be of size $(p+1)mN \times (p+1)mN$, and the overall matrix $\mathbb{K}$ will be of size $(p+1)mNn_f \times (p+1)mNn_f$. Block row i of the matrix will contain dense blocks for face i and for each of the faces that are *adjacent* to face i. Adjacency is defined as follows: a face is adjacent to an element K if it is on the boundary ∂K of element K; and two faces are adjacent to each other if they

are adjacent to the same element [87]. For example, in a two-dimensional triangular mesh, block-row i of the matrix will contain up to 5 dense blocks, corresponding to the face i and up to 4 adjacent faces (exactly 4 if face i is not a boundary face). This sparsity pattern is illustrated schematically in Figure 2-1. Note that generalizing this illustration to $N > 1$ snapshots will increase the size of each matrix block by a factor N, but will not change the overall block structure shown. Also, the global matrix $\mathbb{K}$ for a mesh with a larger number of elements will be much more sparse. For a two-dimensional triangular mesh, no block-row will contain more than 5 dense blocks, but the overall matrix will have a size corresponding to $n_f \times n_f$ blocks, where n_f is the total number of faces in the mesh.

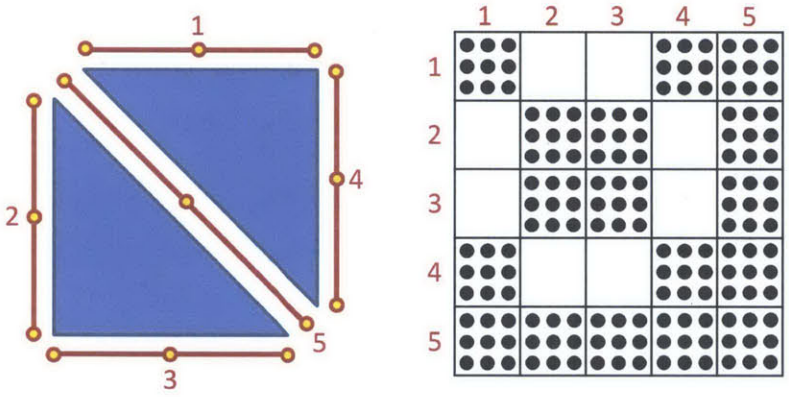

Figure 2-1: Schematic illustration of the sparsity pattern of HDG global matrix $\mathbb{K}$, for a scalar problem with $N = 1$ snapshot on a simple $p = 2$ mesh (*left*) with 2 elements and 5 faces. Trace degrees of freedom are defined at the nodes depicted on each face.

Note that the Time-Spectral HDG method with N snapshots will yield a linear system which is N times larger than the linear system of the BDF-HDG method. However, while the BDF-HDG linear system would need to be solved repeatedly for a large number of timesteps (N_{BDF}), we only need to solve the Time-Spectral HDG system once to obtain the numerical solution over the whole time span $[0, T]$. In many cases the periodic unsteadiness of the flow will be spectrally simple enough to allow $N \ll N_{BDF}$, and it is these cases where the Time Spectral HDG method will have its greatest advantage.

2.4 Solution Techniques for Time-Spectral HDG Equations

The Time-Spectral HDG method yields a system of steady nonlinear equations that must be solved numerically. The simplest approach is to choose an initial guess for the solution at all N snapshots, and then apply a Newton-Raphson method to iteratively solve the nonlinear problem. In practice, complications can arise when the initial guess is too far from the numerical solution and Newton iteration fails to converge. In such cases, there are two strategies that can aid convergence towards a solution.

The first strategy is to use time-marching in a pseudo-time direction in order to more gradually approach a solution. This kind of relaxation is analogous to pseudo-time marching strategies often used to solve steady flow problems, and has even been used for solving flow equations in the frequency domain (e.g. [79]). In the present context, we begin with the Time-Spectral HDG equations (Equation 2.14) and add a new time-derivative term $\partial \boldsymbol{u}_h / \partial \tilde{t}$ in a "pseudo-time" direction $\tilde{t}$:

$$\begin{aligned} (\boldsymbol{q}_h, \boldsymbol{v})_{\mathcal{T}_h} - (\boldsymbol{u}_h, \nabla \cdot \boldsymbol{v})_{\mathcal{T}_h} + \langle \widehat{\boldsymbol{u}}_h, \boldsymbol{v} \cdot \boldsymbol{n} \rangle_{\partial \mathcal{T}_h} &= 0, \\ \left(\frac{\partial \boldsymbol{u}_h}{\partial \tilde{t}} + \boldsymbol{D}\boldsymbol{u}_h, \boldsymbol{w} \right)_{\mathcal{T}_h} - (\boldsymbol{F}(\boldsymbol{u}_h, \boldsymbol{q}_h), \nabla \boldsymbol{w})_{\mathcal{T}_h} + \left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_h, \boldsymbol{q}_h, \widehat{\boldsymbol{u}}_h) \cdot \boldsymbol{n}, \boldsymbol{w} \right\rangle_{\partial \mathcal{T}_h} &= (\boldsymbol{f}, \boldsymbol{w})_{\mathcal{T}_h} \\ \left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_h, \boldsymbol{q}_h, \widehat{\boldsymbol{u}}_h) \cdot \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h \setminus \partial \Omega} + \left\langle \widehat{\boldsymbol{B}}_h, \mu \right\rangle_{\partial \Omega} &= 0 \end{aligned} \tag{2.18}$$

An initial condition at $\tilde{t} = 0$ may be specified as the steady-state solution for the problem, assigned to all N snapshots. Integrating the equations forward in pseudo-time, one will ultimately reach a state where $\partial \boldsymbol{u}_h / \partial \tilde{t} \sim 0$, and this will be the numerical solution to the original problem. Alternatively, one can take a limited number of steps forward in pseudo-time, and use the resulting solution as an improved initial guess for Newton iteration of the original problem. Note that the pseudo-time variation of the solution is not physical and does not need to be accurately resolved – the only goal here is to reach a state where $\partial \boldsymbol{u}_h / \partial \tilde{t} \sim 0$, and to do so as rapidly as possible. For this purpose, highly stable implicit time-marching schemes such as Backward Euler

are a very appropriate choice, and may be used with rapidly growing pseudo-timestep sizes.

A second strategy is to employ *continuation* in the number of snapshots N, and this may be a more attractive option in many cases. Beginning with an initial guess (such as the steady-state solution), a series of Time-Spectral HDG problems may be solved with a successively larger number of snapshots. At each stage, the solution for the previous N is interpolated in time onto the new set of N' snapshots as an initial guess for Newton iteration. An advantage of this approach is that the problem solved at each stage is substantially smaller than the final version of the problem, since problem size scales with N. By contrast, in the pseudo-time-marching approach each step is as expensive as the final problem. Another advantage of the N-continuation approach is that the solution at each stage is a valid periodic solution of the problem, limited to a subset of the total number of frequencies that appear in a converged solution. If a user's output of interest pertains to a phenomenon occurring at a low frequency, such an output may converge with N more rapidly than the overall solution, allowing the user to save a potentially large amount of computational effort by avoiding larger N. In fact, in Chapter 4 of this thesis we will see examples where such early convergence occurs for blade loading quantities in rotor/stator flow simulations.

In this thesis, the N-continuation strategy is employed for all Time-Spectral HDG simulations, and it has been found sufficiently robust for the range of flow problems examined herein.

2.5 Demonstrations of the Time-Spectral HDG method

This section serves to demonstrate the convergence properties of the Time-Spectral HDG method, through two examples. First, a linear convection problem on a spatially periodic domain, and then viscous compressible flow over a pitching airfoil. These results are as reported by Chaurasia *et al.* 2013 [14].

2.5.1 Convection on a Periodic Domain

Governing Equation

For our first demonstration of the Time-Spectral HDG method, we solve a time-periodic convection problem in one and two dimensions. A prescribed initial condition will be convected through a domain with spatially periodic boundary conditions, such that the solution wraps around and returns to where it began, making the exact solution of this problem periodic in time. The governing equation can be written:

$$\begin{aligned} \frac{\partial u}{\partial t} + \nabla \cdot (\boldsymbol{c}u) &= 0 \qquad && \text{in } \Omega \times (0,T] \\ u &= 0, && \text{on } \Gamma_D \times (0,T] \\ u &= u_0, && \text{in } \Omega \text{ for } t = 0 \end{aligned} \tag{2.19}$$

where $\Omega \in \mathbb{R}^d$ is the spatial domain, $T \in \mathbb{R}$ is the fundamental period, $\boldsymbol{c} \in (L^\infty(\Omega))^d$ is the convection velocity field, Γ_D is a Dirichlet boundary (for the 2D case), and u_0 is the initial condition of the solution u. In the one-dimensional case, we choose a convection velocity of $c = 1$ and a unit domain $\Omega = [0,1]$ with periodic boundary conditions on the left and right, such that the fundamental period of the solution is $T = 1$. The initial condition is chosen to be a Gaussian $u(x,0) = \exp[-200(x-0.5)^2)]$. In the two-dimensional case, we choose a convection velocity of $\boldsymbol{c} = (1,0)$ and a unit square domain $\Omega = [0,1] \times [0,1]$ with periodic boundary conditions on the left and right, such that the fundamental period of the solution is $T = 1$. For the initial condition we choose a Gaussian function $u(\mathbf{x},0) = \exp[-200((x-0.5)^2+(y-0.5)^2))]$. Homogeneous Dirichlet boundary conditions ($u = 0$) are imposed on the top and bottom boundaries of the square domain (Γ_D), and spatially periodic boundary conditions are imposed on the left and right boundaries.

Note that in this time-periodic convection problem, there is a prescribed initial condition ($t = 0$ solution in the periodic cycle). This contrasts to problems such as pitching airfoil flows, where the $t = 0$ solution is not known *a priori*. In cases where an initial condition is known and required to be prescribed, it is necessary to eliminate

the $t = 0$ solution snapshot from the Time-Spectral HDG linear system, reducing by one the number of snapshots to be solved and resulting in an augmented source term.

1D Results

We first present results from the solution of a one-dimensional version of the time-periodic convection problem described above. High order elements with $p = 4$ polynomials were used. Figure 2-2 shows that we attain the expected exponential convergence of our Time-Spectral method, as measured in the space-time L^2-norm of solution error relative to the known exact solution. Further, this result demonstrates the very small number of snapshots required to fully resolve the solution in time, with only $N = 15$ snapshots (7 Fourier modes) required to fully resolve the solution on the finest grid (20 $p = 4$ high-order elements). Note the behavior of the space-time solution error as the grid is refined – this shows that the total accuracy of the solution is limited by the accuracy of the spatial discretization. With the high-order HDG method, grid refinement increases solution accuracy more efficiently than for low-order methods.

2D Results

We next present results from the application of our Time-Spectral HDG method to a time-periodic convection problem in two dimensions (as defined in Section 2.5.1). Figure 2-3 shows a few representative snapshots of the time-periodic solution, as computed using implicit time-marching (Backward Euler with 200 timesteps per period) and the Time-Spectral HDG method (with 21 snapshots, or 10 Fourier modes), both on the same spatial grid (128 $p = 3$ high-order elements). These plots visually illustrate the strong numerical dissipation that arises from using low-order implicit time-marching methods without sufficiently small timestep sizes, in contrast to the very high solution accuracy that can be obtained with a much smaller number of Fourier modes by a Time-Spectral method. In this example, 200 timesteps of Backward Euler produced a first-period solution that is visibly much poorer than a Time-Spectral solution with only 21 snapshots. While most practical time-marching codes would employ higher-

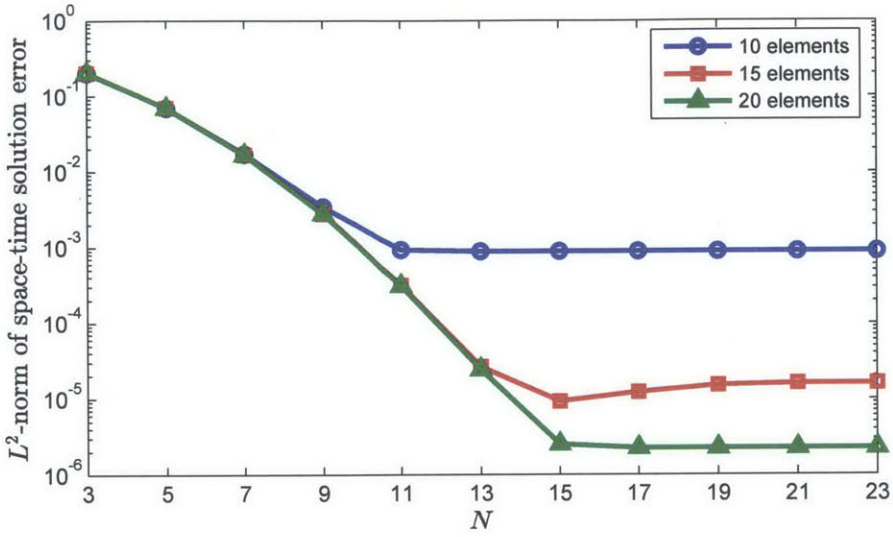

Figure 2-2: Convergence of the Time-Spectral HDG method for a 1D time-periodic convection problem, measured by the L^2-norm of solution error in both space and time.

order time-marching schemes than Backward Euler, this example simply serves to illustrate the point that a Time-Spectral solution is more naturally suited to solving time-periodic problems.

Figure 2-4 quantifies the convergence properties of our Time-Spectral HDG method in this 2D convection application, showing that we attain the expected exponential convergence in the space-time L^2 error norm. Here we also show a comparison of the convergence behavior for high-order spatial grids with the same number of elements but different polynomial orders p. Increasing p offers a convenient and efficient way to decrease solution error. Here we see an illustration of both the spatial accuracy advantages of the high-order HDG method, and the temporal accuracy attainable with the Time-Spectral method.

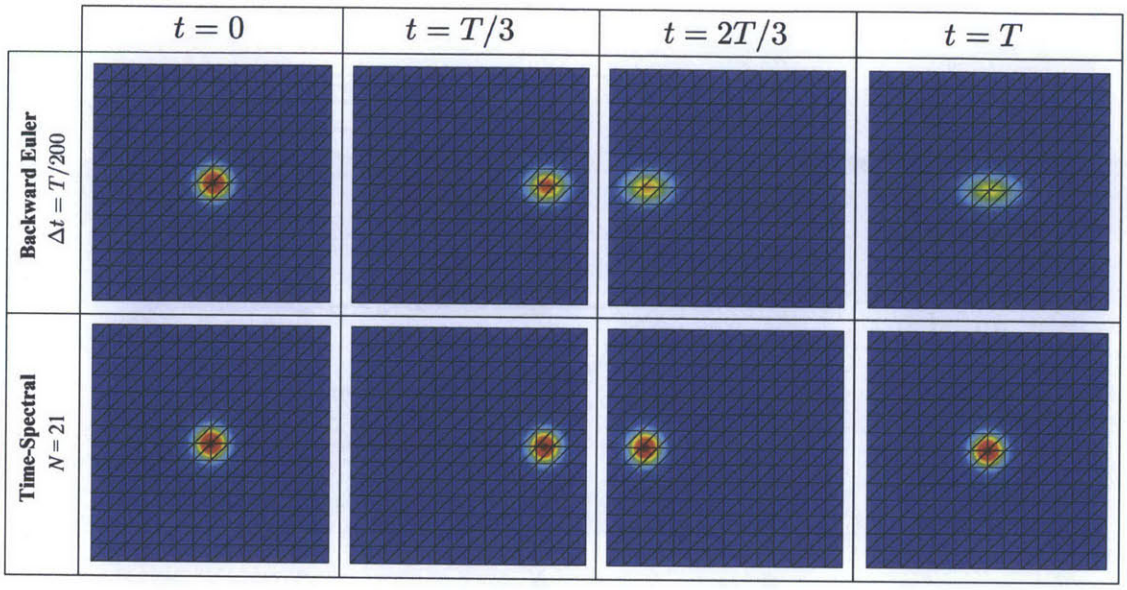

Figure 2-3: Solution snapshots for a 2D time-periodic convection problem, contrasting the behavior of implicit Backward Euler time-marching with 200 timesteps per period (*upper plots*) against the present work's Time-Spectral method with only $N = 21$ snapshots (10 harmonic modes) (*lower plots*).

2.5.2 Viscous Flow over a Pitching Airfoil

Here we demonstrate the performance of the Time-Spectral HDG method in a nonlinear setting by solving the periodic flow over a pitching airfoil at $Re = 1000$.

Problem Description

The airfoil has a symmetric NACA 0012 profile, and moves with an oscillatory vertical translation and angle of attack defined by (in nondimensional terms):

$$y(t) = A\cos(2\pi t/T), \qquad \alpha(t) = B\sin(2\pi t/T). \tag{2.20}$$

Parameters chosen for this example are: period $T = 5$, heaving amplitude $A = 0.125$, and pitch amplitude $B = 5°$. These parameters correspond to a Strouhal number of $St = 2fA/U = 0.05$, where $f = 1/T$ is the oscillation frequency, and $U = 1$ is the freestream velocity. The Reynolds number of the flow is $Re = 1000$ and the in-

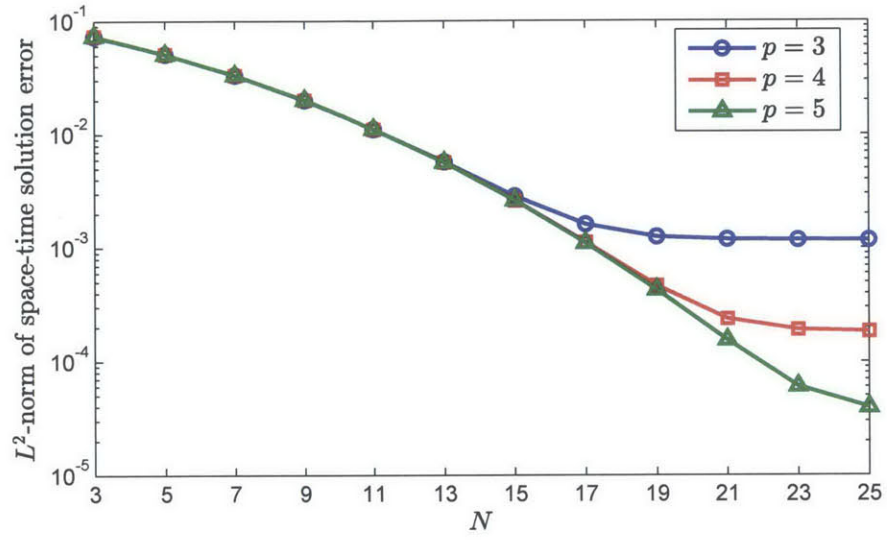

Figure 2-4: Convergence of the Time-Spectral HDG method for 2D time-periodic convection problem, measured by the L^2-norm of solution error in both space and time.

flow Mach number is $M_\infty = 0.2$. The governing equations for this problem are the laminar compressible Navier-Stokes equations, incorporating an Arbitrary Lagrangian-Eulerian (ALE)[83] formulation to account for mesh motion. The computational domain is discretized by a high-order C-mesh with 560 elements, shown in Figure 2-5.

Computational Results

Flow around a pitching airfoil was solved using the Time-Spectral HDG method with several different numbers of snapshots N, and on meshes with three different spatial orders ($p = 3, 4, 5$). To assess the accuracy of the Time-Spectral HDG method in time, solutions were also obtained using an HDG method with a Diagonally Implicit Runge-Kutta (DIRK) time-marching scheme [1, 81]. Time-marched solutions obtained by a 3-stage, 3^{rd}-order DIRK scheme with a very small timestep size ($\Delta t = T/200$) were used as "truth" solutions for comparison with the Time-Spectral results.

Several interesting observations follow from the results of these computations. First

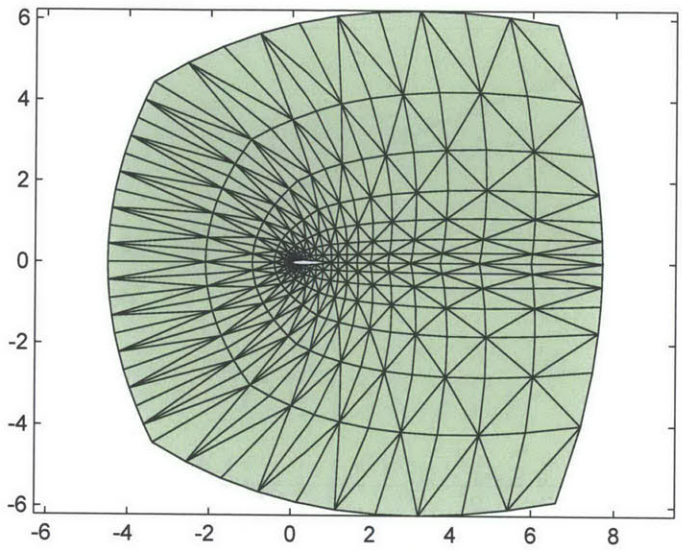

Figure 2-5: High-order mesh for pitching airfoil problem.

of all, Figure 2-6 illustrates the effects of initial transient flow behavior on time-marched and Time-Spectral solutions. Shown here through the lift coefficient time-series, a fully resolved DIRK(3,3) time-marched solution undergoes an initial transient flow behavior that takes at least 3 full oscillation periods to subside. This behavior is not due to numerical inaccuracy – rather, it is the physical behavior of the flow following the necessarily imperfect initial condition prescribed for the flow at the beginning of time-marching. The result is that 3 full periods of time-integration (300 or 600 timesteps in the example shown) are required before the computed flow can reach a repeating periodic state. In contrast, the fully resolved Time-Spectral solution shown in this figure completely avoids the initial transient behavior of the flow. The Time-Spectral method solves the entire periodic flow state simultaneously. The ability to avoid the cost of resolving undesired initial transient behavior is a key advantage of the Time-Spectral method.

The next observation is that the Time-Spectral HDG solution converges very rapidly to a fully time-resolved solution. For example, Figure 2-7 shows the airfoil

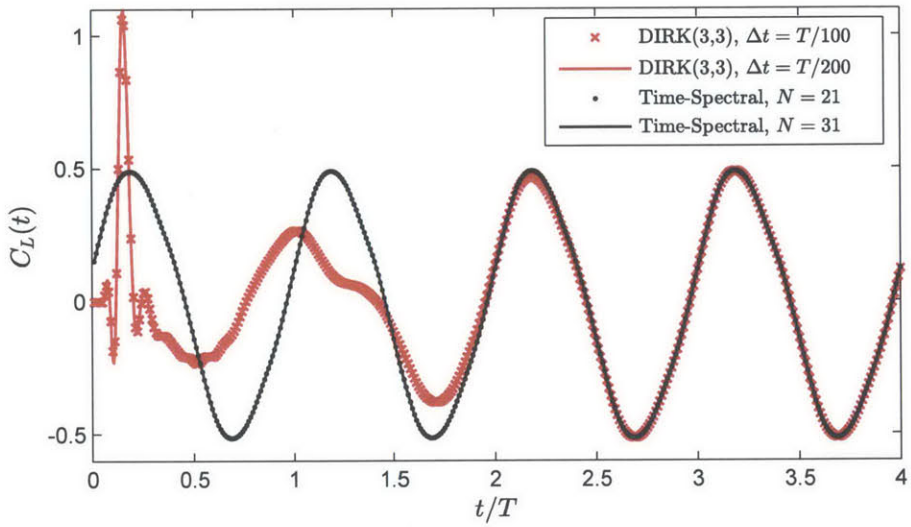

Figure 2-6: The effect of undesired initial transient behavior is evident in the lift coefficient $C_L(t)$ computed from a fully resolved DIRK(3,3) time-marched solution (*red*), compared with a fully resolved Time-Spectral solution (*black*). In this case, the time-marched solution must be integrated for 3 full periods (300 or 600 timesteps) before the initial transient gives way to a periodic flow state.

lift coefficient time-series computed by the Time-Spectral HDG method with several different numbers of snapshots, N. Here we see that with only $N = 5$ snapshots (corresponding to just 2 harmonic modes), the lift coefficient time-series is surprisingly close to the fully resolved solution. With $N = 11$ snapshots it is even closer, and at $N = 21$ snapshots we can no longer see the difference visually.

For another perspective, Figure 2-8 uses the computed flow-fields to illustrate convergence of the Time-Spectral HDG method. Here we show the flow-field at a few representative snapshots in time, as computed by a Time-Spectral HDG method with $N = 5$ snapshots, a Time-Spectral HDG method with $N = 27$ snapshots, and a highly resolved time-marched HDG solution serving as a "truth" reference (computed using DIRK(3,3) with timestep $\Delta t = T/200$ as previously mentioned). The flow-field from the $N = 5$ Time-Spectral solution is clearly under-resolved in time, smearing out the wake region of the flow. However, since the flow near the airfoil appears well-

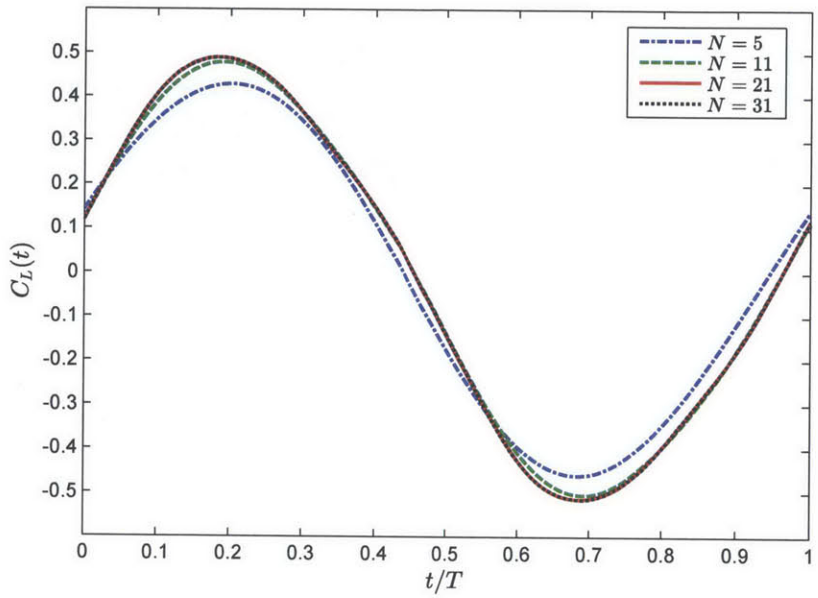

Figure 2-7: Rapid convergence of the Time-Spectral HDG solution to the pitching airfoil problem is seen in the lift coefficient $C_L(t)$, shown here for different numbers of snapshots, N.

resolved, the airfoil lift coefficient (Figure 2-7) can be predicted more accurately than one might expect. At $N = 27$ snapshots, the flow field is almost indistinguishable from the time-marched solution on the right.

To quantify the convergence of the Time-Spectral HDG method, Figure 2-9 presents the L^2-norm of the error in the lift coefficient timeseries across a range of Time-Spectral HDG solutions with different numbers of snapshots N and different spatial orders p. For each spatial order p, the error norm is computed with respect to a very highly resolved time-marched solution on a $p = 5$ mesh, obtained using the 3-stage, 3^{rd}-order DIRK scheme mentioned previously. This plot shows the exponential convergence in N that we expect of the Time-Spectral method, as we observed in the convection problems presented earlier in this paper. The different curves for each spatial order p illustrate the high-order accuracy of the HDG method, the ease with which we can obtain a more accurate solution on the same mesh, and the large gains in accuracy

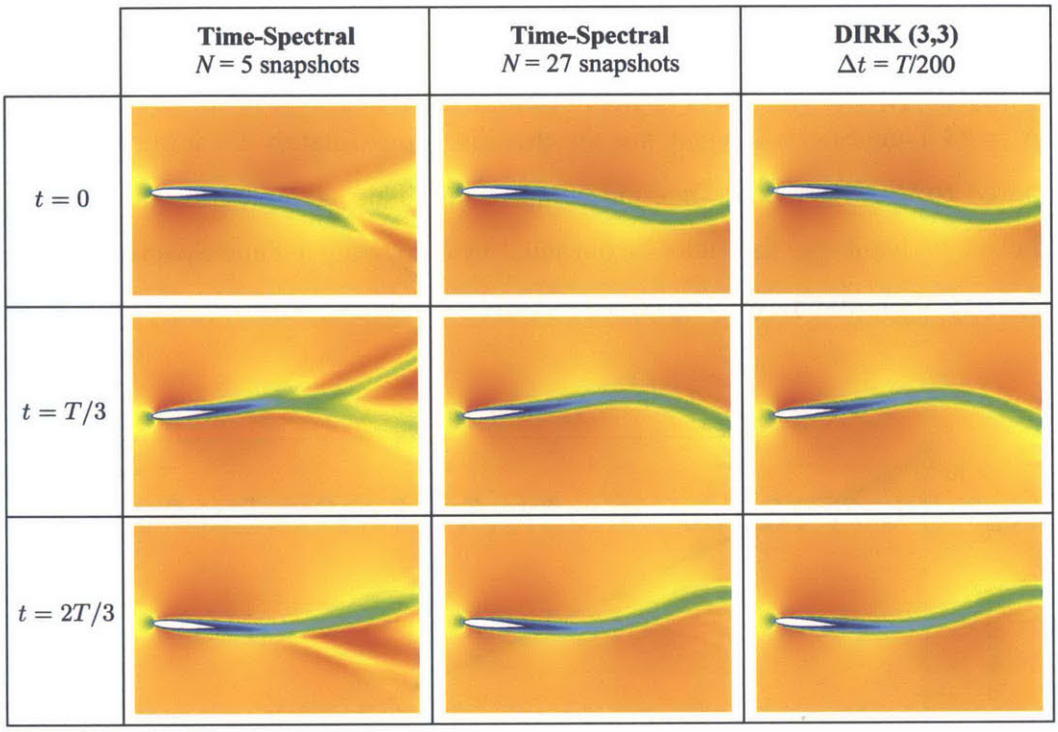

Figure 2-8: Snapshots of the periodic flow solution for our pitching airfoil problem, solved by 3 different time discretization schemes: Time-Spectral HDG with $N = 5$ snapshots; Time-Spectral HDG with $N = 27$ snapshots; and a very highly resolved time-marching HDG solution, used as a "truth" reference. The time-marched solution was obtained by a 3-stage, 3^{rd}-order Diagonally Implicit Runge-Kutta (DIRK) scheme with 200 timesteps per period over 6 periods of integration.

that can be achieved at higher p.

Finally, we note an interesting point of comparison between the Time-Spectral HDG method and the DIRK time-marched HDG method. A periodic flow solution was obtained for the pitching foil problem using a 2-stage, 2^{nd}-order DIRK scheme with $\Delta t = T/100$ over 6 periods on a $p = 5$ high-order mesh. Measuring solution error with the same metric as in Figure 2-9, the error norm was found to be 0.0036. Referring to Figure 2-9, we find that the Time-Spectral HDG solution with $N = 23$ snapshots has an error no greater than the DIRK(2,2) solution. That is, for this demonstration problem, $N = 23$ snapshots are sufficient to predict airfoil loading with the same accuracy as 600 timesteps of a 2-stage 2^{nd}-order Diagonally Implicit

Runge-Kutta time-marching scheme.

In terms of computation time, in the implementation used in the present thesis, the $N = 23$ Time-Spectral result was obtained in approximately $4\times$ fewer CPU hours, compared to the DIRK(2,2) time-marching result. This example illustrates the kind of efficiency advantages that may be obtained by employing a Time-Spectral approach rather than time-marching.

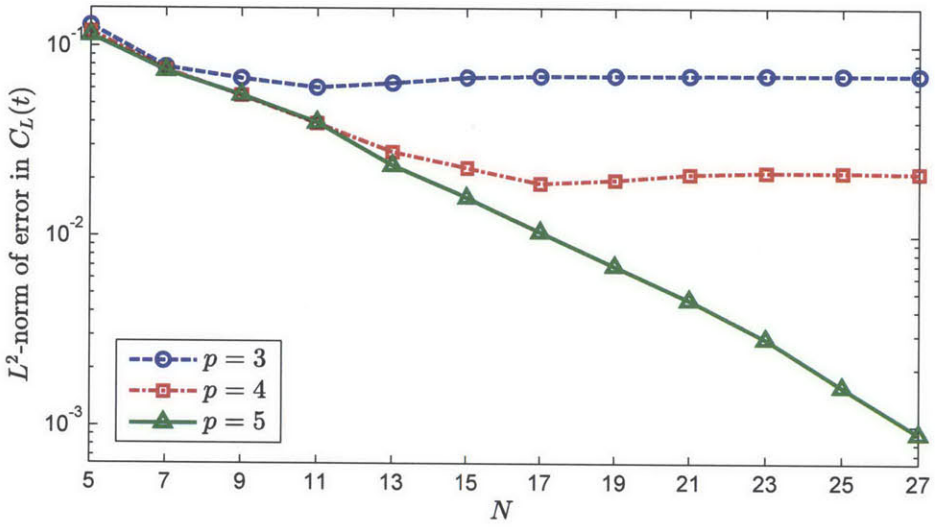

Figure 2-9: Spectral convergence of computed airfoil lift coefficient $C_L(t)$ using the Time-Spectral HDG method with an increasing number of snapshots N. The 3 curves represent different spatial polynomial orders p on the same grid, highlighting the high-order accuracy of our method. At lower spatial orders (p), we see that fewer snapshots N are required to fully resolve the solution in time.

2.6 Summary: Advantages of Time-Spectral HDG

We conclude this chapter by summarizing key advantages of the Time-Spectral HDG method for periodic flow problems:

1. **High-order accuracy in space.** The HDG spatial discretization provides a higher order of accuracy than the low-order Finite Volume methods commonly in use today. This leads to reduced numerical dissipation, and in many cases enables a higher level of solution accuracy for a lower computational cost.

2. **Fewer globally-coupled degrees of freedom than other DG methods.** This property of HDG is particularly attractive for Time-Spectral applications, since the system of equations is N times larger and thus it is even more important to minimize the size of the global system.

3. **Spectral accuracy in time.** Time-marching methods can require hundreds of timesteps before an accurate periodic flow state is reached. The Time-Spectral method has the potential to achieve the same accuracy with only a handful of Fourier modes, realizing substantial savings in computation time.

4. **No need to resolve transient behavior.** In contrast to time-marching methods, a Time-Spectral approach obviates the need to accurately resolve transient behavior of the flow during start-up.

5. **A natural framework for treating boundary conditions.** This property of HDG facilitates the formulation of a sliding mesh interface coupling technique, essential for turbomachinery problems.

A key challenge of the method is the large size of the system of equations, since the entire space-time solution with N snapshots must be solved simultaneously. However, the results in this thesis will show relevant examples where these challenges may be overcome, allowing Time-Spectral HDG to deliver high-order accurate solutions of periodic flow problems for a computational cost that is competitive with conventional methods.

Chapter 3

High-Order Sliding Interface Coupling Method for HDG

This chapter introduces a technique for coupling the solutions on two distinct subdomains which are connected by a sliding interface boundary. This technique is developed in the specific context of the Hybridizable Discontinuous Galerkin (HDG) method [72, 73, 75], but may also prove applicable to other methods of spatial discretization. This chapter first presents the formulation of a sliding interface coupling method for HDG (Section 3.1), then provides results for static and sliding grid problems to demonstrate performance characteristics of the coupling method (Section 3.2). Finally, an additional anti-aliasing filter is introduced which has proven important for Time-Spectral computations (Section 3.3)

3.1 Interface Coupling Formulation

Background

A sliding interface coupling technique is required for simulations of flow through multiple blade rows that are moving relative to one another. Sliding mesh interfaces have been treated in many different ways for different numerical methods, such as overlaid grids in Finite Difference [17], mortar methods in conforming Finite Elements [65] and

Discontinuous Galerkin (DG) methods [12], and mortar-free DG methods [39]. Mortar methods involve adding an intermediary trace element to couple the solutions on either side of the interface, avoiding any problems with nonconforming meshes. Discontinuous Galerkin methods, owing to the discontinuous nature of their discretization, are capable of performing such a coupling without a mortar element at the cost of greater implementation complexity (for example, to handle the evaluation of integrals along the interface geometry with an arbitrary number of hanging nodes). In the HDG setting a mortar element approach is very natural and simpler to implement, and for these reasons is selected for development in the present thesis. The approach bears some resemblance to the Hybridized Multiscale Discontinuous Galerkin Method (HMDG) by Nguyen *et al.* [78], though application to sliding meshes requires a more general formulation.

Formulation

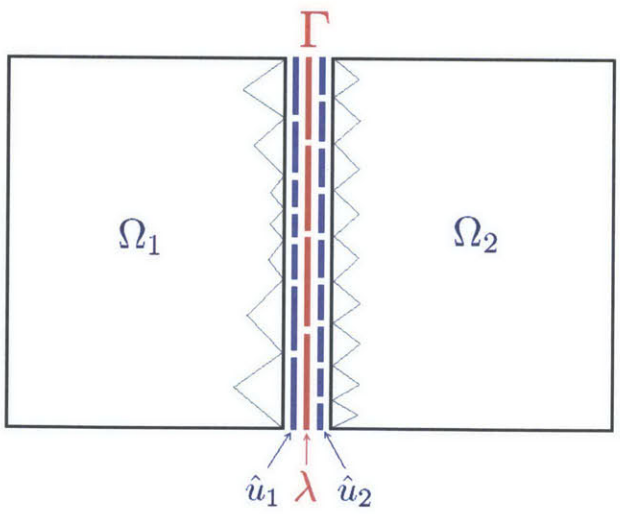

Figure 3-1: Coupling variables for a high-order interface coupling technique. A mortar variable λ is defined on the interface Γ, along which the two subdomains Ω_1 and Ω_2 are permitted to slide (vertically, in this illustration).

Figure 3-1 schematically illustrates the type of problem that requires a sliding interface coupling. A rotor/stator flow is an example of such a problem, and multi-

row turbomachinery flows are simple extensions of this base problem. Shown in the figure are two subdomains Ω_1 and Ω_2 with independent triangulations that do not necessarily conform at their shared sliding interface. Our goal is to couple the HDG solution in Ω_1 to the HDG solution in Ω_2.

Let $\mathcal{P}^p(D)$ denote the set of polynomials of degree at most p on a domain D. For any element K of the collection $\mathcal{T}_h$ we denote $\boldsymbol{W}^p(K) \equiv (\mathcal{P}^p(K))^m$ and $\boldsymbol{V}^p(K) \equiv (\mathcal{P}^p(K))^{m\times d}$, where m is the number of conserved quantities and d is the number of spatial dimensions. Also define $\boldsymbol{M}^p(e) \equiv (\mathcal{P}^p(e))^m$ for any edge in a set of edges $\mathcal{E}_h$, and let $\mathcal{E}_{h1}$ and $\mathcal{E}_{h2}$ be the complete set of faces in triangulations $\mathcal{T}_{h1}$ and $\mathcal{T}_{h2}$ of subdomains Ω_1 and Ω_2, respectively. The interface Γ will be discretized by a set of edges denoted Γ_h. Note that the sets of edges $\mathcal{E}_{h1}$ and $\mathcal{E}_{h2}$ are not limited to those edges coinciding with the interface Γ; rather, these are the complete sets of edges on the triangulation of each respective subdomain.

We can now introduce discontinuous finite element spaces on each subdomain. On subdomain Ω_1, we have approximation spaces for $(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}, \widehat{\boldsymbol{u}}_{h1})$ as follows:

$$\begin{aligned}
\boldsymbol{W}_{h1}^{p_1} &= \{\boldsymbol{w} \in (L^2(\Omega_1))^m \ : \boldsymbol{w}|_K \in \boldsymbol{W}^{p_1}(K)\ \forall K \in \mathcal{T}_{h1}\},\\
\boldsymbol{V}_{h1}^{p_1} &= \{\boldsymbol{v} \in (L^2(\Omega_1))^{m\times d} \ : \ \boldsymbol{v}|_K \in \boldsymbol{V}^{p_1}(K)\ \forall K \in \mathcal{T}_{h1}\},\\
\boldsymbol{M}_{h1}^{p_1} &= \{\boldsymbol{\mu} \in (L^2(\mathcal{E}_{h1}))^m \ : \ \boldsymbol{\mu}|_e \in \boldsymbol{M}^{p_1}(e), \forall e \in \mathcal{E}_{h1}\}.
\end{aligned}$$

and on subdomain Ω_2 for $(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}, \widehat{\boldsymbol{u}}_{h2})$:

$$\begin{aligned}
\boldsymbol{W}_{h2}^{p_2} &= \{\boldsymbol{w} \in (L^2(\Omega_2))^m \ : \boldsymbol{w}|_K \in \boldsymbol{W}^{p_2}(K)\ \forall K \in \mathcal{T}_{h2}\},\\
\boldsymbol{V}_{h2}^{p_2} &= \{\boldsymbol{v} \in (L^2(\Omega_2))^{m\times d} \ : \ \boldsymbol{v}|_K \in \boldsymbol{V}^{p_2}(K)\ \forall K \in \mathcal{T}_{h2}\},\\
\boldsymbol{M}_{h2}^{p_2} &= \{\boldsymbol{\mu} \in (L^2(\mathcal{E}_{h2}))^m \ : \ \boldsymbol{\mu}|_e \in \boldsymbol{M}^{p_2}(e), \forall e \in \mathcal{E}_{h2}\}.
\end{aligned}$$

Finally, an additional approximation space on the $d-1$ dimensional interface Γ for the mortar variable $\boldsymbol{\lambda}_h$:

$$\boldsymbol{N}_h^{p_\lambda} = \{\boldsymbol{\nu} \in (L^2(\Gamma))^m \ : \ \boldsymbol{\nu}|_e \in \boldsymbol{M}^{p_\lambda}(e), \forall e \in \Gamma_h\} \tag{3.1}$$

The interface (or mortar) variable $\boldsymbol{\lambda}_h$ belongs to this space $\boldsymbol{N}_h^{p_\lambda}$, and has polynomial degree p_λ on each face of the interface Γ_h. This polynomial order is not necessarily the same as the polynomial degree of the solutions on the left and right (p_1 and p_2). Each mortar element is discretized using Chebyshev points, and is associated with test functions $\boldsymbol{\nu}$ in the space $\boldsymbol{N}_h^{p_\lambda}$.

We can now proceed to describe the HDG equations for two subdomains coupled through an interface Γ_h. The system of equations consists of a separate set of HDG equations from each subdomain, taken from Section 2.2 and slightly modified by a coupling term with $\boldsymbol{\lambda}_h$. This coupling term implements a Dirichlet boundary condition on the interface, coupling each subdomain's solution to the mortar variable $\boldsymbol{\lambda}_h$. In addition to these sets of equations on each subdomain, a final set of equations is defined to enforce flux continuity across the interface Γ_h.

The HDG system of equations for two coupled subdomains is thus as follows:

$$
\begin{aligned}
(\boldsymbol{q}_{h1}, \boldsymbol{v}_1)_{\mathcal{T}_{h1}} - (\boldsymbol{u}_{h1}, \nabla \cdot \boldsymbol{v}_1)_{\mathcal{T}_{h1}} + \langle \widehat{\boldsymbol{u}}_{h1}, \boldsymbol{v}_1 \cdot \boldsymbol{n}_1 \rangle_{\partial \mathcal{T}_{h1}} &= 0, \\
\left(\frac{\partial \boldsymbol{u}_{h1}}{\partial t}, \boldsymbol{w}_1 \right)_{\mathcal{T}_{h1}} - (\boldsymbol{F}(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}), \nabla \boldsymbol{w}_1)_{\mathcal{T}_{h1}} + \left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}, \widehat{\boldsymbol{u}}_{h1}) \cdot \boldsymbol{n}_1, \boldsymbol{w}_1 \right\rangle_{\partial \mathcal{T}_{h1}} &= (\boldsymbol{f}_1, \boldsymbol{w}_1)_{\mathcal{T}_{h1}} \\
\left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}, \widehat{\boldsymbol{u}}_{h1}) \cdot \boldsymbol{n}_1, \boldsymbol{\mu}_1 \right\rangle_{\partial \mathcal{T}_{h1} \setminus \partial \Omega} + \left\langle \widehat{\boldsymbol{B}}_{h1}, \boldsymbol{\mu}_1 \right\rangle_{\partial \Omega} - \langle \boldsymbol{\lambda}_h, \boldsymbol{\mu}_1 \rangle_{\Gamma_h} &= 0
\end{aligned}
$$

$$
\begin{aligned}
(\boldsymbol{q}_{h2}, \boldsymbol{v}_2)_{\mathcal{T}_{h2}} - (\boldsymbol{u}_{h2}, \nabla \cdot \boldsymbol{v}_2)_{\mathcal{T}_{h2}} + \langle \widehat{\boldsymbol{u}}_{h2}, \boldsymbol{v}_2 \cdot \boldsymbol{n}_2 \rangle_{\partial \mathcal{T}_{h2}} &= 0, \\
\left(\frac{\partial \boldsymbol{u}_{h2}}{\partial t}, \boldsymbol{w}_2 \right)_{\mathcal{T}_{h2}} - (\boldsymbol{F}(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}), \nabla \boldsymbol{w}_2)_{\mathcal{T}_{h2}} + \left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}, \widehat{\boldsymbol{u}}_{h2}) \cdot \boldsymbol{n}_2, \boldsymbol{w}_2 \right\rangle_{\partial \mathcal{T}_{h2}} &= (\boldsymbol{f}_2, \boldsymbol{w}_2)_{\mathcal{T}_{h2}} \\
\left\langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}, \widehat{\boldsymbol{u}}_{h2}) \cdot \boldsymbol{n}_2, \boldsymbol{\mu}_2 \right\rangle_{\partial \mathcal{T}_{h2} \setminus \partial \Omega} + \left\langle \widehat{\boldsymbol{B}}_{h2}, \boldsymbol{\mu}_2 \right\rangle_{\partial \Omega} - \langle \boldsymbol{\lambda}_h, \boldsymbol{\mu}_2 \rangle_{\Gamma_h} &= 0
\end{aligned}
$$

$$
\left\langle \left(\widehat{\boldsymbol{F}}_1(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}, \boldsymbol{\lambda}_h) \cdot \boldsymbol{n}_1 + \widehat{\boldsymbol{F}}_2(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}, \boldsymbol{\lambda}_h) \cdot \boldsymbol{n}_2 \right), \boldsymbol{\nu} \right\rangle_{\Gamma_h} = 0 \tag{3.2}
$$

for all:

$$
(\boldsymbol{v}_1, \boldsymbol{w}_1, \boldsymbol{\mu}_1, \boldsymbol{v}_2, \boldsymbol{w}_2, \boldsymbol{\mu}_2, \boldsymbol{\nu}) \in \boldsymbol{Q}_{h1}^{p_1} \times \boldsymbol{W}_{h1}^{p_1} \times \boldsymbol{M}_{h1}^{p_1}(0) \times \boldsymbol{Q}_{h2}^{p_2} \times \boldsymbol{W}_{h2}^{p_2} \times \boldsymbol{M}_{h2}^{p_2}(0) \times \boldsymbol{N}_h^{p_\lambda}
$$

The boundary flux terms $\widehat{\boldsymbol{B}}_{h1}$ and $\widehat{\boldsymbol{B}}_{h2}$ are defined according to the boundary conditions chosen on each subdomain, with homogeneous Dirichlet boundary condi-

tions applied on the boundaries coinciding with Γ_h. The numerical flux at the sliding interface is defined:

$$\widehat{\boldsymbol{F}}_1(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}, \boldsymbol{\lambda}_h) \cdot \boldsymbol{n}_1 = \boldsymbol{F}_1(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}) \cdot \boldsymbol{n}_1 + \tau(\boldsymbol{u}_{h1} - \boldsymbol{\lambda}_h) \tag{3.3}$$

$$\widehat{\boldsymbol{F}}_2(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}, \boldsymbol{\lambda}_h) \cdot \boldsymbol{n}_2 = \boldsymbol{F}_2(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}) \cdot \boldsymbol{n}_2 + \tau(\boldsymbol{u}_{h2} - \boldsymbol{\lambda}_h) \tag{3.4}$$

with a constant stabilization term τ.

Equations 3.2-3.4 define a set of coupled nonlinear equations which simultaneously describe the solution on the left subdomain, the right subdomain, and the interface. These equations are solved by applying a Newton-Raphson procedure, and applying static condensation to the resulting linearized system. Defining the following:

$$\boldsymbol{R}_{\lambda 1} = \left\langle \widehat{\boldsymbol{F}}_1(\boldsymbol{u}_{h1}, \boldsymbol{q}_{h1}, \boldsymbol{\lambda}_h) \cdot \boldsymbol{n}_1, \boldsymbol{\nu} \right\rangle_{\Gamma_h} \tag{3.5}$$

$$\boldsymbol{R}_{\lambda 2} = \left\langle \widehat{\boldsymbol{F}}_2(\boldsymbol{u}_{h2}, \boldsymbol{q}_{h2}, \boldsymbol{\lambda}_h) \cdot \boldsymbol{n}_2, \boldsymbol{\nu} \right\rangle_{\Gamma_h} \tag{3.6}$$

we ultimately obtain a coupled global matrix system to be solved at each Newton iteration. This system is of the form:

$$\begin{bmatrix} \mathbb{K}_1 & 0 & -\mathbb{B}_1 \\ 0 & \mathbb{K}_2 & -\mathbb{B}_2 \\ \mathbb{G}_1 & \mathbb{G}_2 & \mathbb{H}_1 + \mathbb{H}_2 \end{bmatrix} \begin{bmatrix} \delta\widehat{\boldsymbol{U}}_1 \\ \delta\widehat{\boldsymbol{U}}_2 \\ \delta\boldsymbol{\Lambda} \end{bmatrix} = \begin{bmatrix} -\mathbb{F}_1 + \mathbb{B}_1\boldsymbol{\Lambda} \\ -\mathbb{F}_2 + \mathbb{B}_2\boldsymbol{\Lambda} \\ -(\boldsymbol{R}_{\lambda 1} + \boldsymbol{R}_{\lambda 2}) \end{bmatrix} \tag{3.7}$$

where $(\delta\widehat{\boldsymbol{U}}_1, \delta\widehat{\boldsymbol{U}}_2, \delta\boldsymbol{\Lambda})$ are the Newton update vectors associated with $(\widehat{\boldsymbol{u}}_{h1}, \widehat{\boldsymbol{u}}_{h2}, \boldsymbol{\lambda}_h)$ respectively. Note that $(\mathbb{K}_1, \mathbb{F}_1)$ and $(\mathbb{K}_2, \mathbb{F}_2)$ are the Jacobian matrices and nonlinear residual vectors arising from linearization of the HDG equations on each subdomain independently, applying homogeneous Dirichlet boundary conditions at the interface Γ. Also, $\mathbb{B}_1$, $\mathbb{B}_2$, $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{H}_1$ and $\mathbb{H}_2$ are coupling matrices which arise from linearization of the coupled system of equations 3.2.

3.2 Solution Accuracy for Static and Sliding Grid Problems

Here we present solutions of problems with sliding interface coupling between subdomains, in order to demonstrate performance characteristics of the method.

3.2.1 Poisson Equation, Static Grid

We begin with a steady Poisson problem on a static grid, to test the performance of the sliding interface coupling method. The governing equations being solved are:

$$\begin{aligned} \boldsymbol{q} + \kappa \nabla u &= 0, \qquad \text{in } \Omega_1 \cup \Omega_2 \\ \nabla \cdot \boldsymbol{q} &= f, \qquad \text{in } \Omega_1 \cup \Omega_2 \\ u &= g_D, \qquad \text{on } \Gamma_D \end{aligned} \tag{3.8}$$

and in this demonstration we have chosen $\kappa = f = 1$. Forming a partitioned domain with Ω_1 and Ω_2, we are able to solve the Poisson problem on a square domain, even though the meshes of the two subdomains are non-conformal at the interface. An example solution is presented in Figure 3-2, with polynomial order $p = 4$ on each subdomain and polynomial order $k = 10$ for λ. As shown in Figure 3-3, we observe optimal convergence of the solution on the partitioned domain with interface coupling, with the L_∞ norm of solution error converging with rate $\mathcal{O}(h^{p+1})$ for three different values of p.

An important question to ask about the interface coupling technique is: how does the polynomial order of λ affect solution accuracy in the subdomains? Figure 3-4 presents the results of an experiment which addresses this question. The Poisson problem described above was solved again with several different polynomial orders p on the same partitioned spatial mesh, first with interface polynomial order $k = 6$ and then with $k = 12$. The results with $k = 12$ show the expected exponential convergence with p, whereas the $k = 6$ case shows that solution accuracy is limited for

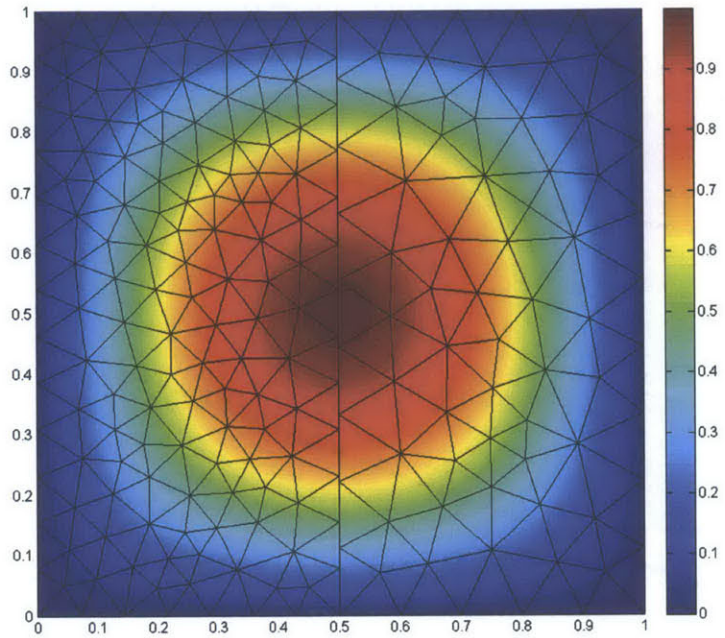

Figure 3-2: Poisson equation solution on partitioned grid with high-order interface.

$p \geq 4$. In this case, the polynomial order of the interface variable λ is too low to maintain accurate coupling of the solution across the interface when $p \geq 4$. This is not surprising, as the λ on each mortar element is required to accurately represent a piecewise discontinuous polynomial of order p from the subdomain solution (this being piecewise discontinuous due to nonconformity between the mortar and subdomain discretizations). To represent a piecewise polynomial with degree p, the mortar variable λ must have a higher polynomial degree $k > p$ to ensure a sufficiently rich approximation space. In the simulations that follow in this thesis, a typical choice will be $p = 4$ and $k = 10$.

3.2.2 Viscous Flow over an Airfoil, Sliding Grid

To demonstrate the sliding interface coupling for a nonlinear and unsteady problem with a sliding grid motion, this subsection presents results for laminar compressible

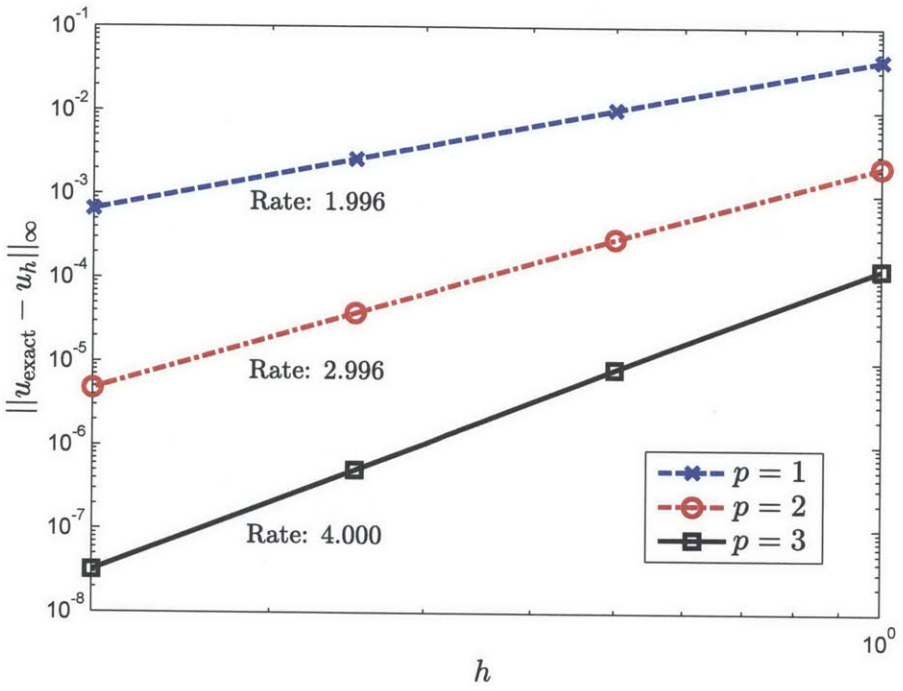

Figure 3-3: Convergence of Poisson solution vs. h for various polynomial orders p, with high-order $k = 10$ interface. Convergence rates are labelled and match the expected optimal convergence rate of $\mathcal{O}(h^{p+1})$. Note that no HDG post-processing was applied here.

flow over a NACA 0006 symmetric airfoil at $Re = 2400$. The mesh for this problem consists of two subdomains coupled through a sliding interface, and the outflow subdomain is sliding with a constant grid velocity. Periodic boundary conditions are employed between the upper and lower boundaries of each subdomain. Inflow Mach number is 0.2, and the polynomial order in each subdomain is $p = 4$. The interface is discretized with 4 elements of polynomial degree $k = 10$. Grid motion of the outflow subdomain is taken into account by an Arbitrary Lagrangian Eulerian (ALE) formulation of the Navier-Stokes equations [83]. The solution is integrated forward in time using implicit time-marching schemes.

Figure 3-5 presents a snapshot of the flow solution during time-marching. Visually, the wake behind the NACA airfoil appears to propagate correctly across the interface

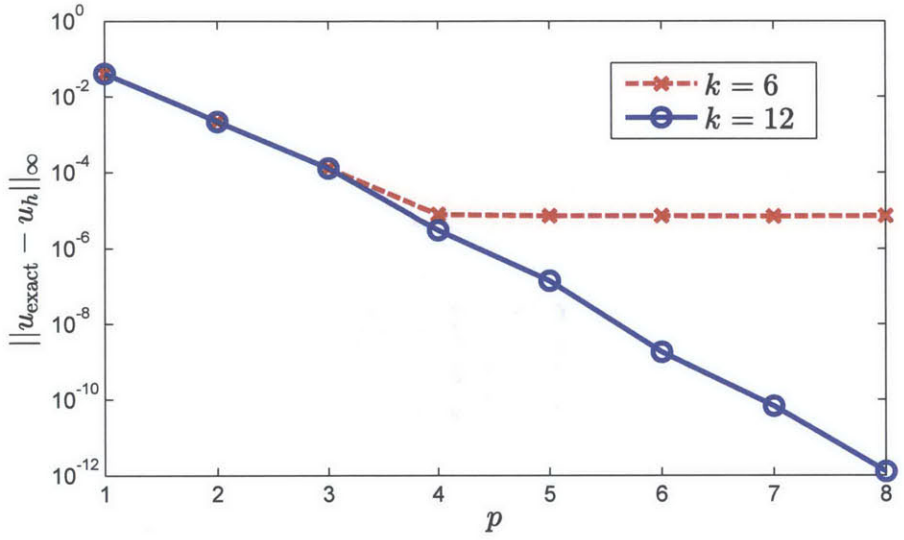

Figure 3-4: Convergence of Poisson solution vs. subdomain polynomial order p, for two different choices of the interface polynomial order k used for λ.

and continues into the outflow domain while keeping its basic character. Measuring the wake velocity profile at the outflow boundary, we find in Figure 3-6 that the wake velocity profile converges to the expected result, which was computed without any grid velocity. In Figure 3-6 we also find that time-marching accuracy is critical for solution accuracy across sliding interfaces, as we see the results from Backward Euler time-marching eventually converge towards the steady "truth" solution. This confirms the correct and accurate functioning of the sliding interface coupling technique for turbomachinery flow problems.

3.3 Anti-Aliasing Filter

Understanding Aliasing Error

When a sliding interface is used together with a Time-Spectral discretization in time, aliasing error can be introduced in the transfer of the solution across the interface. Such aliasing error has also been reported for Time-Spectral Finite Volume simulations

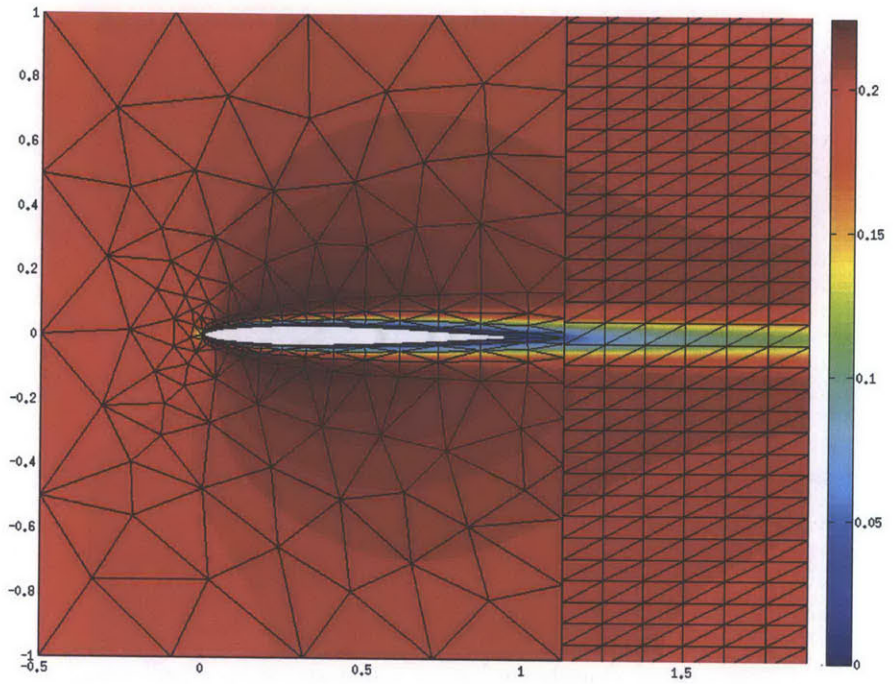

Figure 3-5: Snapshot of NACA laminar flow problem with vertically sliding outflow domain (*right*) and sliding interface coupling. Contours of Mach number.

by Gopinath *et al.* [41].

Consider a flow problem such as in Figure 3-5. Here we have a stationary subdomain on the left, and a vertically sliding subdomain on the right. The wake that has formed behind the airfoil must be transferred across the interface. Note that the width of the wake is quite small relative to the total length of the interface.

Flow variables will be steady in time on the left-hand subdomain, but on the right-hand subdomain each grid point should see a time-varying solution. In fact, nodes at the sliding interface boundary on the right will see the solution vary in time the same way as the solution on the left varies in space (y direction), due to the vertical sliding motion of the right-hand subdomain. However, since the Time-Spectral method is based on a discrete set of snapshots in time, a node on the right-hand side of the interface will not see a continuous view of the solution from the left, but rather a *discretely sampled* version of the continuous time variation. This discrete sampling of the time-varying solution across the interface can result in *aliasing error* if the

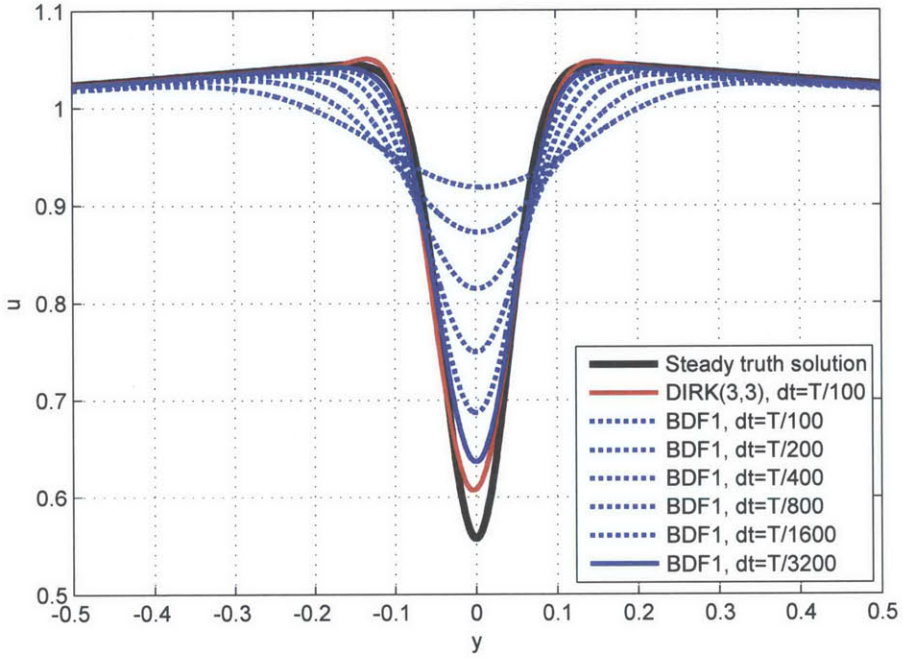

Figure 3-6: Effect of time-marching accuracy on wake profile at outflow boundary, for flow over a NACA airfoil with sliding outflow subdomain.

spectral content of the transferred solution is too high to be accurately represented on the given set of discrete time-samples.

To illustrate the issue further, Figure 3-7 shows two examples of a narrow solution feature and the spectral truncation error arising from approximating this profile with a finite number of modes K. In this case, we use a Gaussian profile as an example of a narrow feature, but flow features such as wakes will behave exactly the same way. In the Figure, we see that as we increase the number of modes used to approximate the solution, the spectral truncation error decreases, but as the solution feature is made narrower, more modes are required to reach the same level of accuracy. This is summarized more generally in Figure 3-8, which shows the number of modes K required to approximate a Gaussian with a given width BP/w at a certain level of accuracy (shown on the contours). It turns out that the number of modes required to

represent a Gaussian-like solution feature is linearly proportional to the width ratio BP/w. This will be an important observation that we will refer to in later chapters when presenting multi-row turbomachinery flow simulations with the Time-Spectral method.

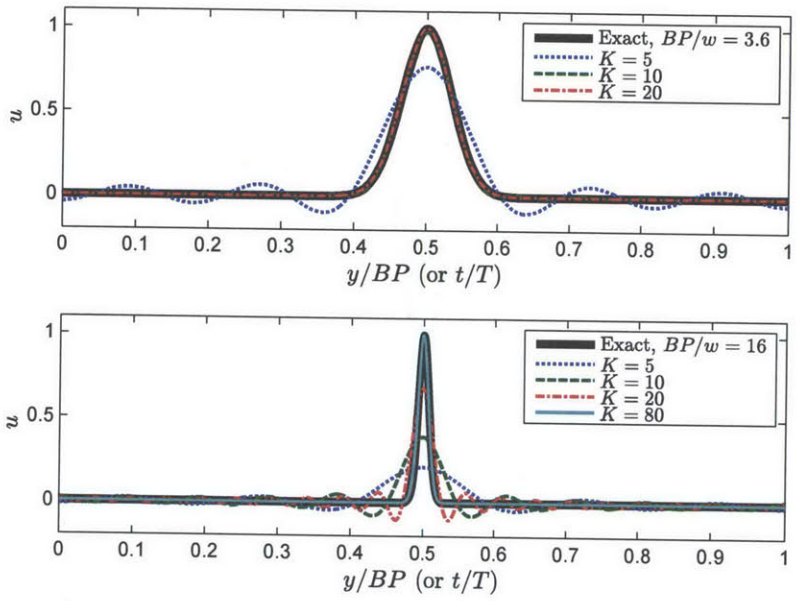

Figure 3-7: Example of a narrow solution feature at the interface, and the spectral truncation error which occurs when approximating with a finite number of Fourier modes K. Two cases are presented here, based on the ratio of the interface length (blade pitch) to the feature width: $BP/w = 3.6$ and $BP/w = 16$.

Controlling Aliasing via an Anti-Aliasing Filter

Now that we understand the problem of aliasing error, we are ready to present the solution employed in the present work. An anti-aliasing filter has been implemented, which combats the fundamental issue of *under-sampling* in time at the interface by choosing instead to *over-sample* for the interface coupling equations.

The operation of the filter can be illustrated as follows. In a Time-Spectral simulation, all spatial degrees of freedom are expanded N times to represent each of

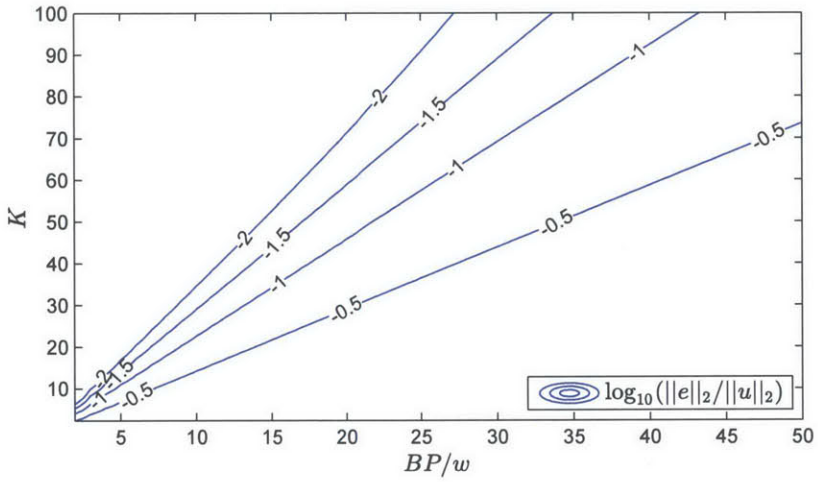

Figure 3-8: Spectral truncation error for Gaussian wake velocity profiles with different width ratios BP/w, evaluated with different total numbers of modes K.

the N snapshots in time. Thus, u_h becomes $\boldsymbol{u_h} = [u_0, \ldots, u_{N-1}]^T$, $\vec{q}_h$ becomes $\boldsymbol{q_h} = [\vec{q}_0, \ldots, \vec{q}_{N-1}]^T$, and $\hat{u}$ becomes $\hat{\boldsymbol{u}}_{\boldsymbol{h}} = [\hat{u}_0, \ldots, \hat{u}_{N-1}]^T$ in each subdomain (see Section 2.3). Normally we might expand the interface variable λ the same way. However, in the anti-aliasing filter approach, we expand λ to more than N snapshots in time – let us call it M snapshots, so that $\boldsymbol{\lambda} = [\lambda_0, \ldots, \lambda_{M-1}]^T$.

When $M = N$, we recover the original Time-Spectral HDG method with sliding interface, where λ_i only couples $(u_h, \vec{q}_h, \hat{u}_h)_i$ across the interface. However, when $M > N$, our interface coupling equations must be expressed at $M > N$ snapshots in time, based on $(u_h, \vec{q}_h, \hat{u}_h)$ defined at only N snapshots. Specifically, with reference to the previously presented set of HDG equations for coupled subdomains (Equation 3.2):

1. To construct the interface flux continuity equations, we must simply interpolate the subdomain solutions $(u_h, \vec{q}_h, \hat{u}_h)_i$ to the time coordinates of the M snapshots.

2. To construct the Dirichlet boundary condition terms on each subdomain, we must project the interface variable $\boldsymbol{\lambda}_h$ from M to N snapshots, retaining only the frequencies that are resolved by $(\boldsymbol{u_h}, \boldsymbol{q_h}, \hat{\boldsymbol{u}}_{\boldsymbol{h}})$ on each side of the interface.

(This is the *anti-aliasing filter.*)

Let us define these interpolation and projection operators more carefully. Consider a time-periodic scalar signal $u(t)$. Let u_M and u_N be vectors containing uniform discrete samples of $u(t)$, where $M > N$. Introducing Discrete Fourier Transform operators E_N and E_M, we have:

$$\tilde{U}_K = E_N u_N \tag{3.9}$$

$$\tilde{U}_L = E_M u_M \tag{3.10}$$

where $\tilde{U}_K$ and $\tilde{U}_L$ are vectors of N and M complex Fourier coefficients, corresponding to $K = (N-1)/2$ and $L = (M-1)/2$ harmonic modes, respectively. Note that E_N and E_M are respectively $N \times N$ and $M \times M$ complex matrices. Defining I_N and I_M as $N \times N$ and $M \times M$ identity matrices, we can now define the projection and interpolation operators in matrix form:

$$\mathcal{P}_\lambda = E_N^{-1} \left[I_N \; 0 \right] E_M \tag{3.11}$$

$$\mathcal{I}_\lambda = E_M^{-1} \begin{bmatrix} I_N \\ 0 \end{bmatrix} E_N \tag{3.12}$$

Thus, we can now write a Time-Spectral version of the sliding interface equations (Eq. 3.2) with an anti-aliasing filter as described above. The form of the equations is largely the same, except for these modifications: Time-Spectral approximation spaces instead of single-snapshot versions; a Time-Spectral time-derivative term; and application of the projection and interpolation operators $\mathcal{P}_\lambda$ and $\mathcal{I}_\lambda$.

Since the adaptation of Eq. 3.2 to a Time-Spectral problem is fairly straightforward, we will not rewrite the equations here in full. However, to highlight the changes that will be made to a sliding Time-Spectral HDG formulation to accommodate an anti-aliasing filter, let us consider only the Time-Spectral HDG equations associated

with the nonlinear residuals for $\widehat{\boldsymbol{u}}_{h1}$ and $\widehat{\boldsymbol{u}}_{h2}$, and the equations enforcing flux continuity across the interface Γ. Let us also use $\boldsymbol{R}_{\hat{u}1}$ and $\boldsymbol{R}_{\hat{u}1}$ to denote the Time-Spectral HDG nonlinear residuals on each subdomain, with homogeneous Dirichlet boundary conditions on the interface Γ. Now, with the application of our anti-aliasing filter, the nonlinear residual equations for $\widehat{\boldsymbol{u}}_{h1}$, $\widehat{\boldsymbol{u}}_{h2}$ and $\boldsymbol{\lambda}_h$ will become:

$$\boldsymbol{R}_{\hat{u}1} - \langle \mathcal{P}_\lambda \boldsymbol{\lambda}_h, \boldsymbol{\mu}_1 \rangle_\Gamma = 0 \quad (3.13)$$

$$\boldsymbol{R}_{\hat{u}2} - \langle \mathcal{P}_\lambda \boldsymbol{\lambda}_h, \boldsymbol{\mu}_2 \rangle_\Gamma = 0 \quad (3.14)$$

$$\left\langle \left(\widehat{\boldsymbol{F}}_1(\mathcal{I}_\lambda \boldsymbol{u}_{h1}, \mathcal{I}_\lambda \boldsymbol{q}_{h1}, \boldsymbol{\lambda}) \cdot \boldsymbol{n}_1 + \widehat{\boldsymbol{F}}_2(\mathcal{I}_\lambda \boldsymbol{u}_{h2}, \mathcal{I}_\lambda \boldsymbol{q}_{h2}, \boldsymbol{\lambda}) \cdot \boldsymbol{n}_2 \right), \boldsymbol{\nu} \right\rangle_\Gamma = 0 \quad (3.15)$$

where we note that in the Time-Spectral context, the interface numerical fluxes $\widehat{\boldsymbol{F}}_1$ and $\widehat{\boldsymbol{F}}_2$ are an assembly of the numerical fluxes computed at all N snapshots in time.

Demonstration

To demonstrate the impact of the anti-aliasing filter on the accuracy of Time-Spectral HDG calculations with sliding interfaces, consider a rotor/stator flow problem with a 2:1 ratio of stator blades to rotor blades. Examples of this kind of flow problem will be presented in much more detail in the later chapters, but here we briefly present one such result for the purpose of illustration.

Figure 3-9 presents the spectrum of total force on the rotor blade, computed several ways – first using a high-order DIRK(3,3) implicit time-marching HDG scheme, and then using a Time-Spectral HDG scheme with several different numbers of modes resolved. The time-marching solution is computed with a very fine time-step so as to make this an acceptable "truth solution" on the given spatial grid. All calculations shown here were performed on the same $p = 4$ high-order spatial mesh.

What we notice from Figure 3-9 is that the truth solution has zero amplitude in the odd frequencies, whereas the Time-Spectral results have nonzero amplitude in those odd frequencies. The 2:1 blade count ratio in this problem guarantees that the rotor force should only contain even frequencies, so these nonzero amplitudes in the odd

modes are a clear sign of error. Note that the computation presented in Figure 3-9 was performed without an anti-aliasing filter at the interface.

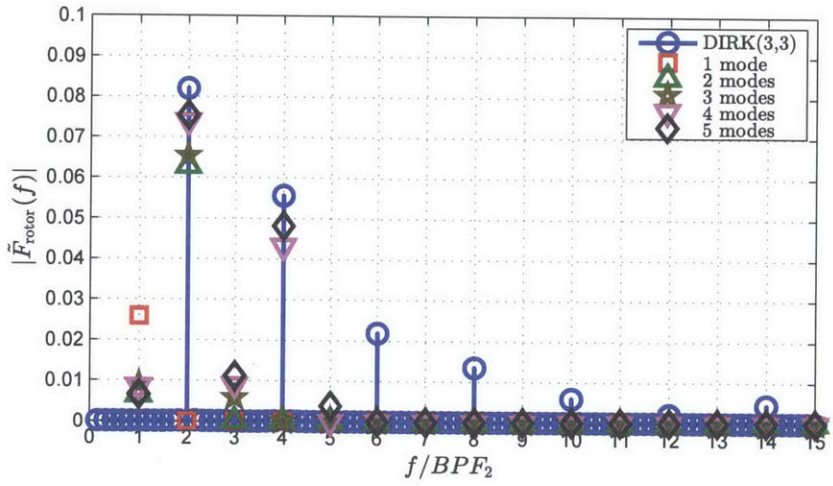

Figure 3-9: Rotor force spectrum for a 2:1 rotor/stator flow problem, solved using Time-Spectral HDG without an anti-aliasing filter at the sliding interface.

Now let us see how the solution changes when an anti-aliasing filter is added at the interface. Figure 3-10 presents the rotor force spectrum resulting from a new computation that includes an anti-aliasing filter at the interface. We immediately see that the rotor force amplitude at odd frequencies has been cut down to nearly zero, a property we expected to see in the exact solution. This is clearly an improvement in the accuracy of solution transfer across the interface.

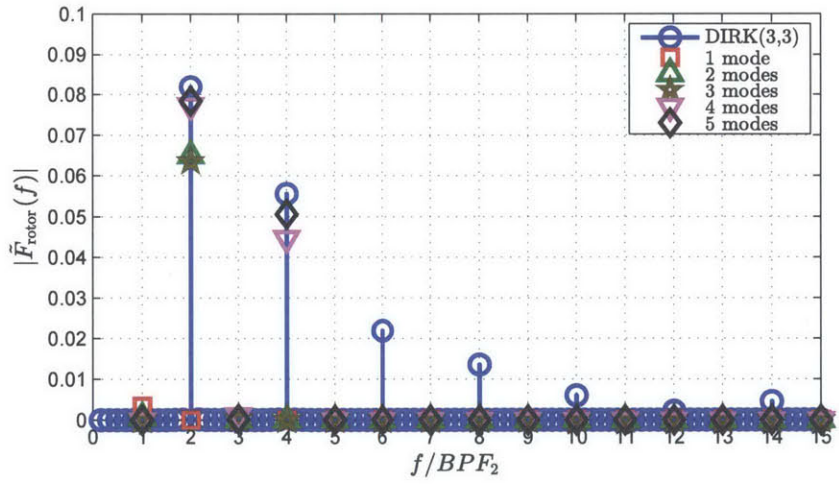

Figure 3-10: Rotor force spectrum for a 2:1 rotor/stator flow problem, solved using Time-Spectral HDG with an anti-aliasing filter at the sliding interface.

Chapter 4

Turbomachinery Flow Simulations

This chapter presents the results of applying Time-Spectral HDG and the sliding interface coupling method to turbomachinery flow problems. These flows are modeled as fully turbulent, and are computed by solving the Reynolds-Averaged Navier-Stokes (RANS) equations.

We first note some necessary regularizations that were applied to the Spalart-Allmaras (SA) turbulence model, to aid numerical stability with our high-order finite element method (Section 4.1). We then present results for a rotor cascade (Section 4.2), rotor/stator flow (Section 4.3) and three-row compressor (Section 4.4). Interesting performance characteristics of the new method are highlighted through comparisons of time-marching with Time-Spectral, and spatially high-order versus low-order (Section 4.5). These comparisons provide a quantitative demonstration of the advantages that may be derived from a Time-Spectral HDG method for solving turbomachinery flows.

4.1 Regularization of the Spalart-Allmaras (SA) Turbulence Model

Background

Industrial turbomachinery flows are most often modeled as fully turbulent, and there are several different approaches for modeling turbulent flow. The more common approach is to use the Reynolds-Averaged Navier-Stokes (RANS) equations together with a closure model for the eddy viscosity $\tilde{\nu}$ which models the effect of turbulence. Among closure models, popular choices include the Spalart-Allmaras (SA) one-equation model [94], the Baldwin-Lomax algebraic turbulence model [5], the $k-\omega$ model introduced by Wilcox [109] and the $k-\epsilon$ model introduced by Jones & Launder [61]. Turbulent flow can also be modeled with greater fidelity by Large Eddy Simulation [89], but such LES computations are currently considered too expensive for most practical applications in industry.

In the present thesis, turbulent flow is modeled using the RANS equations with the SA model. This choice was made due to the popularity of this method in industry and the relative simplicity of the model compared to the two-equation models. The HDG method has been applied to solve steady RANS-SA flow problems by Moro *et al.* [69], and in this thesis the application is extended to unsteady turbulent flows. As we will see, this has introduced a need for new regularizations of the SA model.

While developing HDG simulations of unsteady turbulent flow, problems were encountered where certain parameters of the SA model caused numerical instability. This has been observed by other researchers solving the RANS-SA equations with high-order finite element methods, including recent work by Allmaras *et al.* [2] and Moro *et al.* [69]. They found that at the boundary between turbulent and irrotational regions of the flow (for example, at the edge of the boundary layer), the eddy viscosity parameter can become negative as it falls away from a high value in the turbulent region to a near-zero value in the irrotational region. Numerical dissipation in low-order Finite Volume methods helps to mitigate this problem, but spatially high-order

accurate methods are much more prone to negative eddy viscosity and oscillations in these boundary zones, which can lead to numerical instability. For this reason, Moro *et al.* and Allmaras *et al.* both proposed modifications to the SA model to regularize the eddy viscosity parameter, and in the present thesis a regularization is also applied. The present work includes regularizations of the modified vorticity $\bar{S}$ and the parameter r of the SA model.

For the sake of completeness, these regularizations are briefly summarized below.

Regularization of eddy viscosity parameter χ

The RANS-SA equations are as follows [2]:

$$\frac{D\tilde{\nu}}{Dt} = c_{b1}\tilde{S}\tilde{\nu} + \frac{1}{\sigma}\left[\nabla\cdot((\nu+\tilde{\nu})\nabla\tilde{\nu}) + c_{b2}(\nabla\tilde{\nu})^2\right] - c_{w1}f_w\left[\frac{\tilde{\nu}}{d}\right]^2 \tag{4.1}$$

with:

$$\mu_t = \rho\nu_t, \quad \nu_t = \tilde{\nu}f_{v1}, \quad f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \chi = \frac{\tilde{\nu}}{\nu}$$

$$\tilde{S} = S + \frac{\tilde{\nu}}{\kappa^2 d^2}f_{v2}, \quad S = \sqrt{2\Omega_{ij}\Omega_{ij}}, \quad f_{v2} = 1 - \frac{\chi}{1+\chi f_{v1}}$$

$$f_w = g\left[\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right]^{1/6}, \quad g = r + c_{w2}(r^6 - r), \quad r = \frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}$$

The first regularization that was applied in the present thesis is a regularization of the eddy viscosity parameter χ, which has a tendency to become negative and oscillatory in the zone between turbulent flow and irrotational flow (for example, at the edge of the boundary layer). To prevent such behavior, we replace χ with a regularized version ψ defined as follows:

$$\psi = \chi(\arctan(b\chi)/\pi + 1/2) + c \tag{4.2}$$

$$b = 100 \tag{4.3}$$

$$c = 1/2 - \arctan(b)/\pi \tag{4.4}$$

This is similar to regularizations presented by Allmaras *et al.* [2] and Moro *et al.* [69], though the functional form of this regularization is slightly different, and we add further regularization of $\tilde{S}$ and r to the χ regularization proposed by Moro *et al.*. The particular forms presented in this thesis were chosen so that regularization can be applied as a single continuous function with continuous gradient, which both simplifies implementation and helps ensure convergence when the Newton-Raphson method is applied to solve the RANS-SA equations. However, it is worth noting that with a careful implementation in the context of the present method, we expect the regularizations by Allmaras *et al.* [2] to perform well also.

Figure 4-1 illustrates the form of the function. The regularization ensures that eddy viscosity never becomes negative. The smooth transition at $\chi = 0$ introduces a small positive eddy viscosity ψ in that condition, though it should be noted that this small value is only 0.3% of the molecular viscosity, so its effect on the flow physics can be expected to be negligible.

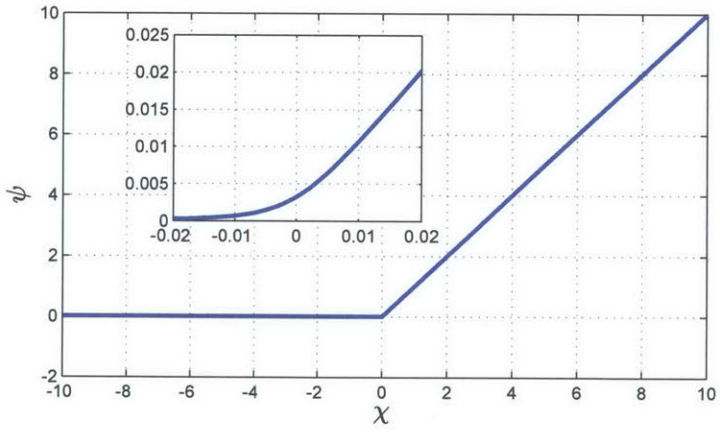

Figure 4-1: Regularization of eddy viscosity parameter, $\psi(\chi)$

Regularization of modified vorticity $\bar{S}$

Another important parameter to regularize is the modified vorticity $\bar{S}$, as physical considerations imply that this should never become negative, though numerically this

is possible in the unregularized model. Negative modified vorticity can lead to numerical stability issues, as it leads to a negative production of eddy viscosity, which can in turn lead to negative eddy viscosity and unstable anti-dissipative effects on the flow.

The regularization employed in the present thesis is another single continuous function, for the same reasons as discussed for $\psi(\chi)$:

$$\tilde{S} = 0.1S + (\bar{S} + 0.9S)(\arctan(b(\bar{S}/S + 0.9))/\pi + 1/2) + cS \tag{4.5}$$

$$b = 100, \quad c = 1/2 - \arctan(b)/\pi \tag{4.6}$$

Figure 4-2 illustrates the form of this function.

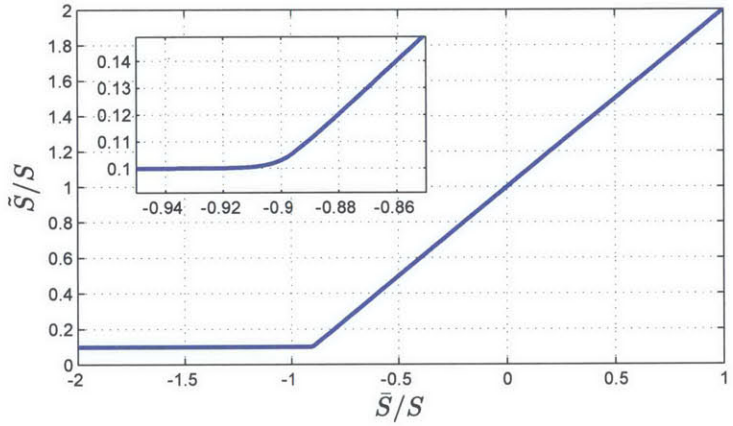

Figure 4-2: Regularization of modified vorticity $\tilde{S}(\bar{S})$

Regularization of r parameter

Finally, the r parameter in the SA model is the source of another numerical failure mode that was observed in unsteady RANS-SA flow experiments with HDG. In the original description of the SA model [94], it was noted that the r parameter can be clipped at a value of 10, as the SA model equations raise r to a large power and this can cause overflow. Moreover, clipping r at a value of 5 or 10 has a negligible impact

on the other model terms – for example, clipping at $r = 5$ only has a $\mathcal{O}(10^{-6})$ impact on f_w.

Rather than simply clipping r, a smooth clipping was implemented to help prevent problems with Newton-Raphson iteration when solving the overall equations. The form of this regularization is provided below and plotted in Figure 4-3.

$$\tilde{r} = r_{\max} - (r_{\max} - r)(\arctan(b(5 - r))/\pi + 1/2) - c \tag{4.7}$$

$$r_{\max} = 5, \quad b = 100, \quad c = 1/2 - \arctan(b)/\pi \tag{4.8}$$

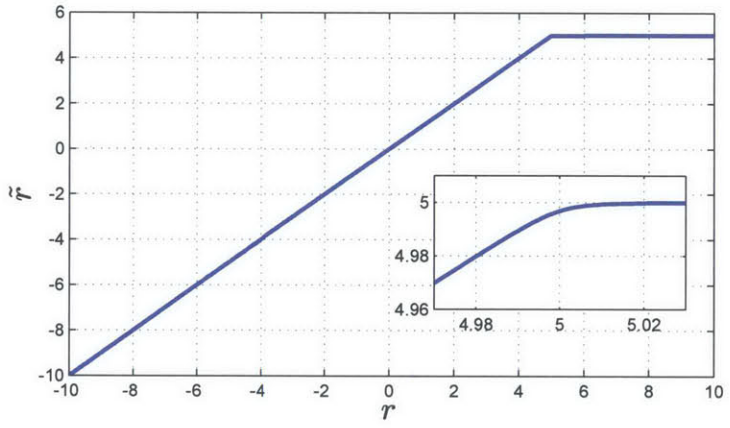

Figure 4-3: Regularization of r parameter, $\tilde{r}(r)$

Summary

Before the three regularizations described above, HDG simulations of unsteady RANS-SA turbulent flows were occasionally observed to "blow up" due to the issues discussed. Since applying these three regularizations to the SA model, numerical stability of the model terms themselves has not been a problem, and for this reason we have seen that regularization of the SA model is important for practical use of high-order methods with the RANS-SA equations.

4.2 Rotor Cascade Flow

In this section we begin the presentation of a series of turbomachinery flow simulations to demonstrate the Time-Spectral or Time-Marching HDG method. First, a steady flow through a single-row rotor cascade.

Geometric and Flow Parameters

The spatial mesh is plotted in Figure 4-4, showing details of the leading edge and the trailing edge. This is a high-order $p = 4$ mesh with 12,330 triangles. The boundary layer is a structured C-mesh, and the outer flow region is an unstructured isotropic mesh. The rotor blade is a NACA 3507 airfoil, at a stagger angle of 60.8°, and with blade pitch to chord length ratio $5\pi/18$. Periodic boundary conditions are employed at the top and bottom boundaries of the blade passage to model the entire cascade.

The Reynolds number of the flow relative to the rotor is $Re_c = 518,000$ and the freestream Mach number is 0.149. At the inlet, the inflow angle is 10° (from axial).

Results and Discussion

Figure 4-5 presents a visualization of the vorticity and pressure fields through this rotor cascade. The wake and boundary layer are well resolved and do not appear unduly dissipative due to the numerical method.

To assess the relative efficiency of a high-order HDG method versus a low-order HDG method (as a proxy for low-order Finite Volume methods), a study was performed to repeat this simulation on a family of similar grids with different resolutions. These simulations were performed with $p = 4$ and $p = 1$, and Figure 4-6 presents the results in terms of solution accuracy and solution total degree of freedom (DOF) count.

What we can see is that the higher-order method is much less expensive at moderate to high levels of solution accuracy, compared to the lower-order method. In the present implementation of the code, the right-most $p = 1$ simulation on the plot took 70×

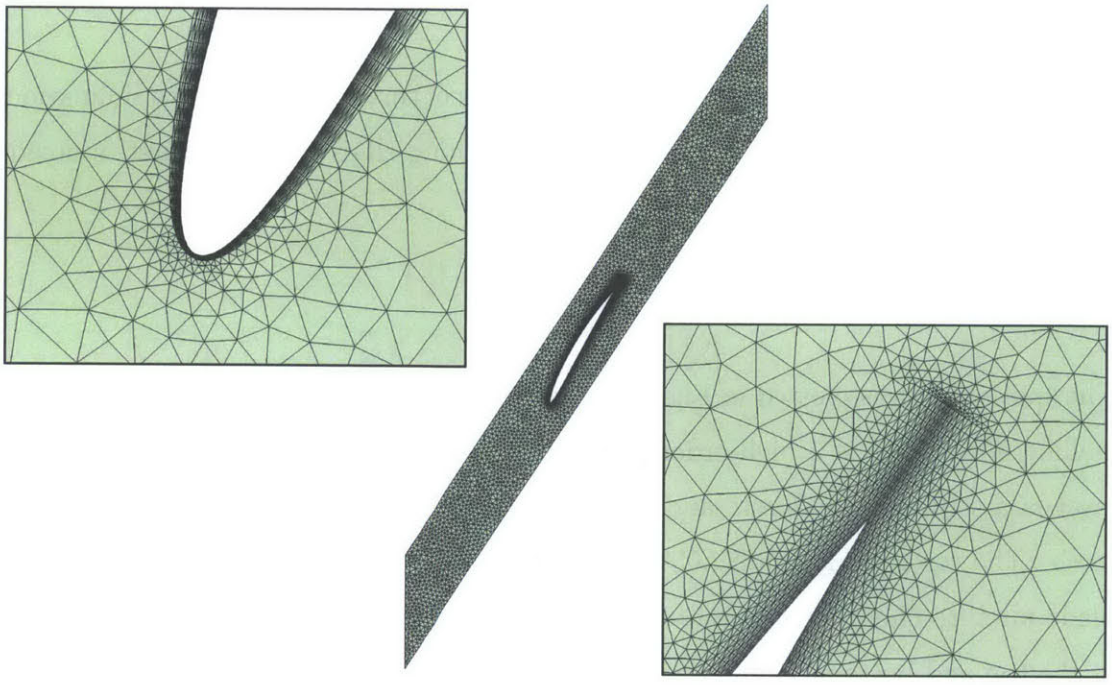

Figure 4-4: Mesh for rotor blade passage, showing detailed views of leading edge and trailing edge.

more CPU time than the $p = 4$ simulation at the same level of accuracy. This is likely explained by the significant numerical dissipation present in lower-order Finite Volume methods, which in this case resulted in a dramatic cost advantage for the higher-order ($p = 4$) HDG method.

4.3 Rotor-Stator Flow

The next flow problem studied here is a rotor/stator flow, with an inlet guide vane (IGV) row followed by a rotor row.

Geometry and Flow Parameters

For this example, the blade pitch of the rotor row has been reduced by half, so that the blade pitch to chord length ratio is now $5\pi/36$. The geometry and motion of the rotor row is otherwise the same as in the previous example. The reduction in

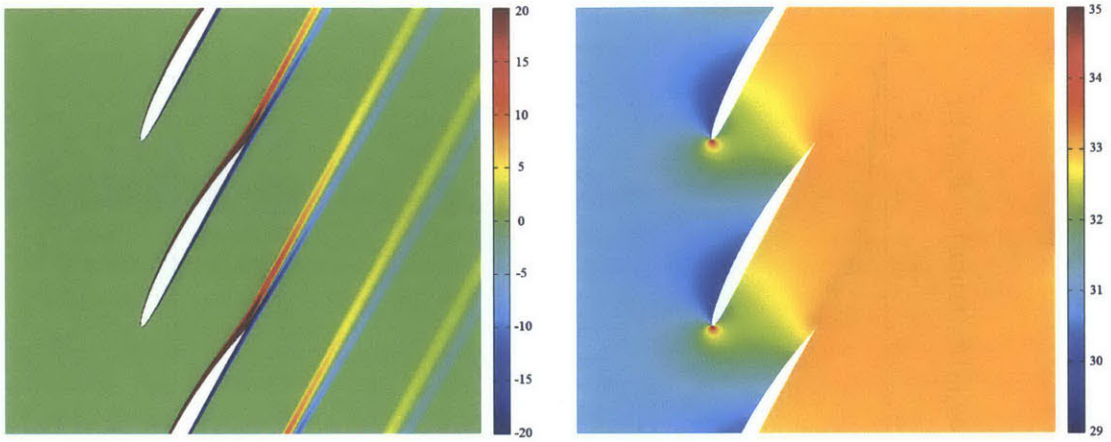

Figure 4-5: High-order ($p = 4$) HDG solution of turbulent flow through a rotor cascade at $Re = 518$k. Flow visualizations above are vorticity (*left*) and pressure (*right*).

blade pitch was chosen to make this example better suited for testing with the Time-Spectral HDG method – as shown in Figure 3-8, the number of modes required to resolve a wake transfer across an interface is directly proportional to the ratio of the blade pitch to wake width, BP/w. So reducing the blade pitch by a factor of 2 allows the Time-Spectral HDG solution to achieve a similar level of accuracy with 2 times fewer modes, which in turn allows us to compute with a larger number of modes than would otherwise fit within the computational memory constraint.

The IGV row has the same blade pitch as the rotor row. The blades are NACA 5507 airfoils with 7.4° stagger angle, chord length $(7/8)c_{\text{rotor}}$, and they are vertically mirrored.

The IGV blade passage mesh has 1,011 $p = 3$ triangles, with a mesh design similar to the design of the rotor mesh in the previous example, featuring a C-mesh boundary layer region and unstructured farfield. The rotor blade passage mesh contains 1,661 $p = 3$ triangles. Rotor speed is the same as in the previous rotor cascade case.

The inflow parameters here are similar to the rotor cascade case, but the rotor Reynolds number was dropped from 518,000 to 259,000 to help widen the wake and further improve BP/w, allowing more frequencies to be resolved within the computational memory constraints arising from our shared memory implementation of the

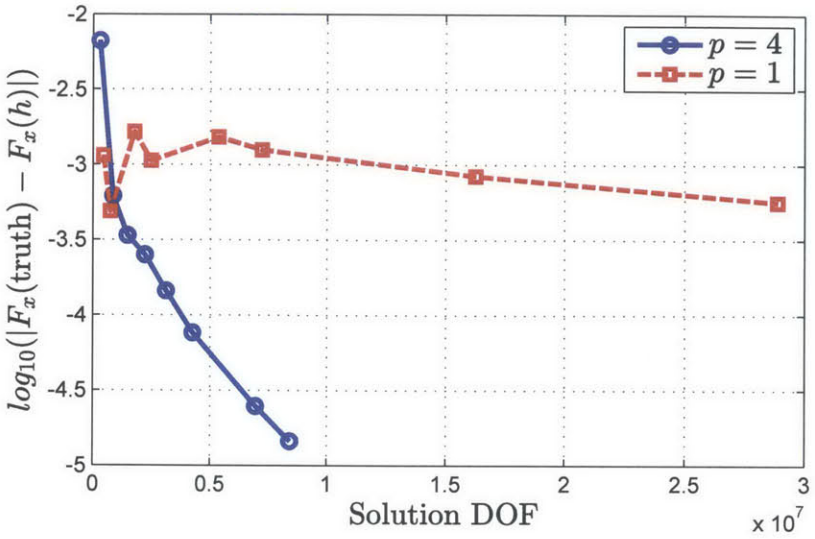

Figure 4-6: Accuracy vs. cost characteristics for rotor flow simulations, comparing low order ($p = 1$) and high order ($p = 4$) in space. The metric used here for cost is the total number of degrees of freedom in the solution (u_h, q_h). Accuracy is measured by the error in the computed x-component of force on the rotor, with respect to a finely resolved "truth" solution on a $p = 5$ grid with 18,175,248 solution degrees of freedom, which is approximately twice the DOF of the finest $p = 4$ result plotted here.

Time-Spectral method. As for the other inlet parameters, the flow angle is $0°$ (from axial), and the inlet Mach number is 0.149.

Results and Discussion

Figure 4-7 presents a set of snapshots of the flow solution at 4 points in the blade-passing period. The flow is visualized by vorticity. On the left side of the Figure are results from an implicit time-marching.HDG simulation, using high-order DIRK(3,3) time-marching with a very fine timestep size ($\Delta t = T/100$). IGV wake propagation across the interface appears to be well resolved, with no noticable errors appearing there due to interface coupling.

On the right side of the Figure is a set of comparable results computed using the Time-Spectral HDG method rather than time-marching. The Time-Spectral computa-

tion used $K = 9$ modes on both blade rows. The flow visualizations are visually nearly identical between time-marching and Time-Spectral, including resolution of the IGV wake transferring across the interface, and the rotor wake propagating downstream. Note that the mesh here is rather coarse, and certainly a lot more coarse than the mesh used to solve the rotor cascade problem in the previous section.

As another point of comparison between the Time-Marching and Time-Spectral solutions, Figure 4-8 and 4-9 present a comparison of the total rotor force timeseries, both in the time domain and the frequency domain. Since the time-marching result was computed with a high-order accurate DIRK(3,3) time-marching scheme, and with rather small time-step size ($\Delta t = T/100$), we have assumed here that the time-marching result is a "truth" result on the given spatial mesh. The Time-Spectral results are then compared against this for accuracy.

In Figure 4-8 we see that the Time-Spectral rotor forces converge to the time-marching truth values rather quickly. With only 5 modes the solution is already looking close in a time-domain sense.

However, what is often of interest to an engineer is the amplitude of a single mode, such as the first harmonic (for reasons associated with structural analysis). Turning to Figure 4-9, we see that the Time-Spectral forces do converge to the Time-Marching results quite rapidly. Of particular note, the solution with only $K = 3$ modes is already within 5% of the correct value for the 1st harmonic of rotor force. If this were the output of interest for an engineer, then a Time-Spectral HDG approach could yield cost savings over time-marching that are even more dramatic than what was seen in the pitching airfoil example in Chapter 2.

4.4 Three-Row Compressor Flow

In this example we compute flow through a three-row compressor, with an IGV row followed by a rotor row and an additional stator row. These computations are performed using a DIRK(3,3) high-order implicit time-marching scheme combined with the HDG method and sliding interface described in this thesis. The flow is computed on three different meshes, with the intent to directly compare the accuracy obtained with a low-order method in space ($p = 1$) versus a high-order method ($p = 4$).

Geometry, Flow and Time-Marching Parameters

Here, IGV and rotor blades are the same as in the previous rotor/stator flow example, however the blade pitch to chord length ratio for the rotor is now back to $5\pi/18$ (as in the original rotor cascade problem), and the blade pitch of the IGV is set to match. The additional stator row is a NACA 5507 airfoil with $-11.3°$ stagger angle, vertically mirrored, and with the same chord length as the IGV. The blade pitch of the stator is the same as the other two blade rows. The rotor Reynolds number is 518,000, the inlet Mach number is 0.149, and the inlet flow angle is $0°$ (from axial).

There are three different meshes considered in this analysis. The first is a low-order ($p = 1$) mesh, with $5,486 + 10,394 + 13,201 = 29,081$ triangles in total. The second is a high-order ($p = 4$) mesh, with $1,091 + 2,090 + 2,328 = 5,509$ triangles total. Finally, there is a "truth" mesh with double the degrees of freedom – this mesh is high-order ($p = 4$) and has $1,932 + 4,040 + 5,080 = 11,052$ triangles in total. For comparison, the total number of spatial degrees of freedom in each of these three meshes is: 1,398,892 for the $p = 1$ mesh, 1,159,960 for the coarser $p = 4$ mesh, and 2,325,760 for the finer $p = 4$ "truth" mesh.

Regarding time-marching, all simulations were performed with a DIRK(3,3) high-order implicit time-marching scheme and timestep size $\Delta t = T/100$.

Results and Discussion

Figure 4-10 presents a snapshot of the time-marching flow solution from the coarser $p = 3$ mesh. The snapshot was taken after 30 blade-passing periods of time integration had elapsed, at which point the flow is very confidently periodic in time. The flow is visualized here by the vorticity field, and we see that wakes are well-resolved behind each blade and that these wakes propagate across the sliding interfaces fairly well.

The simulation shown in Figure 4-10 was repeated for all 3 meshes (coarse $p = 1$, coarse $p = 4$, and fine $p = 4$). Upon reaching periodicity, the spectrum of total rotor force timeseries was measured from each simulation, and these results are plotted in Figure 4-11.

The rotor force spectra presented in Figure 4-11 demonstrate a very interesting result. Recall that the spatial meshes for the coarse $p = 1$ and coarse $p = 4$ cases were purposefully chosen to have approximately the same total number of degrees of freedom. (In fact, the $p = 1$ mesh has approximately 21% more degrees of freedom.) Given the parity in degrees of freedom, it is remarkable that the error in the rotor force spectrum (relative to the finer "truth" solution) is much larger for the $p = 1$ solution than the $p = 4$ solution. In this case, the $p = 4$ result features approximately 1% error in the first harmonic, and the $p = 1$ result features approximately 20% error, despite having 21% more spatial degrees of freedom.

Since the time-marching scheme was kept constant among all three cases, this result is an interesting quantification of the accuracy that can be gained by using a high-order method in space, as opposed to a lower-order method.

4.5 Discussion: Relative Performance of Discretization Methods

We conclude this chapter by briefly summarizing some key results which highlight interesting performance characteristics of the Time-Spectral HDG method applied to turbomachinery flows.

4.5.1 High-Order vs. Low-Order in Space

One sensible practical question to ask about the method presented in this thesis is: what do you gain by changing from a low-order method (such as $p = 1$ Finite Volume) to a high-order method (such as the present $p = 4$ HDG)? In response, two key results from this chapter can be highlighted:

- In Figure 4-6, we saw a comparison of a rotor cascade simulation performed with successively finer meshes, using $p = 1$ and $p = 4$. These results showed that at a moderate or higher level of accuracy, the $p = 4$ simulation required far fewer spatial degrees of freedom than a $p = 1$ simulation with the same accuracy. In the worst case, the penalty associated with a $p = 1$ method was a factor of 35 in degrees of freedom and roughly 70 in CPU time (on the current implementation).

- In Figure 4-11, we compared the rotor force spectrum obtained from a $p = 1$ simulation and a $p = 4$ simulation, relative to a $2\times$ refined $p = 4$ simulation that served as "truth". In this experiment, we saw a substantial improvement in accuracy (from 20% to 1%) when switching from $p = 1$ to $p = 4$, even after giving up 17% of the degrees of freedom included in the $p = 1$ result. This is a clear example of a high-order method delivering a more accurate solution at a more efficient cost.

4.5.2 Time-Spectral vs. Time-Marching

Another salient question to ask about the method presented in this thesis is: what benefits are gained by changing from a time-marching HDG method to a Time-Spectral HDG method? The key to this is presented in Figure 4-9.

Figure 4-9 shows that the rotor force spectrum obtained from Time-Spectral HDG computations is remarkably similar to that obtained by Time-Marching. Specifically, with a cost of simultaneously solving only $N = 7$ snapshots of the flow, we see that we can obtain the amplitudes of the 1st and 2nd harmonics within 2% of the truth solution. The truth solution here was computed with a 3^{rd}-order implicit time-marching scheme (DIRK(3,3)), with 100 timesteps per period and 30 periods of time integration. While this study did not specifically identify a coarser time-marching scheme that achieves the same level of accuracy as the $N = 7$ Time-Spectral solution, these results nonetheless suggest that substantial gains in computational efficiency can be obtained by using a Time-Spectral approach.

In the case of a pitching airfoil flow (Section 2.5.2), the comparison between time-marching and Time-Spectral was seen more precisely. For the low Reynolds number pitching airfoil flow studied, the same level of solution accuracy was obtained from a Time-Spectral HDG solution with $N = 23$ snapshots, and a 2^{nd}-order accurate DIRK(2,2) time-marching scheme with 100 timesteps per period over 6 periods. In the implementation used in this thesis, the Time-Spectral result was obtained in approximately $4\times$ fewer CPU hours than the time-marching result. This is one demonstration of the gains in computational efficiency which may be obtained by using a Time-Spectral approach rather than time-marching.

However, note that this advantage will not be achievable for every type of periodic flow problem. The key condition for Time-Spectral methods to have an advantage is that there should be only a small number of dominant harmonic modes in the flow. The reason for this is that the Time-Spectral HDG method is memory intensive, as all N snapshots of the flow solution must be solved simultaneously. The memory cost of a Time-Spectral solution is therefore $\mathcal{O}(N^2)$ larger than a single timestep of a time-

marching method for the same problem. When a large number of harmonic modes are required, N is large and the requirements of solving the linear system may exceed the system memory available. Thus, the choice of Time-Spectral versus time-marching HDG methods will often be a trade-off between memory cost and total computational effort, with the balance being more in favor of a Time-Spectral approach when the flow can be adequately described by only a few harmonic modes.

One further limitation of the Time-Spectral approach is that it requires the specification of a time period T, which excludes any periodic time variation that does not occur as a harmonic of the fundamental frequency $f_1 = 1/T$. There are many problems where this is not a strong limitation, for example turbomachinery flows with known blade-passing periods and negligible vortex shedding (such as the flow problems studied in this chapter). However, in problems where phenomena such as vortex shedding are significant, it might be difficult to determine the fundamental period T ahead of time. Approaches have been proposed to handle this uncertainty in T [43], but not without an associated computational cost.

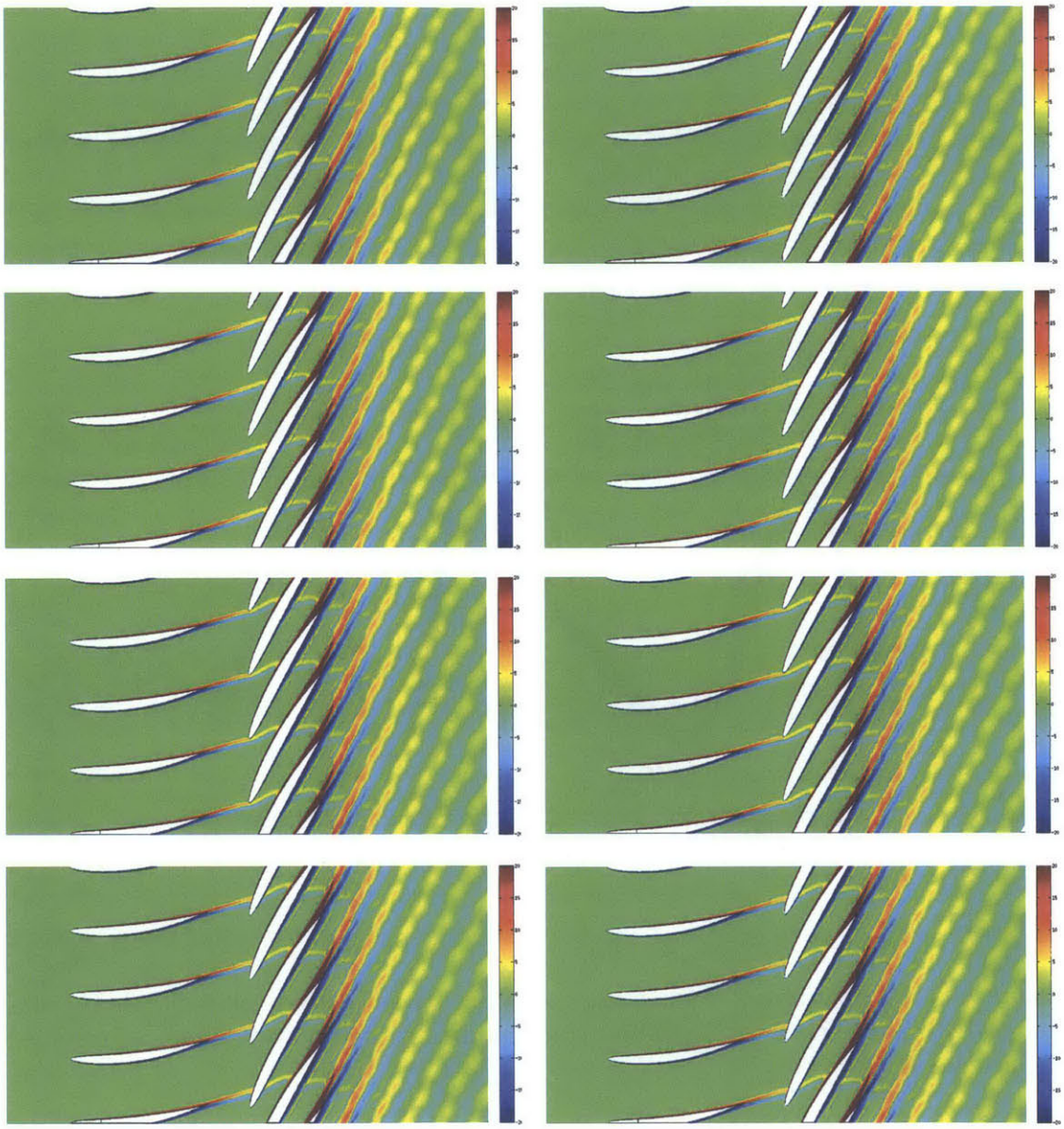

Figure 4-7: Snapshots of rotor/stator flow solution, comparing DIRK(3,3) time-marching with $\Delta t = T/100$ (*left*) and a Time-Spectral solution with $K = 9$ modes resolved (*right*). Vorticity is plotted between limits [-20,20] for 4 snapshots during the blade-passing period T: $t = 0$, $t = T/4$, $t = T/2$, and $t = 3T/4$ (from top to bottom). Both solutions are computed on the same $p = 3$ unstructured spatial mesh.

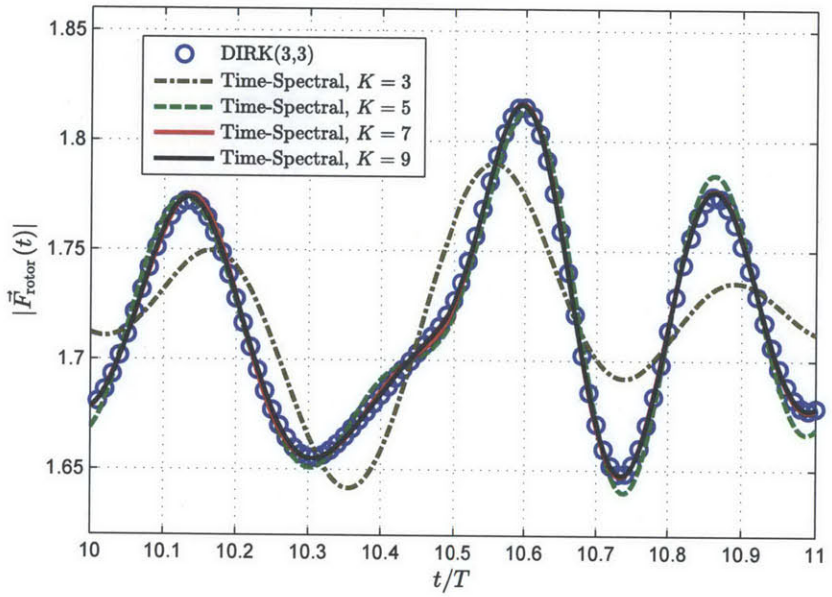

Figure 4-8: Comparison of rotor force timeseries obtained from time-marching solution (DIRK(3,3)) and from Time-Spectral solutions with various numbers of resolved frequencies K.

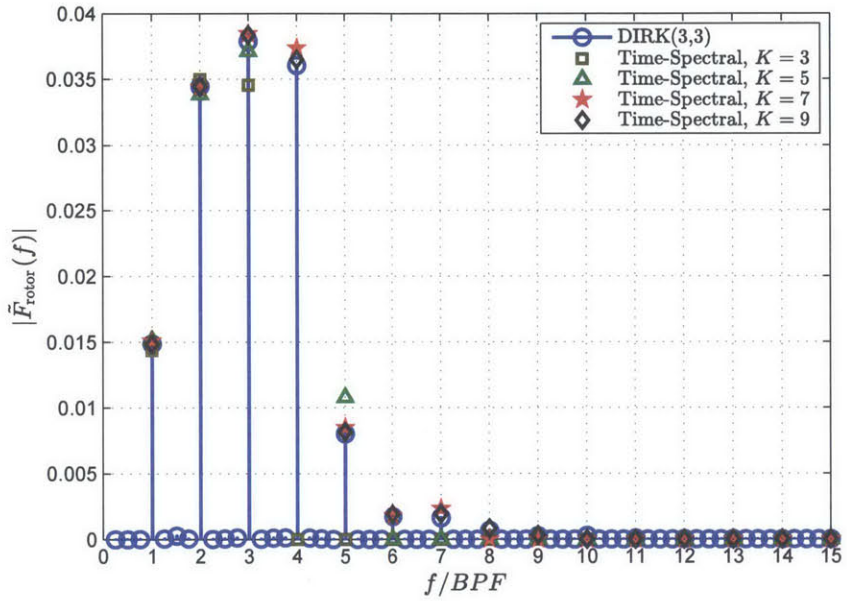

Figure 4-9: Comparison of rotor force spectra obtained from time-marching (DIRK(3,3)) and from Time-Spectral solutions with various numbers of resolved frequencies K.

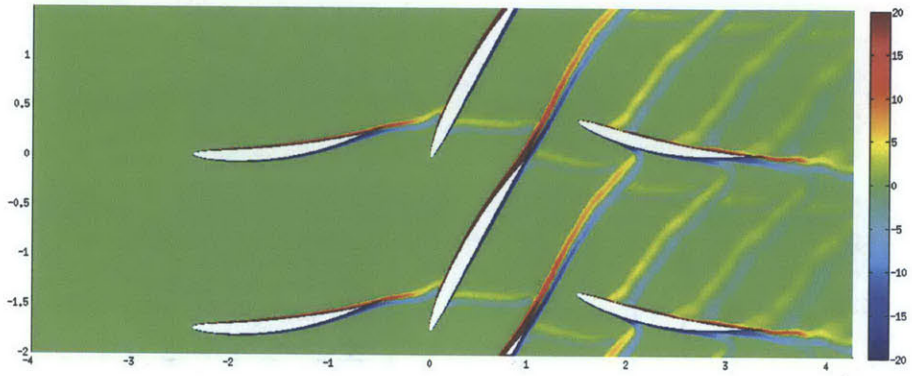

Figure 4-10: Snapshot of high-order flow solution for three-row compressor, visualized by flow vorticity.

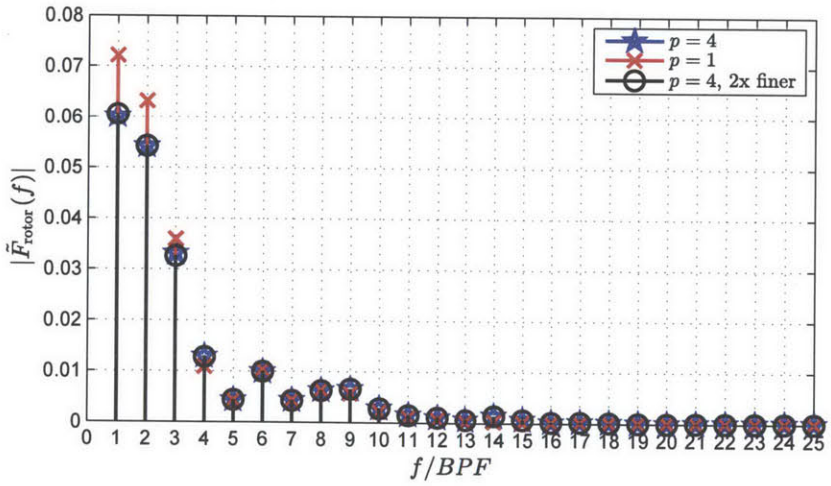

Figure 4-11: Rotor force spectrum, comparing high order solution vs low order solution

Chapter 5

A Frequency-Adaptive Time-Spectral Method

This chapter presents an approach to Time-Spectral computations which extends the method presented in previous chapters, and achieves two new goals:

1. Solve a periodic problem by resolving a particular set of frequencies, rather than all frequencies up to a given maximum

2. Provide an automatic means to guide selection of the most relevant frequencies in the numerical solution

The sections that follow will present a motivation (Section 5.1), then formulation and application to ordinary differential equations (ODEs) (Section 5.2), and finally a formulation and application to Time-Spectral HDG simulations of multi-row turbomachinery flows (Section 5.3).

5.1 Motivation

In most time-periodic flows, there are some frequencies that are more important than others. For example, in the context of industrial turbomachinery flows with multiple blade rows, flow behavior will be dominated by a set of frequencies based on the blade passing frequencies of each row. This leads to one desirable feature of any method for solving turbomachinery flows – that the method should be capable of solving for flow behavior limited to certain pre-selected frequencies, thereby saving unnecessary computational expense. This is an aspect that will be demonstrated in this chapter for Time-Spectral HDG.

Another desirable feature would be to have an automatic way of guiding the user's selection of the most important frequencies. For example, there may be some uncertainty about which frequencies are most important in a turbomachinery flow, making it unclear which frequencies should be resolved first before refining with additional frequencies. Depending on the size of the simulation, it may be quite expensive to resolve a large number of frequencies, and so careful selection of frequencies can be vital. Towards addressing this challenge, this chapter will present a technique for frequency selection based on the Time-Spectral residual.

5.2 Adaptive Time-Spectral Method for Ordinary Differential Equations

5.2.1 Formulation of Error Indicator

To begin, consider a time-periodic ordinary differential equation (ODE) as follows:

$$\frac{du}{dt} + F(u,t) = 0 \tag{5.1}$$

Here, F may be a linear or nonlinear function of u and t which results in a time-periodic solution u. Note that a Time-Spectral discretization of this problem would

be as follows:

$$D_n \boldsymbol{u}_n + \boldsymbol{F}_n(\boldsymbol{u}_n, t_n) = 0 \tag{5.2}$$

where D_n is the $N \times N$ Time-Spectral coupling matrix (Eq. 2.7), $\boldsymbol{u}_n$ is a vector of N snapshots of u, and $\boldsymbol{F}_n$ is a vector of N snapshots of F.

Now return to the original ODE and define a residual r_h:

$$r_h = \frac{du_h}{dt} + F(u_h, t) \tag{5.3}$$

where u_h is the numerical approximation of u.

Note that in the numerical solution from the Time-Spectral method described above, this residual r_h will be driven to zero at the N collocation times t_n, but may be nonzero at other times. We will exploit this feature in the formulation of an inexpensive error indicator.

First, interpolate the Time-Spectral solution $\boldsymbol{u}_n$ to a finer set of M equispaced time-points t_m using the chosen Fourier mode basis (result $\tilde{\boldsymbol{u}}_m$), and compute the residual $\tilde{r}_m$ at times t_m:

$$\tilde{\boldsymbol{u}}_m = \tilde{E}^{-1}(E\boldsymbol{u}_n) \tag{5.4}$$

$$\tilde{\boldsymbol{r}}_m = D_m \tilde{\boldsymbol{u}}_m + \boldsymbol{F}_m(\tilde{\boldsymbol{u}}_m, t_m) \tag{5.5}$$

Finally, evaluate the projection of this residual onto a set of K candidate Fourier modes. For frequency f:

$$\tilde{R}_f = \left| \sum_{m=1}^{M} \tilde{r}_m \cdot \exp(-i2\pi f m/M) \right| \tag{5.6}$$

This amplitude $\tilde{R}_f$ is the approximate error indicator for frequency f. Comparing the error indicator for several different frequencies f will provide useful guidance in selecting which frequencies are most important in the solution, as we will demonstrate in the following sections.

5.2.2 Collocation Points and Coupling Matrix for Arbitrary Frequencies

Collocation Points

When we select frequencies sequentially (e.g. $f_r = \{1, 2, 3, 4\}$) for solving a periodic problem by the Time-Spectral method, the natural choice of temporal collocation points is a uniform distribution of $2K + 1$ points, where K is the total number of frequencies resolved. However, when we select frequencies any other way (e.g. $f_r = \{1, 3, 7, 8\}$), we can no longer use uniform collocation points in time, as this may lead to poor conditioning of the linear system and large numerical errors.

The question of selecting high quality collocation points was addressed in recent work by Guedeney *et al.* [45] for harmonic balance computations. They pointed out that the condition number of the Discrete Fourier Transform (DFT) matrix is a useful metric of quality for a set of collocation points, given a corresponding set of frequencies. In the present thesis, collocation points are chosen by explicitly seeking to minimize the condition number of the DFT matrix.

We may define the Inverse Discrete Fourier Transform (IDFT) matrix as:

$$E^{-1}_{n,k} = \exp(i2\pi f_k t_n / T) \tag{5.7}$$

where $n \in \{0, ..., N-1\}$ is the temporal collocation point index, $k \in \{-K, ..., K\}$ is the frequency index, and T is the fundamental period. Regarding the frequencies, $f_0 = 0$ and $f_{-k} = -f_k$ but K independent frequencies are otherwise free to choose arbitrarily. For example, with $K = 2$ we could have $f = \{-5, -2, 0, 2, 5\}$.

Now, since condition number $\kappa(E^{-1}) = \kappa(E)$, we can write down the optimization statement which describes our optimal collocation points:

$$t^* = \arg\min_t \kappa(E^{-1}(f, t)), \quad \text{s.t. } t \in [0, T]^N \tag{5.8}$$

In the present thesis, a simple SQP optimizer was used to solve this optimization

problem for the optimal collocation points t^* given the resolved frequencies f and the fundamental period T. This approach relies on having good starting guesses for t^*, which here are found via the "magic points" algorithm of Maday *et al.* [66]. The magic points algorithm chooses interpolation point locations in a greedy fashion, identifying the location of the largest interpolation error at each iteration. The resulting points are not optimal, but they have often proven to be good starting points for the optimization routine, and they are very inexpensive to compute.

Coupling Matrix

Now that we have the optimal collocation points t^* defined for a given set of frequencies f, it is simple to define the appropriate Time-Spectral time-derivative coupling matrix:

$$D = \frac{\partial E^{-1}}{\partial t} E \tag{5.9}$$

Here, $E = E(f, t^*)$ is the Discrete Fourier Transform matrix derived by first forming the Inverse Discrete Fourier Transform E^{-1} (Eq. 5.7) and then inverting the result.

The matrix D above can be calculated for any choice of frequencies and associated temporal collocation points, and this will be used to perform "Adaptive Time-Spectral" computations where the choice of frequencies is generally non-sequential.

5.2.3 Results for a Linear Equation

To demonstrate the proposed error indicator and frequency selection methodology, consider a linear ODE:

$$\frac{du}{dt} = 6\cos(2\pi t) + 6\cos(6\pi t), \quad u(0) = 0 \tag{5.10}$$

In our earlier terminology, this is a specific case of the general ODE (Eq. 5.1) with $F(u,t) = -6\cos(2\pi t) - 6\cos(6\pi t)$. Discretizing this equation with the Adaptive

Time-Spectral approach, we can compute solutions and also error indicators (Eq. 5.6) for several different frequencies. The results of this analysis for several different choices of the resolved frequencies f_r are presented in Figure 5-1 below.

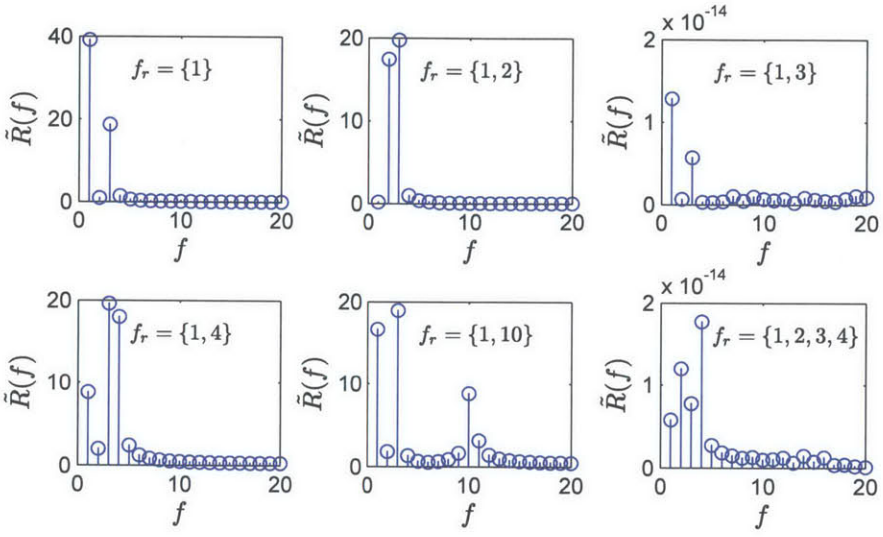

Figure 5-1: Time-Spectral Error Indicator $\tilde{R}$ as a function of frequency f, for several different sets of resolved modes f_r.

The key result shown in Figure 5-1 is that the error indicator always indicates exactly the right frequency, in this linear ODE demonstration. The exact solution should contain only frequencies 1 and 3, and this is correctly identified by the method. Looking at the case in Figure 5-1 with $f_r = \{1\}$, the error indicator $\tilde{R}$ has large values in frequencies 1 and 3, suggesting that the next frequency to choose should be 3 (correctly). Looking at the case with $f_r = \{1,2\}$, the next indicated frequency is also $f = 3$ (correctly). When $f_r = \{1,3\}$, this matches the exact solution and the error indicator is evaluated to be $\mathcal{O}(10^{-14})$ for all frequencies, suggesting that no new frequencies need to be selected (again, correct). Similarly for the cases $f_r = \{1,4\}$, $f_r = \{1,10\}$ and $f_r = \{1,2,3,4\}$, the correct frequency to choose next is always indicated, unless the exact frequencies have already been chosen, in which case the error indicator remains at $\mathcal{O}(10^{-14})$.

In the case of a linear ODE like this, such accuracy in frequency selection is expected, as the residual $\tilde{r}_m$ effectively captures the unresolved portion of the linear source term. The method can be seen to be "exact" in the linear ODE case. The next test is to apply this method to nonlinear equations and see if beneficial properties are still observed.

5.2.4 Results for a Nonlinear Equation

Consider the following nonlinear ODE as a model problem:

$$\frac{du}{dt} + u^3 = 20(\cos(2\pi t) + \cos(8\pi t) + \cos(12\pi t)) \tag{5.11}$$

Much like the example in the previous section, this equation has time-periodic solution, but now we have also introduced a nonlinear term to the equation. Solving this ODE with a very large number of snapshots, we find that the "truth" solution has frequency content as plotted in Figure 5-2. In descending order of amplitude, the most important frequencies appear to be 1,4,6,2,3,8.

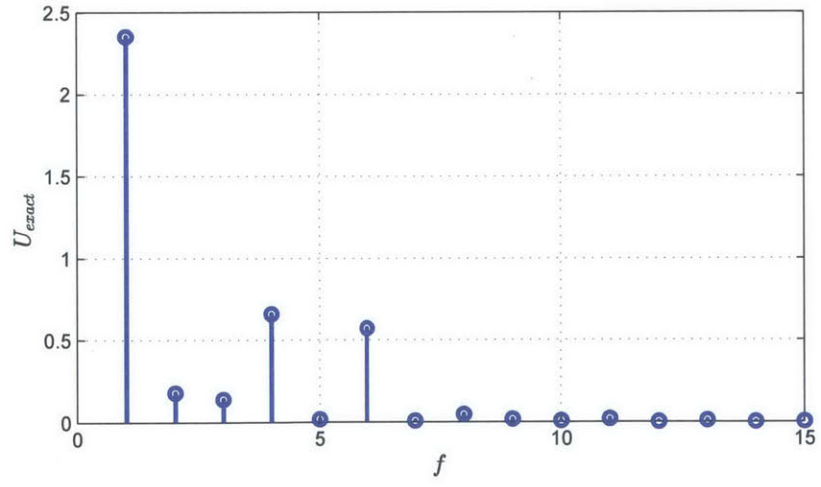

Figure 5-2: Amplitude spectrum of truth solution for nonlinear ordinary differential equation used as a model problem (Eq 5.11)

Discretizing by the Time-Spectral method, we can solve the equation for just the constant mode ($K = 0$) and obtain the error indicator at all candidate frequencies (Figure 5-3). This will suggest a new frequency to be added to the resolved set. After including the new frequency, we can recompute the solution and repeat the process, iteratively extending the resolved set of frequencies f_r.

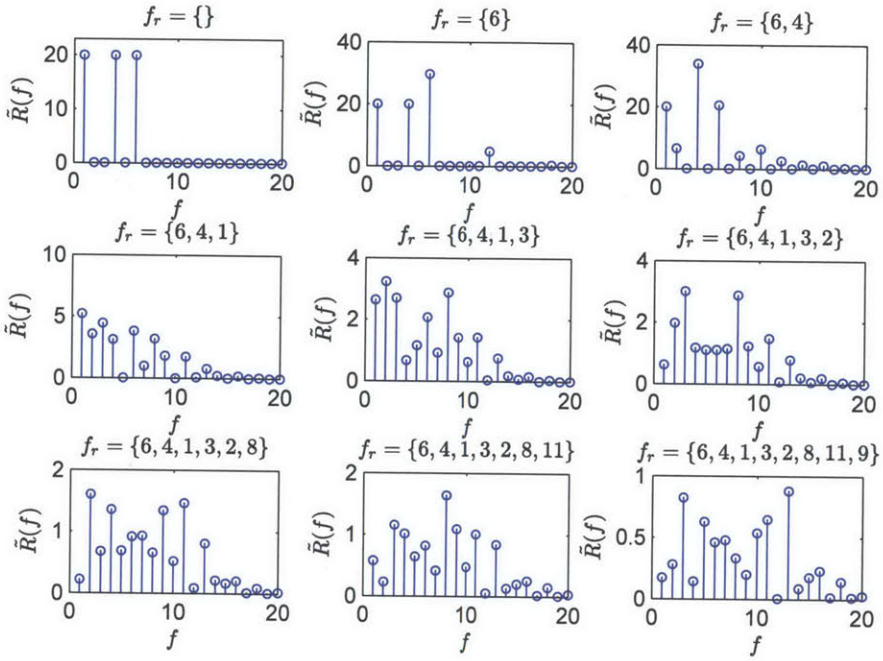

Figure 5-3: Error indicator Nonlinear ODE residual plots

Figure 5-4 plots the improvement in solution accuracy as the number of resolved frequencies is increased. In blue we see the result if frequencies are selected sequentially, i.e. applying the original Time-Spectral method. The line in red shows how solution accuracy changes when we instead choose frequencies adaptively (the Adaptive Time-Spectral approach).

We see in Figure 5-4 a clear advantage of using the Adaptive Time-Spectral approach. In the early stages of the adaptation process, when the number of resolved modes is small, the Adaptive Time-Spectral algorithm led to better choices of the

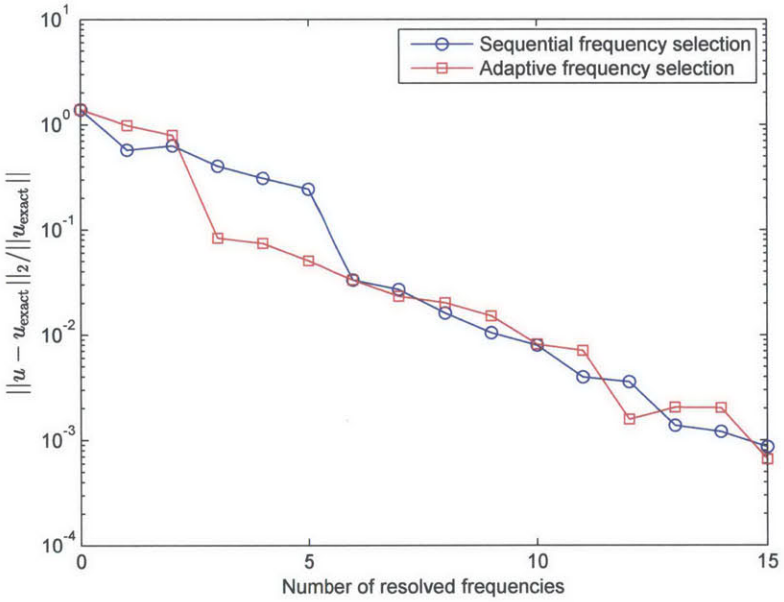

Figure 5-4: Convergence of the numerical solution to a nonlinear ordinary differential equation (Eq 5.11), comparing sequential frequency selection to the proposed adaptive algorithm. Solution error is measured by the RMS error in u.

frequencies to resolve. In fact, the method selected frequencies {6,4,1,3,2} for the first 5 frequencies, which is remarkably similar to the truth solution of {1,4,6,2,3}. In this demonstration, the guidance provided by the error indicator appears to have helped resolve the solution more efficiently than a sequential selection approach, particularly over the first 5 frequencies. This suggests that the Adaptive Time-Spectral approach may be valuable in the nonlinear PDE context as well, such as our RANS-SA turbomachinery flow simulations.

Finally, a note on interpreting the meaning of the error indicator amplitude. For each iteration of the Adaptive Time-Spectral computation, Figure 5-5 plots the maximum value of the time-domain version of the interpolated residual ($\tilde{R}(t)$), versus the actual measured solution error at that iteration of the Adaptive Time-Spectral algorithm. The result is remarkably well-correlated, suggesting that the magnitude of the residual $\tilde{R}$ can serve as a useful proxy for solution accuracy in some cases. It also

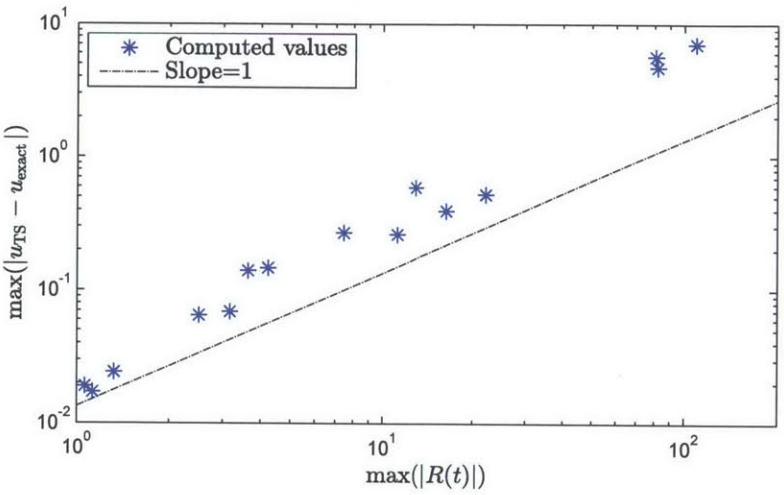

Figure 5-5: Correlation between solution accuracy and Adaptive Time-Spectral error indicator, for the nonlinear ordinary differential equation example (Eq 5.11). Solution accuracy is measured by the maximum difference between the computed and truth solutions u, and this is correlated with the maximum absolute value of the error indicator $R(t)$.

implies that working towards decreasing the residual will also result in a decrease in solution error.

5.3 Adaptive Time-Spectral HDG Method

5.3.1 Formulation

We now extend the method presented in the previous section to Time-Spectral HDG discretizations of time-periodic problems. The residual quantity $\tilde{\boldsymbol{r}}$ (Equation 5.5) will be adapted for the new context of a partial differential equation solved by a Time-Spectral HDG method.

The Time-Spectral HDG equivalent of the residual comes from first interpolating the solution quantities to a larger number of time points (see Eq. 3.12), then

computing the following functional:

$$\boldsymbol{R_u} = (D\boldsymbol{u}_h, w)_{\mathcal{T}_h} - (\boldsymbol{F}(\boldsymbol{u}_h, \boldsymbol{q}_h), \nabla w)_{\mathcal{T}_h} + \langle \widehat{\boldsymbol{F}}(\boldsymbol{u}_h, \boldsymbol{q}_h, \hat{\boldsymbol{u}}_h) \cdot n, w \rangle_{\partial\mathcal{T}_h} \tag{5.12}$$

which can then be converted to a function form by solving the following equation to obtain the residual $\boldsymbol{r}_h$:

$$(\boldsymbol{r}_h, w)_{\mathcal{T}_h} = \boldsymbol{R_u} \tag{5.13}$$

This residual (a function of time) will be defined at every spatial degree of freedom in the problem, though what we actually require is a global measure of this. For frequency selection purposes, an effective choice seems to be the maximum residual amplitude value at each candidate frequency, across all spatial degrees of freedom of the problem:

$$\tilde{R}_f = \max_{i,c,e} \left| \sum_{m=1}^{M} \tilde{r}_{h,m}(i,c,e) \cdot \exp(-i2\pi f m/M) \right| \tag{5.14}$$

where i, c, e are the indices of the spatial node, flow component and element number respectively, and m is the time snapshot index. Equation 5.14 is the Adaptive Time-Spectral error indicator we will use for partial differential equations.

Note that the rationale for this maximization is that all governing equations are nondimensionalized to $\mathcal{O}(1)$, so comparing values across all spatial nodes and flow components should be a fair comparison.

5.3.2 Results for a Rotor-Stator Flow Problem

To demonstrate the Adaptive Time-Spectral HDG method introduced in the previous section, we will apply the method to compute flow through a rotor/stator assembly with a 2:1 blade count ratio. The useful property of this flow example is that the total force on the rotor should contain only even frequencies, whereas the forces on the stator should contain both even and odd frequencies. This flow problem is therefore a good test of the Adaptive Time-Spectral HDG method, because an ideal frequency-selection algorithm should choose only even frequencies on the rotor subdomain.

Geometry, Flow and Time Discretization Parameters

The geometry of this flow problem can be seen in Figure 5-6. Details of the geometry, flow, mesh and time discretization parameters are as follows:

- Inlet Guide Vane (IGV) row:
  - NACA 5507 airfoil, 7.4° stagger angle, flipped vertically
  - Blade pitch: $BP_{\text{IGV}} = BP_{\text{rotor}}/2$
  - Chord length: $c_{\text{IGV}} = (7/8)c_{\text{rotor}}$
- Rotor row:
  - NACA 3507 airfoil, 60.8° stagger angle
  - Blade pitch to chord length ratio = $5\pi/18$
- Flow:
  - Rotor Reynolds number = 259,000
  - Freestream Mach number = 0.149
  - Inlet flow angle = 0° (from axial)
- Mesh: $p = 4$, with a total of $1,824 + 2,315 = 4,139$ triangles.
- Time discretization: limit IGV to $K_1 = 5$ modes; rotor unlimited.

Results

The flow problem described above was solved using Adaptive Time-Spectral HDG and implicit time-marching HDG, and these results are presented in Figure 5-6. On the left is the result from time-marching, and on the right is the result from the 7th iteration of the Adaptive Time-Spectral HDG method. At this iteration, the Adaptive Time-Spectral HDG solution was computed with frequencies $f_1/BPF_2 = \{1, 2, 3, 4, 5\}$ on the 1st row and $f_2/BPF_2 = \{2, 4, 6, 8, 10, 12, 14\}$ on the 2nd row, where BPF_2 is

the blade-passing frequency of the rotor row. These frequencies were automatically identified by the Adaptive Time-Spectral HDG method, and correctly so, beginning from 1st iteration values of $f_1/BPF_2 = \{1\}$ and $f_2/BPF_2 = \{2\}$. The optimal choice was to select frequencies sequentially on the IGV, and only even frequencies on the rotor, and this is exactly what the Adaptive Time-Spectral error indicator proceeded to do (see Figure 5-7).

Looking at the flow solution itself, we can see that there is visually very little difference between the time-marching and Adaptive Time-Spectral solutions. However, there is a large difference in cost. In this implementation of the method, the time-marching result (26 periods with 100 DIRK(3,3) timesteps per period) was computed in 133 hours, and the Adaptive Time-Spectral result (7 iterations) was computed in just 39 hours – saving more than a factor of 3 in computational expense.

If we examine the rotor forces more carefully, in Figure 5-8 we see that the Adaptive Time-Spectral HDG results converge to the time-marching HDG results quite quickly. Moreover, examining the spectrum of the computed rotor force in Figure 5-9, we see again that the Adaptive Time-Spectral results converge rapidly to the time-marching results, but we also see that the amplitude of the 1st harmonic converges even more rapidly than the overall rotor force. In fact, with only 1 frequency resolved on each subdomain ("ATS step 1"), the rotor force amplitude at the lowest frequency is already within 1% of the correct converged value. Therefore, if an engineer's output of interest happens to be only related to phenomena occurring at low frequency (such as the amplitude of the 1st harmonic of the rotor force), the cost savings arising from a Time-Spectral approach (Adaptive or otherwise) will be even greater than the factor of 3 we have previously identified. In this implementation, the CPU time savings would be a factor of almost 200 over the full time-marching HDG result.

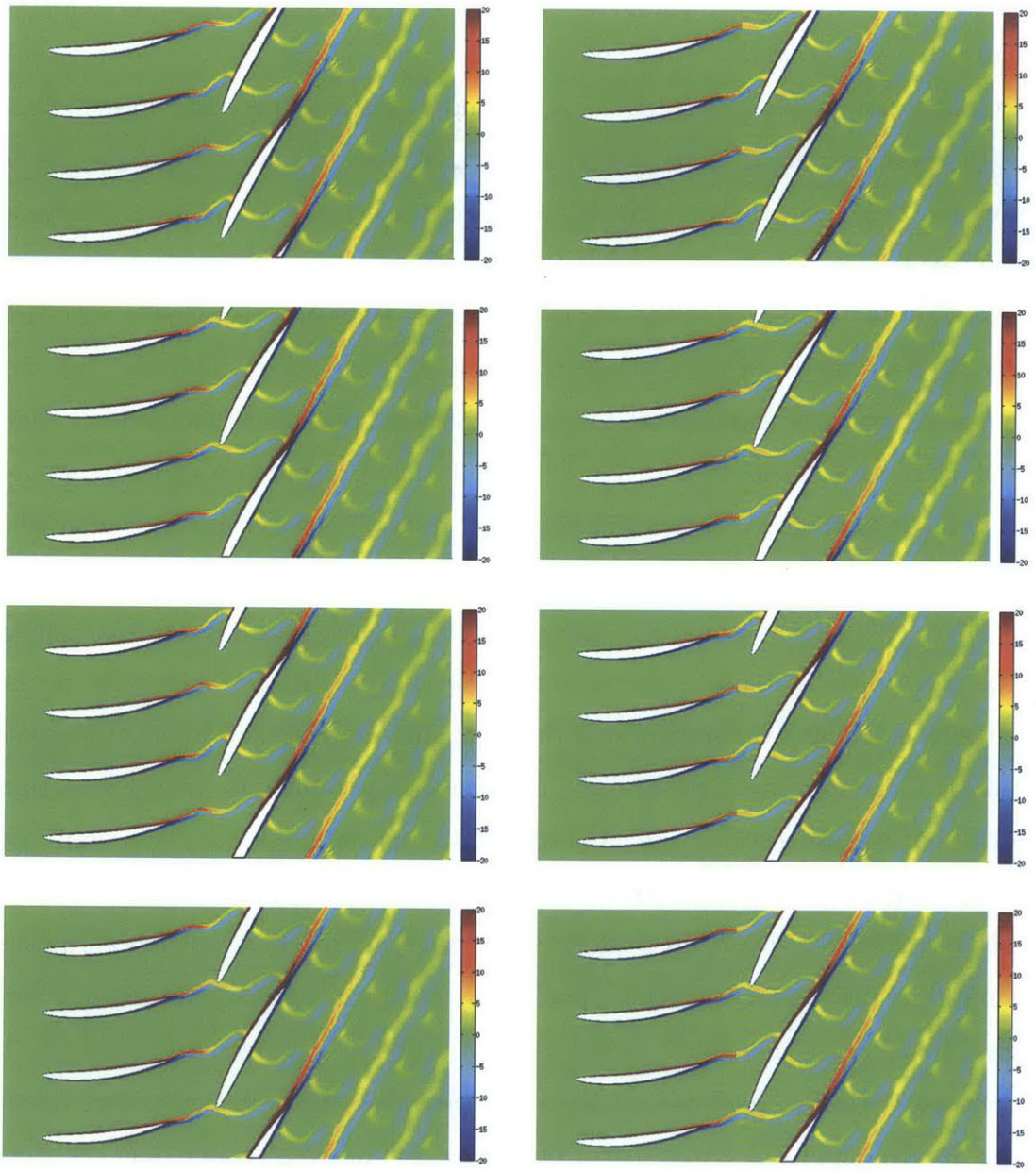

Figure 5-6: Snapshots of rotor/stator flow solution, comparing DIRK(3,3) time-marching with $\Delta t = T/100$ (*left*) and an Adaptive Time-Spectral solution with $K_1 = 5$ and $K_2 = 7$ modes resolved on the 1st and 2nd blade rows respectively. Frequencies were selected automatically using the Adaptive Time-Spectral algorithm, resulting in modes $f_1/BPF_2 = \{1, 2, 3, 4, 5\}$ on the 1st row and $f_2/BPF_2 = \{2, 4, 6, 8, 10, 12, 14\}$ on the 2nd row, where BPF_2 is the blade-passing frequency for the rotor row. Vorticity is plotted for 4 stages of the rotor blade-passing period T: $t = 0$, $t = T/4$, $t = T/2$, and $t = 3T/4$ (from top to bottom). Both solutions are computed on the same p = 4 unstructured spatial mesh.

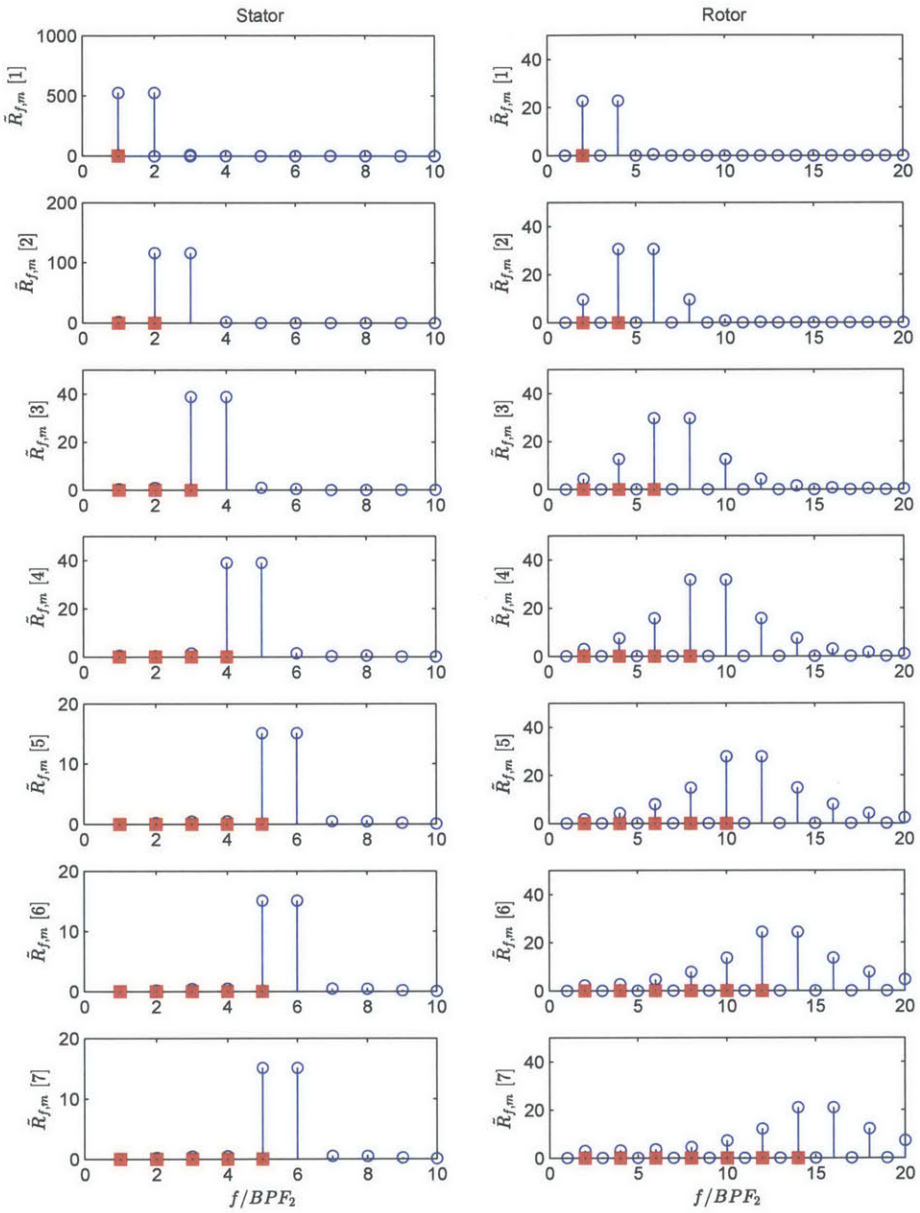

Figure 5-7: Error indicator $\tilde{R}_{f,m}(f)$ for 7 iterations of the Adaptive Time-Spectral rotor/stator flow solution, shown for each blade row (stator on the left, rotor on the right). Red squares identify the frequencies selected by the ATS algorithm at each iteration.

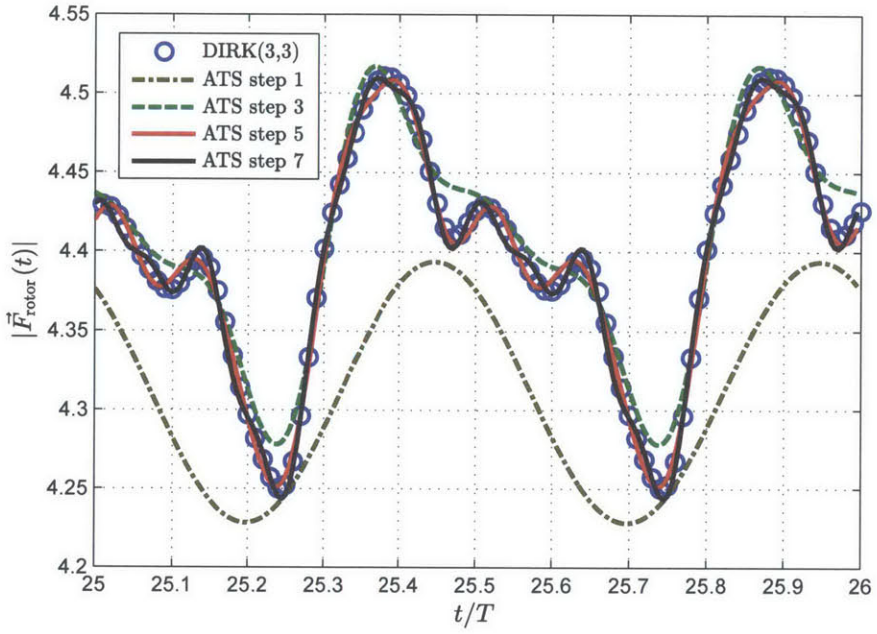

Figure 5-8: Convergence of rotor force timeseries from Adaptive Time-Spectral HDG results vs. a comparable time-marching HDG result.

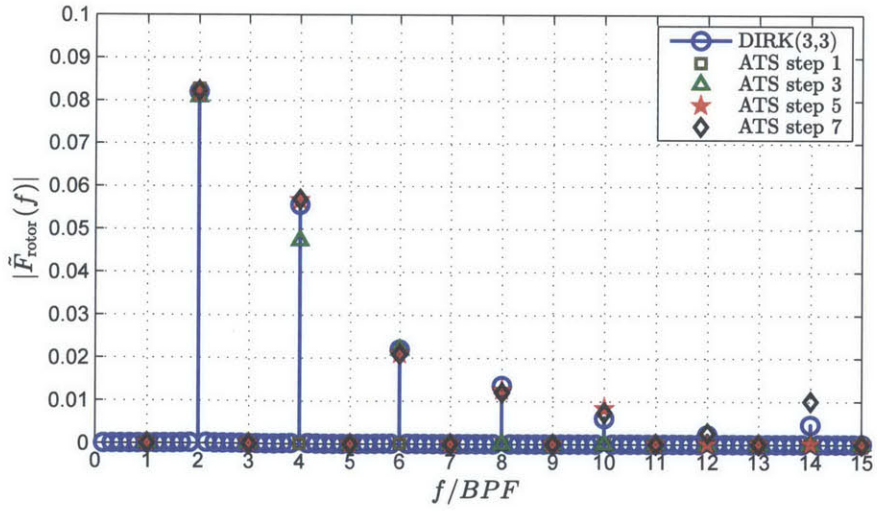

Figure 5-9: Convergence of rotor force spectrum from Adaptive Time-Spectral HDG results vs. rotor force spectrum from a comparable time-marching HDG result.

Chapter 6

Conclusions

6.1 Summary of Findings

This thesis has presented a novel Time-Spectral Hybridizable Discontinuous Galerkin (HDG) method for periodic flow problems, with applications to flow through cascades and rotor/stator assemblies in aeronautical turbomachinery. The present work combines a Fourier-based Time-Spectral discretization in time with an HDG discretization in space, realizing the dual benefits of spectral accuracy in time and high-order accuracy in space. Low numerical dissipation and favorable stability properties are inherited from the high-order HDG method, together with a reduced number of globally coupled degrees of freedom compared to other DG methods. HDG provides a natural framework for treating boundary conditions, which is exploited in the development of a new high-order sliding mesh interface coupling technique for multiple-row turbomachinery problems. A regularization of the Spalart-Allmaras turbulence model has also been employed to ensure numerical stability of unsteady flow solutions obtained with high-order methods.

Turning to the temporal discretization, the Time-Spectral method enables direct solution of a periodic flow state, bypasses initial transient behavior, and can often deliver substantial savings in computational cost compared to implicit time-marching. An important driver of computational efficiency is the ability to select and resolve

only the most important frequencies of a periodic problem, such as the blade-passing frequencies in turbomachinery flows. To this end, the present work has developed an adaptive frequency selection technique, using the Time-Spectral residual to form an inexpensive error indicator. Having selected a set of frequencies, the accuracy of the Time-Spectral solution is greatly improved by using optimally selected collocation points in time. For multi-domain problems such as turbomachinery flows, an anti-aliasing filter is also needed to avoid errors in the transfer of the solution across the sliding interface. All of these aspects contribute to the Adaptive Time-Spectral HDG method developed in this thesis.

Performance characteristics of the method were demonstrated through applications to periodic ordinary differential equations, a convection problem, laminar flow over a pitching airfoil, and turbulent flow through a range of single- and multiple-row turbomachinery configurations. In particular, the following results highlight the advantages of the method:

- Solving a steady rotor cascade flow problem, a $p = 4$ high-order method was shown to be up to $70\times$ less expensive than a $p = 1$ low-order method (with the same convergence rate as a typical Finite Volume method), when compared at the same level of accuracy. (Figure 4-6)

- Solving a rotor/stator flow problem, a Time-Spectral HDG approach results in rotor forces that converge to the same values obtained from a more expensive implicit time-marching HDG approach. (Figure 4-9)

- Solving a three-row compressor flow problem, a $p = 4$ high-order HDG method with implicit time-marching had 1% error in the 1st harmonic of rotor loading, relative to a truth solution, while a $p = 1$ method with 21% more spatial degrees of freedom was actually much less accurate at 20% error. This demonstrates that spatially high-order methods can be more accurate than low-order methods, per unit computational cost. (Figure 4-11)

- Applying the Adaptive Time-Spectral method to a nonlinear ordinary differential

equation, the error indicator automatically selects frequencies in approximately the correct manner. (Section 5.2.4)

- Solving a rotor/stator flow with 2:1 blade count ratio, the Adaptive Time-Spectral HDG method converges to the correct rotor forces, and correctly selects the (even) frequencies required on the rotor subdomain, incurring a computational cost roughly $3\times$ lower than a comparable implicit time-marching result.

- For an engineering output of interest which is simply the rotor force amplitude of the first harmonic, the Adaptive Time-Spectral HDG method converges to within 1% of the true value using only 1 resolved frequency on each subdomain. This results in computational cost savings of a factor of roughly 200, over a comparable time-marching result. (Section 5.3.2)

Together, these observations highlight the fact that the combination of a Time-Spectral discretization in time and a high-order HDG discretization in space can result in substantial savings in computational cost, and/or substantial gains in solution accuracy, when compared to more conventional state of the art methods such as time-marching and 2nd-order Finite Volume. In particular, the benefits of high-order HDG will be realized when the desired level of solution accuracy is at least moderately high, and the benefits of a Time-Spectral approach will be realized when the flow physics of interest can be captured in a relatively small number of harmonic modes.

6.2 Future Work

Building upon the foundation laid by the present thesis, there are a few potential directions for future work that can be suggested:

- Extension of the method to 3D flow problems, together with a parallel implementation of the code to enable the solution of larger problems. A parallel 3D implementation of the HDG method has recently been developed for compressible flow by Roca *et. al.* [87], and published results indicate favorable weak

scaling characteristics for low Reynolds number flow problems. Extension of the Time-Spectral HDG method and RANS-SA model to this parallel implementation will enable 3D turbomachinery flow simulations with strong relevance to industrial applications. We expect our main conclusions regarding efficiency of the Time-Spectral method to apply in 3D just as well as they do in 2D, though memory constraints will become a more significant factor with the larger size of 3D problems.

A parallel implementation will also enable 2D studies of much larger turbomachinery problems, with more blade rows and more resolved harmonic modes. Such simulations may reveal interactions between distant blade rows, which may be more easily captured by a high-order HDG method rather than lower-order Finite Volume methods, due to the lower numerical dissipation of high-order HDG.

- Development of a phase-lagged boundary condition for the top and bottom boundaries of each blade passage will allow studies of turbomachinery flow problems with more general blade count ratios. The form of the phase-lagged boundary condition could be similar to that proposed by Gopinath *et. al.* [41], adapted to the HDG setting. Rather than blade count ratios of 1:1 or 2:1, phase-lagged boundary conditions will allow simulations with ratios such as 18:19, which are more typical in industrial designs. Aside from increased relevance to industry, these studies may yield interesting flow behavior due to the slight differences in fundamental flow frequency between blade rows.

- Extension of the method to transonic turbomachinery flow problems with shocks. This will require implementation of a shock-capturing method, for which two approaches have already been developed in the HDG context by Nguyen *et. al.* [71, 78]. Applying the Time-Spectral HDG method to transonic turbomachinery flows with shocks will be a new and stringent test of the method, as shocks will introduce additional sharp features to the flow that must be transferred

across the sliding interfaces between blade rows and also resolved with temporal harmonic modes. We expect this to increase the number of harmonic modes required to adequately resolve the time-varying flowfield, but what remains to be seen is exactly how costly it will be to meet a desired level of accuracy for an industrial output of interest. The results in the present thesis suggest the possibility that key outputs of interest may converge much faster than the overall flowfield, but this must now be tested for flow problems with shocks.

- Adjoint-based adaptation of the spatial mesh could be applied to the Time-Spectral HDG method. The Time-Spectral HDG method essentially transforms an unsteady flow problem into a larger steady problem, with the entire period of time-variation determined simultaneously, and this makes the problem amenable to adjoint-based adaptation. (As opposed to a time-marching approach, for which adjoint-based adaptation is significantly more complex.) While adjoint-based adaptation is yet to be applied to HDG methods, it has been a significant area of research in the broader context of Finite Volume and Discontinuous Galerkin methods. For example, the Dual Weighted Residual error estimate [9, 10] has been applied in several adaptation strategies [52, 101, 102, 113]. These existing techniques could be applied to the HDG context to develop a spatially adaptive Time-Spectral HDG method. We may expect this method to be capable of achieving a high level of solution accuracy in an even more efficient manner than could be obtained without adaptation, at the cost of additional implementation complexity.

Bibliography

[1] R. Alexander. Diagonally implicit Runge-Kutta methods for stiff O.D.E.'s. *SIAM Journal on Numerical Analysis*, 14(6):1006–1021, 1977.

[2] S.R. Allmaras, F.T. Johnson, and P.R. Spalart. Modifications and clarifications for the implementation of the spalart-allmaras turbulence model. In *Seventh International Conference on Computational Fluid Dynamics (ICCFD7)*, Big Island, Hawaii, July 2012. Paper ICCFD7-1902.

[3] I. Babuska and M. Suri. The P and H-P versions of the finite element method, basic principles and properties. *SIAM Review*, 36(4):578–632, 1994.

[4] I. Babuska, B.A. Szabo, and I.N. Katz. The p-version of the finite element method. *SIAM Journal of Numerical Analysis*, 18(3):515–545, 1981.

[5] B.S. Baldwin and H. Lomax. Thin layer approximation and algebraic model for separated turbulent flows. In *16th AIAA Aerospace Sciences Meeting*, Huntsville, Alabama, January 1978. AIAA Paper 78-257.

[6] T. Barth and M. Ohlberger. Finite volume methods: foundation and analysis. *Encyclopedia of computational mechanics*, 2004.

[7] T.J. Barth and P.O. Frederickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. In *28th Aerospace Sciences Meeting*, Reno, Nevada, January 1990. AIAA Paper 90-0013.

[8] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. *Journal of Computational Physics*, 131(2):267–279, 1997.

[9] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *East West Journal of numerical mathematics*, 4:237–264, 1996.

[10] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica 2001*, 10:1–102, 2001.

[11] C. Breard. Acoustic propagation and radiation modeling of lined duct with linear and non-linear frequency-domain solver. In *9th AIAA/CEAS Aeroacoustics Conference and Exhibit*, Hilton Head, South Carolina, May 2003. AIAA Paper 2003-3265.

[12] A. Buffa, I. Perugia, and T. Warburton. The mortar-discontinuous Galerkin method for the 2d Maxwell eigenproblem. *Journal of Scientific Computing*, 40(1):86–114, 2009.

[13] N. Butsuntorn and A. Jameson. Time spectral method for rotorcraft flow. In *46th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2008. AIAA Paper 2008-0403.

[14] H.K. Chaurasia, N.C. Nguyen, and J. Peraire. A time-spectral hybridizable discontinuous Galerkin method for periodic flow problems. In *21st AIAA Computational Fluid Dynamics Conference*, San Diego, CA, June 2013. AIAA Paper 2013-2861.

[15] G. Chavent and B. Cockburn. The local projection p^0-p^1-discontinuous Galerkin finite element method for scalar conservation laws. *Mathematical Modelling and Numerical Analysis*, 23(4):565–592, 1989.

[16] T. Chen, P. Vasanthakumar, and L. He. Analysis of unsteady blade row interaction using nonlinear harmonic approach. *Journal of Propulsion and Power*, 17(3):651–658, 2001.

[17] Y.S. Chen. 3-D stator-rotor interaction of the SSME. In *AIAA/ASME/SAE/ASEE 24th Joint Propulsion Conference*, Boston, Massachusetts, July 1988. AIAA Paper 88-3095.

[18] S. Choi, J. Alonso, E. Weide, and J. Sitaraman. Validation study of aerodynamic analysis tools for design optimization of helicopter rotors. In *25th AIAA Applied Aerodynamics Conference*, Miami, FL, June 2007. AIAA Paper 2007-3929.

[19] W.S. Clark and K.C. Hall. A time-linearized Navier-Stokes analysis of stall flutter. *Journal of Turbomachinery*, 122:467–476, 2000.

[20] B. Cockburn, B. Dong, and J. Guzmán. A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems. *Mathematics of Computation*, 77(264):1887–1916, 2008.

[21] B. Cockburn, B. Dong, J. Guzmán, M. Restelli, and R. Sacco. A hybridizable discontinuous Galerkin method for steady-state convection-diffusion-reaction problems. *SIAM Journal on Scientific Computing*, 31(5):3827–3846, 2009.

[22] B. Cockburn and J. Gopalakrishnan. A characterization of hybridized mixed methods for second order elliptic problems. *SIAM Journal on Numerical Analysis*, 42(1):283–301, 2004.

[23] B. Cockburn and J. Gopalakrishnan. The derivation of hybridizable discontinuous Galerkin methods for Stokes flow. *SIAM Journal on Numerical Analysis*, 47(2):1092–1125, 2009.

[24] B. Cockburn, J. Gopalakrishnan, and R. Lazarov. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. *SIAM Journal on Numerical Analysis*, 47(2):1319–1365, 2009.

[25] B. Cockburn, J. Gopalakrishnan, N.-C. Nguyen, J. Peraire, and F.J. Sayas. Analysis of HDG methods for Stokes flow. *Mathematics of Computation*, 80(274):723, 2010.

[26] B. Cockburn, J. Gopalakrishnan, and F.J. Sayas. A projection-based error analysis of HDG methods. *Mathematics of Computation*, 79(271):1351, 2010.

[27] B. Cockburn, J. Guzmán, S.C. Soon, and H.K. Stolarski. An analysis of the embedded discontinuous Galerkin method for second-order elliptic problems. *SIAM Journal on Numerical Analysis*, 47(4):2686–2707, 2009.

[28] B. Cockburn, J. Guzmán, and H. Wang. Superconvergent discontinuous Galerkin methods for second-order elliptic problems. *Mathematics of Computation*, 78(265):1–24, 2009.

[29] B. Cockburn and C.-W. Shu. The Runge-Kutta local projection p^1-discontinuous Galerkin method for scalar conservation laws. *RAIRE Model. Math. Anal. Nume.*, 25:337–361, 1991.

[30] B. Cockburn and C.-W. Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.*, 35(6):2440–2463, 1998.

[31] B. Cockburn and C.-W. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3):173–261, 2001.

[32] C.H. Custer, J.M. Weiss, V. Subramanian, K.C. Hall, and W.S. Clark. Unsteady simulation of a 1.5 stage turbine using an implicitly coupled nonlinear harmonic balance method. In *ASME Turbo Expo 2012*, Copenhagen, Denmark, June 2012. Paper GT2012-69690.

[33] K. Ekici and K.C. Hall. Nonlinear analysis of unsteady flows in multistage turbomachines using the harmonic balance technique. In *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2006. AIAA Paper 2006-422.

[34] K. Ekici and K.C. Hall. Nonlinear analysis of unsteady flows in multistage turbomachines using harmonic balance. *AIAA Journal*, 45:1047–1057, 2007.

[35] K. Ekici and K.C. Hall. Nonlinear analysis of unsteady flows in multistage turbomachines using harmonic balance. *AIAA Journal*, 45(5):1047–1057, 2007.

[36] K. Ekici and K.C. Hall. Nonlinear frequency-domain analysis of unsteady flows in turbomachinery with multiple excitation frequencies. *AIAA Journal*, 46(8):1912–1920, 2008.

[37] K. Ekici, K.C. Hall, and E.H. Dowell. Computationally fast harmonic balance methods for unsteady aerodynamic predictions of helicopter rotors. *Journal of Computational Physics*, 227(1):6206–6225, 2008.

[38] K. Ekici, K.C. Hall, and R.E. Kielb. Harmonic balance analysis of blade row interactions in a transonic compressor. *Journal of Propulsion and Power*, 26(2):335–343, 2010.

[39] E. Ferrer and R.H.J. Willden. A high order discontinuous galerkin fourier incompressible 3d navierstokes solver with rotating sliding meshes. *Journal of Computational Physics*, 231(21):7037 – 7056, 2012.

[40] M.B. Giles. Calculation of unsteady wake/rotor interaction. *J. Propulsion*, 4(4):356–362, 1988.

[41] A.K. Gopinath. *Efficient Fourier-Based Algorithms for Time-Periodic Unsteady Problems*. PhD thesis, Stanford University, 2007.

[42] A.K. Gopinath and A. Jameson. Time spectral method for periodic unsteady computations over two- and three- dimensional bodies. In *43rd Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2005. AIAA Paper 2005-1220.

[43] A.K. Gopinath and A. Jameson. Application of the time spectral method to periodic unsteady vortex shedding. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2006. AIAA Paper 2006-0449.

[44] A.K. Gopinath, E. Van Der Weide, J.J. Alonso, A. Jameson, K. Ekici, and K.C. Hall. Three–dimensional unsteady multi–stage turbomachinery simulations using the harmonic balance technique. In *45th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2007. AIAA Paper 2007-892.

[45] Thomas Gudeney, Adrien Gomar, Franois Gallard, Frdric Sicot, Guillaume Dufour, and Guillaume Puigt. Non-uniform time sampling for multiple-frequency harmonic balance computations. *Journal of Computational Physics*, 236:317–345, 2013.

[46] K.C. Hall. Computation of unsteady nonlinear flows in cascades using a harmonic balance technique. In *Kerrebrock symposium - a symposium in honor of Professor Jack L. Kerrebrocks 70th birthday*, Massachusetts Institute of Technology, Cambridge, MA, 1998.

[47] K.C. Hall and E.F. Crawley. Calculation of unsteady flows in turbomachinery using the linearized Euler equations. *AIAA Journal*, 27:777, 1989.

[48] K.C. Hall, K. Ekici, J.P. Thomas, and E.H. Dowell. Harmonic balance methods applied to computational fluid dynamics problems. *International Journal of Computational Fluid Dynamics*, (ahead-of-print):1–16, 2012.

[49] K.C. Hall, J.P. Thomas, and W.S. Clark. Computation of unsteady nonlinear flows in cascades using a harmonic balance technique. In *9th international symposium on unsteady aerodynamics, aeroacoustics and aeroelasticity of turbomachines*, Lyon, France, September 2000.

[50] K.C. Hall, J.P. Thomas, and W.S. Clark. Computation of unsteady nonlinear flows in cascades using a harmonic balance technique. *AIAA Journal*, 40(5):879–886, May 2002.

[51] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 71(2):231–303, 1987.

[52] R. Hartmann and P. Houston. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. *Journal of Computational Physics*, 183(2):508–532, 2002.

[53] L. He. Fourier methods for turbomachinery applications. *Progress in Aerospace Sciences*, 46(8):329–341, 2010.

[54] L. He, T. Chen, R.G. Wells, Y.S. Li, and W. Ning. Analysis of rotorrotor and statorstator interferences in multi-stage turbomachines. *Journal of Turbomachinery*, 124(4):564–571, 2002.

[55] L. He and J.D. Denton. Three-dimensional time-marching inviscid and viscous solutions for unsteady flows around vibrating blades. *Journal of Turbomachinery*, 116:469–476, 1994.

[56] L. He and W. Ning. Efficient approach for analysis of unsteady viscous flows in turbomachines. *AIAA Journal*, 36(11):2005–2012, 1998.

[57] T.J.R. Hughes. A simple scheme for developing upwind finite elements. *Internat. J. Numer. Methods Engrg.*, 12:1359–1365, 1978.

[58] T.J.R. Hughes and T.E. Tezduyar. Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *Comput. Methods Appl. Mech. Engrg.*, 45:217–284, 1984.

[59] A. Jameson, W. Schmidt, and E. Turkel. Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes.

In *AIAA 14th Fluid and Plasma Dynamics Conference*, Palo Alto, California, June 1981. AIAA Paper 81-1259.

[60] C. Johnson and J. Pitkaranta. An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. *Mathematics of Computation*, 46(173):1–26, 1986.

[61] W.P. Jones and B.E. Launder. The prediction of laminarization with a two-equation model of turbulence. *International Journal of Heat and Mass Transfer*, 15(2):301–314, 1972.

[62] R. Knapke, M. Galbraith, M. Turner, and P. Orkwis. An implicit harmonic balance method with a discontinuous Galerkin spatial scheme. In *51st AIAA Aerospace Sciences Meeting*, Grapevine, Texas, January 2013. AIAA Paper 2013-0513.

[63] M. Kumar and V.R. Murthy. Analysis of flow around multibladed rotor using CFD in the frequency domain. In *25th AIAA Applied Aerodynamics Conference*, Miami, FL, June 2007. AIAA Paper 2007-3806.

[64] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[65] Y. Maday, C. Mavriplis, and A.T. Patera. Nonconforming mortar element methods: application to spectral discretizations. In *Domain Decomposition Methods*, pages 392–418. SIAM, Philadelphia, PA, 1989.

[66] Y. Maday, N.C. Nguyen, A.T. Patera, and G.S.H. Pau. A general multipurpose interpolation procedure: the magic points. 8(1):383–404, January 2009.

[67] M. McMullen, J.J. Alonso, and A. Jameson. Acceleration of convergence to a periodic steady state in turbomachinery flows. In *Proceedings of the 39th AIAA Aerospace Sciences Meeting*. Citeseer, 2001. AIAA Paper 2001-0152.

[68] M. McMullen, A. Jameson, and J. Alonso. Application of a non–linear frequency domain solver to the Euler and Navier–Stokes equations. In *Proceedings of the 40th AIAA Aerospace Sciences Meeting*, 2002. AIAA Paper 2002-0120.

[69] D. Moro, N.C. Nguyen, and J. Peraire. Navier-Stokes solution using Hybridizable Discontinuous Galerkin methods. In *20th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, June 2011. AIAA Paper 2011-3407.

[70] S. Nadarajah, M. McMullen, and A. Jameson. Optimum shape design for unsteady flows using time accurate and non-linear frequency domain methods. In *33rd AIAA Fluid Dynamics Conference and Exhibit*, Orlando, Florida, June 2003.

[71] N.-C. Nguyen and J. Peraire. An adaptive shock-capturing HDG method for compressible flows. In *20th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, June 2011. AIAA Paper 2011-3060.

[72] N.-C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for linear convection–diffusion equations. *Journal of Computational Physics*, 228(9):3232–3254, 2009.

[73] N.-C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations. *J. Comp. Phys.*, 228:8841–8855, 2009.

[74] N.-C. Nguyen, J. Peraire, and B. Cockburn. A hybridizable discontinuous Galerkin method for stokes flow. *Computer Methods in Applied Mechanics and Engineering*, 199(9):582–597, 2010.

[75] N.-C. Nguyen, J. Peraire, and B. Cockburn. Hybridizable discontinuous Galerkin methods. *Spectral and High Order Methods for Partial Differential Equations*, pages 63–84, 2011.

[76] N.-C. Nguyen, J. Peraire, and B. Cockburn. Hybridizable discontinuous Galerkin methods for the time-harmonic Maxwell's equations. *Journal of Computational Physics*, 230(19):7151–7175, 2011.

[77] N.-C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for the incompressible navier–stokes equations. *Journal of Computational Physics*, 230(4):1147–1170, 2011.

[78] N.-C. Nguyen, X. Roca, D. Moro, and J. Peraire. A hybridized multiscale discontinuous Galerkin method for compressible flows. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2013. AIAA Paper 2013-0689.

[79] R.H. Ni and F. Sisto. Numerical computation of nonstationary aerodynamics of flat plate cascades in compressible flow. *Journal of Engineering for Power*, 98:165–170, 1976.

[80] A.T. Patera. A spectral element method for fluid dynamics: laminar flow in a channel expansion. *Journal of Computational Physics*, 54(3):468–488, 1984.

[81] J. Peraire, N.-C. Nguyen, and B. Cockburn. A hybridizable discontinuous Galerkin method for the compressible Euler and Navier-Stokes equations. In *48th AIAA Aerospace Sciences Meeting and Exhibit*, Orlando, FL, January 2010. AIAA Paper 2010-363.

[82] J. Peraire and P.-O. Persson. The compact discontinuous Galerkin (CDG) method for elliptic problems. *SIAM J. Sci. Comput.*, 30(4):1806–1824, 2008.

[83] P.-O. Persson, J. Bonet, and J. Peraire. Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains. *Computer Methods in Applied Mechanics and Engineering*, 198(17-20):1585 – 1595, 2009.

[84] M.M Rai. Unsteady three-dimensional Navier-Stokes simulations of turbine rotor-stator interaction. In *AIAA/SAE/ASME/ASEE 23rd Joint Propulsion Conference*, San Diego, California, June 1987. AIAA Paper 87-2058.

[85] W.H. Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. *Los Alamos Report LA-UR-73-479*, 1973.

[86] G.R. Richter. An optimal-order error estimate for the discontinuous Galerkin method. *Mathematics of Computation*, 50(181):75–88, 1988.

[87] X. Roca, N.C. Nguyen, and J. Peraire. Scalable parallelization of the hybridized discontinuous Galerkin method for compressible flow. In *21st AIAA Computational Fluid Dynamics Conference*, San Diego, CA, June 2013. AIAA Paper 2013-2939.

[88] A. Da Ronch, M. Ghoreyshi, K.J. Badcock, S. Gortz, M. Widhalm, R.P. Dwight, and M.S. Campobasso. Linear frequency domain and harmonic balance predictions of dynamic derivatives. In *28th AIAA Applied Aerodynamics Conference*, Chicago, Illinois, June 2010. AIAA Paper 2010-4699.

[89] Pierre Sagaut. *Large eddy simulation for incompressible flows.* Springer, 2002.

[90] F. Sicot, G. Dufour, and N. Gourdain. A time-domain harmonic balance method for rotor/stator interactions. *Journal of Turbomachinery*, 134(1), 2012.

[91] F. Sicot, G. Puigt, and M. Montagnac. Block-jacobi implicit algorithms for the time spectral method. *AIAA Journal*, 46(12), 2008.

[92] G.A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1–31, 1978.

[93] S.C. Soon, B. Cockburn, and H.K. Stolarski. A hybridizable discontinuous Galerkin method for linear elasticity. *International Journal for Numerical Methods in Engineering*, 80(8):1058–1092, 2009.

[94] P.R. Spalart and S.R. Allmaras. A one-equation turbulence model for aerodynamic flows. In *30th AIAA Aerospace Sciences Meeting and Exhibit*, January 1992. AIAA paper 92-0439.

[95] J.P. Thomas, C.H. Custer, E.H. Dowell, and K.C. Hall. Unsteady flow computation using a harmonic balance approach implemented about the OVERFLOW 2 flow solver. In *19th AIAA Computational Fluid Dynamics Conference*, San Antonio, Texas, June 2009. AIAA Paper 20094270.

[96] J.P. Thomas, E.H. Dowell, and K.C. Hall. Nonlinear inviscid aerodynamic effects on transonic divergence, flutter, and limit-cycle oscillations. *AIAA Journal*, 40(4):638–646, 2002.

[97] J.P. Thomas, E.H. Dowell, and K.C. Hall. Modeling limit cycle oscillation behavior of the f-16 fighter using a harmonic balance approach. In *45th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Palm Springs, California, April 2004. AIAA Paper 20041696.

[98] J.P. Thomas, E.H. Dowell, and K.C. Hall. Further investigation of modeling limit cycle oscillation behavior of the f-16 fighter using a harmonic balance approach. In *46th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Austin, Texas, April 2005. AIAA Paper 20051917.

[99] J.P. Thomas, K.C. Hall, and E.H. Dowell. A harmonic balance approach for modeling nonlinear aeroelastic behavior of wings in transonic viscous flow. In *44th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Norfolk, Virginia, April 2003. AIAA Paper 20031924.

[100] E. van der Weide, A. Gopinath, and A. Jameson. Turbomachinery applications with the time spectral method. In *35th AIAA Fluid Dynamics Conference and Exhibit*, Toronto, Ontario, June 2005. AIAA Paper 20054905.

[101] D.A. Venditti and D.L. Darmofal. Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow. *Journal of Computational Physics*, 164(1):204–227, 2000.

[102] D.A. Venditti and D.L. Darmofal. Grid adaptation for functional outputs: application to two-dimensional inviscid flows. *Journal of Computational Physics*, 176(1):40–69, 2002.

[103] J.M. Verdon, J.J. Adamczyk, and J.R. Caspar. Subsonic flow past an oscillating cascade with steady blade loading – basic formulation. In *Unsteady Aerodynamics, edited by R. B. Kinney, proceedings of a symposium held at the University of Arizona*, volume 2, pages 827–851, Tucson, Ariz., March 1975.

[104] J.M. Verdon and J.R. Caspar. Subsonic flow past an oscillating cascade with finite mean flow deflection. *AIAA Journal*, 18(5):540–548, May 1980.

[105] J.M. Verdon and J.R. Caspar. A linearized unsteady aerodynamic analysis for transonic cascades. *Journal of Fluid Mechanics*, 149:403–429, 1984.

[106] S. Vilmin, E. Lorrain, Ch. Hirsch, and M. Swoboda. Unsteady flow modeling across the rotor/stator interface using the nonlinear harmonic method. In *ASME Turbo Expo 2006: Power for Land, Sea and Air*, Barcelona, Spain, May 2006. Paper GT2006-90210.

[107] J.M. Weiss and K.C. Hall. Simulation of unsteady turbomachinery flows using an implicitly coupled nonlinear harmonic balance method. In *ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition (GT2011)*, Vancouver, British Columbia, Canada, June 2011. ASME Paper GT2011-46367.

[108] G.E. Welch, I.M. Milanovic, and K. Zaman. Application of harmonic balance technique to synthetic jets in cross-flow. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, January 2005. AIAA Paper 20051111.

[109] D.C. Wilcox. Reassessment of the scale-determining equation for advanced turbulence models. *AIAA Journal*, 26(11), 1988.

[110] M.A. Woodgate and K.J. Badcock. Implicit harmonic balance solver for transonic flow with forced motions. *AIAA Journal*, 47(4), 2009.

[111] M.A. Woodgate and G.N. Barakos. Implicit CFD methods for fast analysis of rotor flows. In *50th AIAA Aerospace Sciences Meeting*, Nashville, Tennessee, January 2012. AIAA Paper 2012-0421.

[112] Z. Yang, D. Mavriplis, and J. Sitaraman. Prediction of helicopter maneuver loads using BDF/time spectral method on unstructured meshes. In *49th AIAA Aerospace Sciences Meeting*, Orlando, Florida, January 2011. AIAA Paper 2011-1122.

[113] M. Yano. *An Optimization Framework for Adaptive Higher-Order Discretizations of Partial Differential Equations on Anisotropic Simplex Meshes.* PhD thesis, Massachusetts Institute of Technology, 2012.