Chapter 1

Efficient Relaxation Methods for High-Order Discretization of Steady Problems

Georg May
Graduate School AICES, RWTH Aachen University,
Schinkelstr 2., 52056 Aachen, Germany,
may@aices.rwth-aachen.de

Antony Jameson
Department of Aeronautics & Astronautics, Stanford University,
Durand Building, Stanford, CA 94305,
jameson@baboon.stanford.edu

We review the current status of solution methods for nonlinear systems arising from high-order discretization of steady compressible flow problems. In this context, many of the difficulties that one faces are similar to, but more pronounced than, those that have always been present in industrial-strength CFD computations. We highlight similarities and differences between the high-order paradigm and the mature solver technology of lower oder discretization methods, such as second order finite-volume schemes.

1. Introduction

Many have anticipated the arrival of high-order discretization as the CFD method of choice for compressible fluid flow. However, for industrial applications in external aerodynamics lower order methods, such as finite-volume schemes, are still far more popular. Numerical schemes of third or higher spatial order are often not efficient enough for high-throughput CFD computations to engineering levels of accuracy. Among the reasons for this is the fact that for established CFD methodologies tailor-suited convergence acceleration techniques have emerged over the past decades [Jameson (1983); Jameson and Yoon (1987); Pierce and Giles (1997); Jameson and Caughey (2001); Mavriplis (2002)]. High-order solvers thus compete with

very mature technology, and consequently novel discretization techniques have to be augmented by extremely efficient solution algorithms.

We present an overview of relaxation methods for steady compressible flow problems. This is to be understood in the sense that by virtue of spatial discretization the steady-state governing equations are converted to a nonlinear algebraic system of equations, which has to be solved. No time accuracy is required in this context, but time-accurate computations may also fall under this relaxation paradigm. For instance when implicit time discretization is employed, the solution of such an algebraic system of equations is required at each time instance.

One may argue that, in principle, the same relaxation methods and the same convergence theory may be applied to high-order discretization and low-order discretization. After all, a nonlinear algebraic system of equations is the result of spatial discretization in both cases. It is nevertheless true that the circumstances change when the order of accuracy is increased. As an example, consider two very popular paradigms in CFD computations for compressible flow, namely nonlinear multigrid methods with explicit multistage schemes, and implicit relaxation methods. Stability restrictions become a major concern for multigrid methods using explicit multistage relaxation, even on non-stretched meshes, as permissible CFL numbers of high-order methods typically behave as CFL $\propto m^{-2}$, where m is the polynomial degree of approximation [Hesthaven and Gottlieb (1999)]. Furthermore, the direct extension of multigrid methods to higher order schemes via the multi$-p$ approach is not entirely straight forward. On the other hand, implicit relaxation methods, such as Newton-Krylov methods, suffer from drawbacks as well, such as excessive storage requirements for high orders of approximation. We present an overview of viable relaxation methods with particular emphasis on constraints imposed by high-order spatial discretization, emphasizing such methods that are applicable to general unstructured grids.

2. Discretization Methods

The current state-of-the art in CFD focuses on solving the Euler or Navier-Stokes equations, the latter with suitable turbulence modeling. We write these equations generically as a system of conservation laws

$$\frac{\partial w}{\partial t} + \nabla \cdot f(w) = S(w) \ , \tag{1}$$

where w is the vector of conserved variables, f is the flux vector, including inviscid and viscous contributions from the governing equations, and possibly a turbulence model. The right-hand side may include a source term that often comes from a turbulence model.

For example, for the two-dimensional Euler equations for inviscid rotational fluid flow, w and f are written as

$$w = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ E \end{pmatrix} , \qquad f_j = \begin{pmatrix} \rho u_j \\ \rho u_j u_1 + p\delta_{j1} \\ \rho u_j u_2 + p\delta_{j2} \\ \rho u_j H \end{pmatrix} , \qquad j = 1, 2 \ . \tag{2}$$

Here ρ is the density, p is the pressure, E is the energy, and $H = (E+p)/\rho$ is the enthalpy. The fluid velocity vector is given by $u = (u_1, u_2)^T$. For a thermally and calorically perfect gas, one closes the equations by the equation of state

$$p = (\gamma - 1) \left(E - \frac{1}{2} \rho ||u||^2 \right) , \tag{3}$$

where γ is the ratio of specific heats.

There are a wide variety of high-order discretization methods for conservation laws, such as high-order finite-volume schemes [Barth (1993)], WENO schemes of finite difference or finite volume type [Shu (2003)], residual-distribution schemes [Abgrall and Roe (2003)], or *hp* finite-element methods [Karniadakis and Sherwin (2005)]. A very popular paradigm in high-order discretization is given by schemes based on piecewise polynomial representation, i.e. such schemes that, for a partition of the computational domain $\mathcal{T}_h = \{T\}$, approximate the solution of (1) as $w \approx w_h \in \mathcal{V}_h^p$, where $\mathcal{V}_h^p$ is the space of functions that are polynomials of degree p in each element, but are discontinuous across elements. Examples are the Discontinuous Galerkin (DG) method [Cockburn and Shu (2001)] or the Spectral Difference method [Liu *et al.* (2006); Wang *et al.* (2007)]. Attempts have been made to put some of these discretization approaches into a unified setting, such as Huynh's flux reconstruction approach [Huynh (2007)], the newly established Lifting Collocation Penalty method [Wang and Gao (2009)], or $P_n P_m$ schemes [Dumbser *et al.* (2008); Dumbser (2010)].

We shall not be overly concerned with discretization methods here, as the focus is very much on relaxation methods for steady problems, which is generally the task of solving the nonlinear algebraic set of equations resulting from the spatial discretization. However, we do emphasize such schemes that are based on local polynomial approximation on unstructured

grids. For example, omitting any limiting or shock capturing terms, a simple DG discretization for a steady hyperbolic conservation law of the type (1) without source term leads to the problem of finding $w_h \in \mathcal{V}_h^p$ such that

$$R(w_h; v_h) := \sum_{T \in \mathcal{T}_h} \left\{ - \int_T f(w_h) \nabla v_h \, dx + \int_{\partial T} g(w_h^+, w_h^-; n) v_h^+ \, ds \right\} = 0 \ , \tag{4}$$

for all $v_h \in \mathcal{V}_h^p$. The function g is a numerical flux, which defines the flux on element boundaries, where the solution is discontinuous, as a function of the solution u_h^+ in element T, and u_h^-, the solution in its neighbor. See e.g. [Roe (1981); Jameson (1995)] for the case of the Euler equations. While usually the DG discretization is formulated using the semilinear form (4), it is clear that once the basis and test functions are chosen, the residual is a function of the solution coefficients for w_h only, and we may suppress the test function v_h in the notation.

Another example is the Spectral Difference scheme for which one seeks $w_h \in \mathcal{V}_h^p$, using a nodal (Lagrange) basis, such that

$$R(w_h) := \nabla \cdot f_h(w_h) = 0 \ , \tag{5}$$

where f_h is a global interpolant of the nonlinear flux function f, which is continuous in normal direction across element interfaces by virtue of using numerical flux functions in the interpolation in a suitable manner, see [Kopriva and Kolias (1996); Liu *et al.* (2006); Wang *et al.* (2007)].

Since we only deal with the numerical solution w_h, we drop the subscript by default, and use it only when reference to a characteristic mesh length h is deemed necessary. Note that in Eq. (4) and Eq. (5) we use w_h to denote the assembled polynomial solution. Naturally, enforcing these equations means solving for discrete degrees of freedom, such as the modal coefficients or the collocation values, that together with corresponding basis functions define the numerical solution. In the following we shall associate w with the vector of discrete degrees of freedom. Likewise R corresponds to the pertaining residual evaluations. Thus we are left with a vector-valued nonlinear system of algebraic equations

$$R(w) = 0 \ , \tag{6}$$

where $R(w)$ is the residual vector.

The core of the present exposition is a pseudo time-dependent relaxation, marching the field equations to a steady state in a method of lines

approach. This means one considers the system of nonlinear ODE

$$\frac{dw}{d\tau} + R(w) = 0 \ . \tag{7}$$

Obviously no time accuracy is required if one wishes to iterate toward the steady state, allowing such convergence acceleration techniques as local time stepping and multigrid methods. The advantage of this approach is that a wide variety of methods may be formulated in this framework.

3. Explicit Multistage Methods

In the early development of DG methods, multistage time-stepping schemes have been very popular. Early publications introduced the Runge-Kutta Discontinuous Galerkin (RKDG) method [Cockburn and Shu (1988)], presenting spatial discretization and multistage time stepping as a combined scheme. While much of this classical work is devoted to presenting and analyzing the method for time-dependent problems, Runge-Kutta methods have since become popular for steady problems as well [Bassi and Rebay (1997); May *et al.* (2010)]. Runge-Kutta methods are easy to implement and parallelize, and have low memory requirements.

Consider the pseudo-time ODE Eq. (7). An M-stage multistage temporal discretization may be written

$$\begin{aligned} w^{(0)} &= w^n \ , \\ w^{(k)} &= \sum_{l=0}^{k-1} \left\{ \alpha_{kl} w^{(l)} - \Delta\tau \beta_{kl} R^{(l)} \right\} \ , \qquad k = 1, \ldots, M \ , \\ w^{n+1} &= w^{(M)} \ , \end{aligned} \tag{8}$$

where w^n is the n^{th} iterate of the solution, and $R^{(l)} := R(w^{(l)})$. Given a discretization that is TVD [Harten (1983)] with forward Euler time stepping, Shu proposed high order multistage schemes [Shu and Osher (1988)], which preserve the TVD property at high CFL numbers. These concepts have since been generalized under the paradigm of strong stability preserving (SSP) Runge-Kutta schemes [Gottlieb *et al.* (2001)]. TVD properties have been shown for Discontinuous Galerkin and Spectral Difference Schemes using standard limiting methods [Cockburn and Shu (1988); May (2006)]. The coefficients of the popular Shu RK3 scheme [Shu and Osher (1988)]

may be written, arranged in matrix form, as

$$\alpha = \begin{pmatrix} 1 & & \\ \frac{3}{4} & \frac{1}{4} & \\ \frac{1}{3} & 0 & \frac{2}{3} \end{pmatrix} , \qquad \beta = \begin{pmatrix} 1 & & \\ 0 & \frac{1}{4} & \\ 0 & 0 & \frac{2}{3} \end{pmatrix} . \tag{9}$$

It should be noted that this scheme allows preservation of TVD properties only at the same CFL number as a forward Euler time stepping scheme [Shu and Osher (1988)]. For time dependent problems this still may lead to superior efficiency due to high order accuracy in time. For steady problems however, temporal order of accuracy is immaterial, and the use of this scheme is merely justified by the fact that simpler schemes, such as forward Euler or the 2-stage TVD RK scheme [Shu and Osher (1988)] are not linearly stable with DG or Spectral Difference methods, which may lead to overactive limiters in the TVD discretization, and hence to compromised accuracy.

An alternative are low order but high-CFL number schemes, such as TVD / SSP schmes [Shu (1988); Gottlieb *et al.* (2001)] or Jameson's high-CFL number multistage schemes [Jameson (1983, 1993, 2004)] , which have been very popular in standard finite-volume CFD computations. These latter schemes have been designed using Fourier analysis for a linear model equation with the aim to maximize the stability region and at the same time provide good high-frequency error damping properties, which improves performance within multigrid algorithms.

The success of such multistage schemes for steady problems depends to a large extent on convergence acceleration techniques. Certainly the use of local time stepping methods is mandatory if no time accuracy is required. Time steps are adjusted so that they are always close to the local stability limit. If the mesh size increases, the time step, which is proportional to the local characteristic mesh length, will also increase, producing an effect comparable to that of an increasing wave speed. Furthermore, the combination of multistage schemes with multigrid, which we address in section 5.1, is one of the classic paradigms in compressible flow simulation. It should not be overlooked that the success of multistage methods in classical CFD methods have also relied on other convergence acceleration methods, such as implicit residual smoothing and related methods [Jameson (1988); Swanson *et al.* (2007)], which have not found a straight forward extension in the realm of higher order discretization methods.

While explicit relaxation methods are attractive due to ease of implementation and parallelization, stability restrictions are a concern. Often

spectra of the (linearized) discrete advection operators are investigated to infer stability properties [Karniadakis and Sherwin (2005)]. In the context of nodal DG schemes, or Spectral Difference schemes, such analysis has revealed that the spectral radius is proportional to m^2, where m is the polynomial order of approximation [Hesthaven and Gottlieb (1999); May (2006)], which suggests that stability for explicit methods necessitates $CFL \sim m^{-2}$. As an example, consider the one-dimensional linear advection equation and Discontinuous Galerkin or Spectral Difference Discretization, where permissible CFL numbers with respect to linear L_2 stability have been explicitly computed [May (2006)], see Table 1. The measure CFL · DOF used in Table 1, where DOF is the number of local degrees of freedom, is appropriate when making comparison with standard finite difference schemes using the same number of total degrees of freedom. The Spectral

Table 1. Linear stability limits for Spectral Difference and DG Schemes with the Shu RK3 scheme, and Jameson's four-stage scheme [Jameson (1983)].

	SD / Jameson RK4		SD / Shu RK3		DG / Shu RK3	
Pol. Order	CFL	CFL · DOF	CFL	CFL · DOF	CFL	CFL · DOF
0	0.696	1.392	0.595	1.190	0.409	0.818
1	0.363	1.089	0.322	0.966	0.209	0.627
2	0.226	0.904	0.201	0.804	0.130	0.520
3	0.156	0.780	0.139	0.695	0.089	0.445
4	0.115	0.690	0.103	0.618	0.066	0.396
5	0.089	0.623	0.079	0.559	0.051	0.357

Difference schemes in Table 1 use Gauss-Legendre quadrature points, augmented with cell interval end-points, which has recently been confirmed to be a stable choice by means of numerical eigenvalue analysis [Huynh (2007)] as well as rigorous proof [Jameson (2009)].

The rapid asymptotic decrease of permissible CFL numbers poses a severe challenge, certainly if the problem is exacerbated by numerical stiffness induced by stretched meshes. It remains to be seen if explicit relaxation methods will remain popular for practical high order viscous CFD computations. This depends to a large extent on whether convergence acceleration techniques such as multigrid methods can be incorporated succesfully.

The popular focus on nonlinear TVD stability theory has to some extent led to negligence of linear stability analysis for high-order schemes. It has to be stressed that many combinations of explicit time integration methods with higher order schemes, such as the Spectral Difference scheme, or stan-

dard RKDG schemes [Cockburn and Shu (2001)] are not unconditionally linearly stable [May (2006)]. While nonlinear stability results may still hold, flux limiters or artificial viscosity techniques are needed for stabilization. These may locally degrade the accuracy, even in smooth regions, if oscillations are generated by linear instability. For example, in the case of the 1D Spectral Difference Scheme, the popular Chebyshev-Lobatto nodes are not unconditionally stable [May (2006); Van den Abeele *et al.* (2008)] (and by extension tensor-products thereof). Linear instability has also been shown for the Spectral Difference scheme using different nodal sets for multivariate interpolation on triangular meshes [Van den Abeele *et al.* (2008)], although more recently a new formulation of the Spectral Difference scheme has been proposed, based on interpolation in Raviart-Thomas spaces, which seems to be linearly stable [May and Schöberl (2010)].

High-order DG or spectral methods for nonlinear equations can be stabilized quite effectively with filtering methods [Hesthaven and Warburton (2007)], meaning attenuation of higher order modes. While it has been demonstrated that even for fixed (intermediate) order schemes such an approach may be viable without significantly degrading the accuracy [Hesthaven and Warburton (2007)], this has not been explored too much for CFD applications.

Regardless of the stabilization method of choice, restrictive stability conditions of the type shown in Tab. 1 always apply for explicit relaxation methods, which makes it absolutely necessary to combine them with convergence acceleration techniques for steady problems.

4. Implicit Relaxation Methods

A linearized backward Euler temporal discretization of Eq. (7) may be written

$$(I + \Delta\tau A(w^n))\,\Delta w^n = -\Delta\tau R(w^n)\ , \tag{10}$$

where $\Delta w^n = w^{n+1} - w^n$ and A is the Jacobian matrix of the residual vector, i.e. the differentiation of the residual vector R with respect to the state vector w. For $\Delta\tau \to \infty$ one obtains a Newton iteration, while finite time steps may be interpreted as damped Newton iterations. For simplicity we shall often use the symbol $\mathbb{M}$ to denote the entire left-hand-side matrix in Eq. (10).

The hallmark of implicit methods is that a large sparse linear system, given by Eq. (10), has to be solved at each nonlinear iteration step n.

For most practical applications direct solution of the system is out of the question, so one has to resort to iterative methods. The key parameters in the implementation are

- Approximation and assembly of the Jacobian matrix A
- The iterative solution method for the linear system
- Preconditioning of the system

Finding the best overall approach is not trivial if the time to solution is to be minimized. A variety of different approaches have been proposed even for standard low-order CFD methods. Nevertheless, two approaches may be identified that are particularly popular: Newton-Krylov methods, corresponding to infinite-time steps in Eq. (10), and finite-time-step implicit methods using classical iterative solvers with convergence acceleration methods.

4.1. *Newton-Krylov Methods*

The Newton iteration potentially achieves quadratic convergence, provided the exact Jacobian matrix is available, and the linear systems arising at each iteration are solved to high precision. Newton's method is often combined with Krylov methods to solve the linearized equations at each iteration. Krylov methods (with a good preconditioner) are often advantageous if solution of the linear system to high precision is desired, and the system by itself is not necessarily well conditioned. This is usually the case if the time step in Eq. (10) is increased to infinity. Methods that rely on diagonal dominance, as many classical iteration methods do [Hackbusch (1994)], may not be a good choice for this case. Among the Krylov methods for non-symmetric systems that arise in CFD applications the GMRES method [Saad and Schultz (1986)] is quite popular. GMRES is very robust, in the sense that it cannot break down, unless the exact solution of the linear system is reached, and furthermore guarantees that the (linear) residual 2-norm is non-increasing. On the other hand, the method is quite expensive due to long recurrences of Krylov vectors, and usually requires good preconditioning to attain acceptable rates of convergence. We defer the issue of preconditioning to section 4.4.

In practice it is very difficult, if not impossible, to quantify *a priori* the region of attraction that must be reached, to attain convergence of the Newton iteration. Therefore some kind of globalization must be added to the method to allow convergence from an initial guess that may be far away

from the converged solution. For CFD applications a simple time-step control of the implicit temporal discretization, based on the size of the residual is usually fairly robust. As an example of a Newton-Krylov method applied to a compressible flow problem consider the test case depicted in Fig. 1. The convergence using a DG method with polynomials of degree $m = 2$

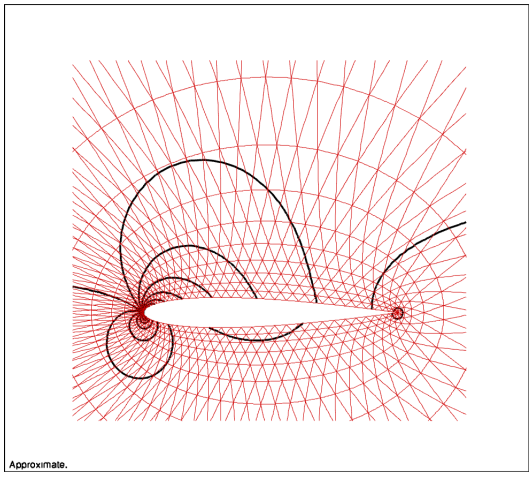

Fig. 1. Mach contours for inviscid flow around the NACA0012 profile. Free-stream Mach number $M_\infty = 0.4$, Angle of Attack $\alpha = 5^o$.

and $m = 4$ in terms of the norm $||R(w)||_\infty$ is shown in Fig. 2 and Fig. 3, respectively. Here and in the following NDOF is the number of degrees of freedom in the numerical approximation, i.e. $NDOF = N_e \times N_m \times N_w$, where N_e is the number of mesh elements, N_m is the number of local degrees of freedom, and N_w is the number of equations. It can be seen that very rapid convergence is attained, once the asymptotic region is reached.

To summarize, Newton-Krylov methods imply increased cost per iteration by requiring

(1) A good approximation of the Jacobian
(2) Solution of the linear system to relatively high precision (at least in the asymptotic region)
(3) A good preconditioner

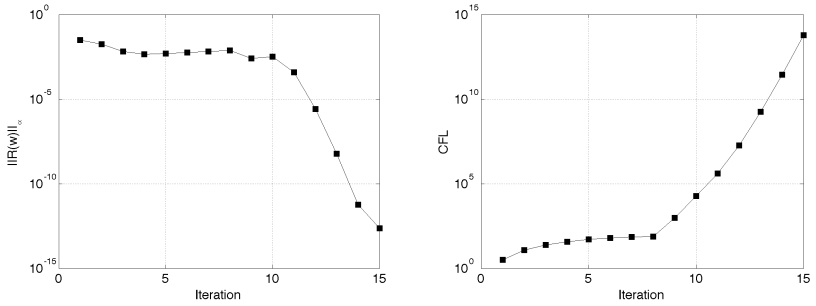

Fig. 2. Inviscid flow around the NACA0012 profile at $M_\infty = 0.4$, $\alpha = 5°$. Degrees of freedom: NDOF $= 61,440$. DG method with polynomial degree $m = 2$. Left: Convergence of the residual against nonlinear iterations. Right: Adaptive CFL number against nonlinear iterations.

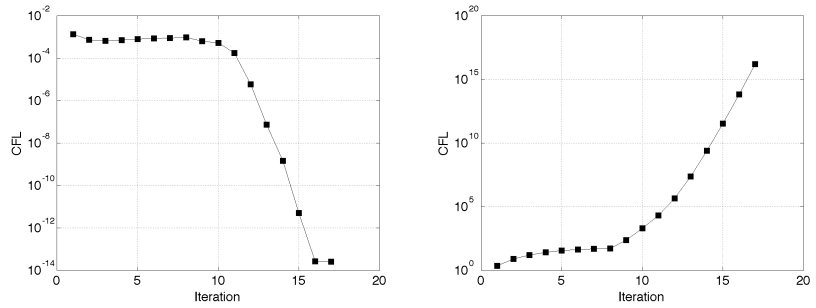

Fig. 3. Inviscid flow around the NACA0012 profile at $M_\infty = 0.4$, $\alpha = 5°$. Degrees of freedom: NDOF $= 61,440$. DG method with polynomial degree $m = 4$. Left: Convergence of the residual against nonlinear iterations. Right: Adaptive CFL number against nonlinear iterations.

4.2. *Implicit Schemes with Finite Time Steps*

Newton-Krylov methods imply a high cost per iteration, but at the same time a very low nonlinear iteration count. The opposite approach may also lead to success. One may use finite time steps in Eq. (10), resulting in a linear system with (relatively) high diagonal dominance, so that classical iteration techniques may be used. If a rather inexact solution of Eq. (10) is accepted, i.e. solving the system to relatively high residual levels and perhaps using a crude approximation of the Jacobian, the result is a significantly higher nonlinear iteration count, but also a dramatically reduced

cost per iteration. The viability of the concept depends on its successful combination with convergence acceleration techniques, such as multigrid, for which classical iterative linear solvers can be quite effective smoothers.

An example for this approach is given by relaxation methods of the Gauss-Seidel type. Consider a splitting $\mathbb{M} = D + L + U$, where D is the (block) diagonal, while L and U denote the strictly lower and upper triangular submatrices, respectively. A standard Gauss-Seidel method may be written upon setting $\Delta w^{n,0} = 0$,

$$(D^n + L^n)\Delta w^{n,k+1} = -R(w^n) - U^n \Delta w^{n,k} , \qquad k = 0, 1, 2, \dots . \quad (11)$$

Very often symmetric Gauss Seidel methods are used, that basically concatenate a forward and backward solve:

$$(D^n + L^n)\Delta w^{n,k+\frac{1}{2}} = -R(w^n) - U^n \Delta w^{n,k} , \quad (12)$$

$$(D^n + U^n)\Delta w^{n,k+1} = -R(w^n) - L^n \Delta w^{n,k+\frac{1}{2}} . \quad (13)$$

Particularly popular is the so-called LU-SGS method [Jameson and Yoon (1987); Yoon and Jameson (1988)], which is basically a one-step symmetric Gauss-Seidel method with zero initial guess. It is a matter of straightforward computation to show that this corresponds to a splitting

$$\Delta w^n = -(D^n + U^n)^{-1} D^n (D^n + L^n)^{-1} R(w^n) . \quad (14)$$

It is quite obvious that a small number of Gauss-Seidel sweeps does not solve the linear system to high precision. Nevertheless, such schemes have been applied to high-order discretizations by numerous reserachers in combination with multilevel convergence acceleration techniques [Luo *et al.* (2006); Nastase and Mavriplis (2006)], making this approach a good example for the trade-off considered above: The cost per iteration is extremely low, so that a higher nonlinear iteration count may be tolerated. Furthermore, the quality of the sweeps can be controlled by appropriately selecting the ordering of the state vector. Many examples exist in the literature where for both classical CFD computations and higher order methods such reorderings have been constructed to reflect lines of strong coupling of the equations [Mavriplis (1998); Fidkowski *et al.* (2005)], while in regions of generally weak coupling the relaxation may even be reduced to a Jacobi iteration. In order to reduce memory overhead, a nonlinear variant of the LU-SGS scheme, similar to that used by Jameson and Caughey in a finite-volume context [Jameson and Caughey (2001)], is sometimes applied to high-order discretization [Sun and Wang (2007); Premasuthan (2010); Parsani *et al.* (2010)].

Whether or not this approach is superior to a Newton iteration is highly problem dependent, and often also depends on the measure of convergence: Since the asymptotic quadratic convergence of Newton's method is very hard to beat, the solution of the nonlinear problem to machine accuracy is often most efficiently done with a good Newton solver. On the other hand, convergence of output functionals, such as lift or drag coefficients, is often very efficiently achieved to engineering levels of accuracy by multigrid-accelerated classical smoothing techniques.

4.3. *Matrix-free Methods*

For higher order methods based on local polynomial approximations, a major difficulty may be already encountered in the assembly of the Jacobian matrix. Let $\mathcal{P}^m$ be the space of polynomials of degree m. In 2D, $\dim(\mathcal{P}^m) \propto m^2$, while in 3D one has $\dim(\mathcal{P}^m) \propto m^3$. Since all local degrees of freedom are coupled in each cell, the overall storage requirements grow with the fourth power of the polyomial order m in 2D, and with the sixth power in 3D.

Storing the Jacobian matrix may not be feasible in some cases, forcing one either to resort to explicit relaxation methods, or to consider matrix-free formulations of implicit methods. For the latter, Krylov methods are particularly suited, since they require, neglecting preconditioning for the moment, the system matrix $\mathbb{M}$ only in the action on Krylov vectors, i.e. matrix-vector products of the form

$$z = \mathbb{M}v \ . \tag{15}$$

Note that the nontrivial part of this operation involves the Jacobian of the residual vector $R(w^n)$, namely in the computation of $A(w^n)v$, which is a projection of the derivative of the residual onto the Krylov vector v. One may generate a numerical approximation to first order accuracy in a small parameter ε by writing

$$A(w^n)v \approx \frac{R(w^n + \varepsilon v) - R(w^n)}{\varepsilon} \ . \tag{16}$$

Here the cost of applying the matrix-vector product is the same as one residual evaluation. There is some freedom in the choice of ε. Several methods have been proposed in the literature to estimate the step size [Knoll and Keyes (2004)]. A simple choice, supposing normalized Krylov vectors, is:

$$\varepsilon = \sqrt{1 + ||w||}\varepsilon_{rel} \ , \tag{17}$$

where the parameter ε_{rel} should roughly represent the square-root of machine accuracy.

4.4. *Preconditioning*

For Newton-Krylov methods it is usually the preconditioning that is most problem-dependent, the rest being a rather generic procedure. In case of the GMRES method, unfortunately the eigenvalue spectrum does not completely specify convergence properties, complicating the process of enabling fast convergence through good preconditioning. Pathological examples with extremely benign spectrum of the matrix, yet extremely poor GMRES convergence, may be constructed [Van der Vorst (2009)]. Preconditioning methods that reflect the physics and numerics of certain problems are often proposed, CFD applications being no exception [Persson and Peraire (2008)]. That being said, standard preconditioners based on incomplete LU factorizations (ILU) [Saad (2003)] are also often used with good results [May *et al.* (2010)].

If the preconditioner is explicitly assembled as a sparse matrix it is normally independent of the Krylov iteration index, i.e. does not change while the linear system is being solved. On the other hand, explicit storage of the preconditioner may not always be possible any more than storage of the matrix itself. For matrix-free preconditioners alternative formulations of the GMRES method, such as the flexible GMRES method [Soulaimani *et al.* (2002)] facilitate the implementation by allowing the preconditioner to depend on the linear iteration index. One generates a preconditioned Krylov vector by solving the linear system

$$P_j \widetilde{w}_j = z_j \ , \tag{18}$$

where $z_j = Av_j$. Since the preconditioning matrix is allowed to depend on the GMRES iteration index, one may use iterative solvers as preconditioners. Noting that the preconditioning matrix P should approximate A one may apply to equation Eq. (18) a few iteration steps with the same GMRES method that is used to solve Eq. (10), i.e. apply GMRES recursively. In particular, this may be done using the same matrix-free approximation. This method is denoted "squared preconditioning", due to the recursive application of the linear GMRES solver [May *et al.* (2010)]. In principle this algorithm could be recursively applied even further.

It should thus be pointed out that the method is completely matrix-free. This means that storage requirements grow linearly in the degrees of

freedom, as opposed to quadratically, which led to the extreme asymptotic storage requirements outlined in section 4.3. For the matrix-free variant memory requirements are now dominated by storage of the Krylov vectors, which in 3D is certainly still considerable, but manageable with modern computer architectures.

As an example of a computation using matrix-free implicit relaxation with squared preconditioning, consider the flow depicted in Fig. 4. Figure 5

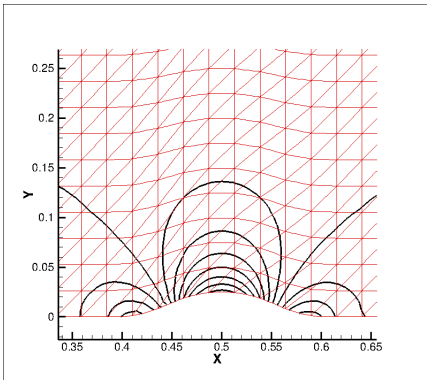

Fig. 4. Mach contours for inviscid flow around a smooth bump. Free-stream Mach number $M_\infty = 0.3$.

shows the convergence in terms of the density residual for different orders of approximation using a Spectral Difference method at constant CFL number (CFL=550). It may be seen that the convergence in terms of linear iterations, i.e. cumulative number of Krylov vectors (excluding preconditioning iterations), deteriorates with higher polynomial orders. This is due to the fact that the condition of the system matrix also deteriorates, and it is usually not advisable to increase the number of preconditioning iterations (inner iterations) too much as a countermeasure, since the matrix-free preconditioner is not as effective as matrix-based ones, and thus net runtime may increase despite fewer linear iterations.

5. Multilevel Methods

5.1. *Geometric Multigrid*

Multigrid is certainly one of the most popular paradigms within the applied CFD community. One may distinguish between linear multigrid methods,

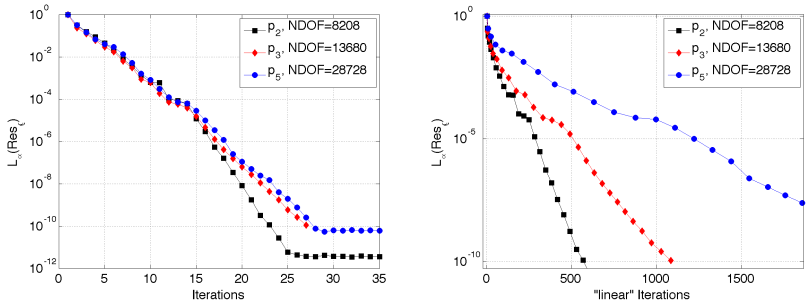

Fig. 5. Convergence of the matrix-free method for smooth inviscid flow around a bump at free-stream Mach number $M_\infty = 0.3$. Spectral Difference computation with polynomial degrees of 2,3, and 5. Left: Convergence against outer, nonlinear iterations. Right: Convergence against linear iterations, i.e. number of generated Krylov vectors.

that may be used as preconditioners in the context of implicit relaxation methods, and nonlinear methods, under the paradigm of the Full Approximation Scheme (FAS) [Brandt (1977)]. The latter is very popular in the combination with explicit multistage methods, following the (now classic) approach in [Jameson (1983)].

The FAS approach has traditionally been associated with geometric multigrid methods, which we consider first. Since it is standard practice to use only first order accurate solution methods on coarse-grid approximations, we first consider the special case $w_h|_T \in \mathcal{P}^0$ for all mesh elements T, and discuss the extension to higher order approximations afterwards.

Assume that the equations have been iterated n steps on a given mesh of characteristic length h, the "fine" mesh, by an explicit multistage scheme, as in section 3. This results in an approximation w_h^n, and residual $R_h^n = R_h(w_h^n)$. Using a suitable coarser mesh of characteristic length H, and defining appropriate restriction operators for the solution and residual, $I_h^H w_h^n$, and $\widetilde{I}_h^H R_h^n$, respectively, one may advance the solution on a coarse grid by the modified multistage scheme

$$w_H^{(k)} = \sum_{l=0}^{k-1} \left\{ \alpha_{kl} w_H^{(l)} + \Delta t \beta_{kl} \left(R_H^{(l)} + S_H \right) \right\} , \qquad k = 1, \ldots, M , \tag{19}$$

where the additional defect correction term

$$S_H = \widetilde{I}_h^H R_h - R_H^{(0)} \tag{20}$$

appears on the right-hand side [Jameson (1983); Mavriplis (2002)]. After

iterating on the coarse mesh for n_c iterations the corrected solution on the fine grid is computed as

$$w_h^+ = w_h^n + I_H^h(w_H^{n_c} - w_H^0) \ , \tag{21}$$

where I_H^h is an interpolation operator, and optionally additional smoothing on the fine mesh may be applied, before the updated solution is declared the new iterate at $n+1$.

One uses recursive application of this concept to extend the method to more than two meshes. Good results are usually obtained using W cycles, following standard nomenclature, see e.g. [Jameson (2004)]. These are defined by allowing transfer to the next higher level only if the solution has been advanced twice on the current mesh. Figure 6 shows a schematic de-

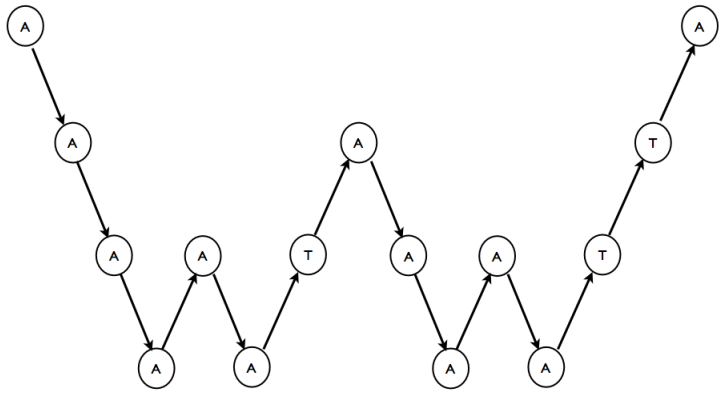

Fig. 6. W-Cycle for a 4-mesh sequence. The letter A stands for advancing the flow solution on a particular level, while the letter T stands for transfer of the solution to the next higher level.

piction of a 4-level W-cycle. This approach has proved particularly effective when combined with a nonlinear variant of the LU-SGS scheme [Jameson and Caughey (2001)].

The geometric approach introduces some overhead, since the availabil-

ity of mesh sequences is implied. Often those are generated by automatic coarsening procedures, such as agglomeration methods [Mavriplis and Venkatakrishan (1995)].

5.2. *Multi-p Methods*

In direct analogy to geometric multigrid we may define multi-p methods in broad terms as computing an approximation for the error of the current solution, $w_h|_T \in \mathcal{P}^m$, using a lower polynomial degree m_c*. However, just as for geometric multigrd methods the optimal mesh coarsening ratio is not always *a priori* clear, with multi$-p$ methods the same question applies to the lower polynomial degree one ought to use. While some Fourier analysis for linear model equations has been carried out to assess convergence factors for multi$-p$ methods [Atkins and Helenbrook (2005); Fidkowski *et al.* (2005)], the issue of how many polynomial levels one ought to include for high m is still open in general, and likely problem and discretization dependent. We shall simply assume that in a two-grid cycle we use polynomial levels m and m_c with $0 \leq m_c < m$.

One may use multi$-p$ in a similar fashion as described in the previous section for geometric multigrid methods. Assume that the equations have been iterated n steps using a discretization of local polynomial degree m, and mesh of characteristic length h, resulting in an approximation w_m^n, and residual $R(w_m^n)$. Note that here the mesh index has been suppressed, as it will not change in the multi$-p$ iteration, and instead the subscript m has been added. Defining appropriate transfer operators for the solution and residual, $I_m^{m_c} w_m^n$, and $\widetilde{I}_m^{m_c} R(w_m^n)$, respectively, one may solve the equation

$$R(w_{m_c}) + S_{m_c} = 0 \tag{22}$$

where the additional defect correction term

$$S_{m_c} = \widetilde{I}_m^{m_c} R(w_m^n) - R(w_{m_c}^0) \tag{23}$$

appears on the right-hand side. After relaxing on the polynomial level m_c for n_c iterations the corrected solution on the level m may be computed, using the prolongation operator $I_{m_c}^m$, as

$$w_m^+ = w_m^n + I_{m_c}^m (w_{m_c}^{n_c} - w_{m_c}^0) \ , \tag{24}$$

which may be declared the next solution iterate, upon optionally applying some further smoothing, as in the geometric multigrid case. Such an ap-

*In our nomenclature we prefer to use multi-p in place of the somewhat more popular, but misleading, term p-multigrid.

proach has been used, for example, in [Premasuthan (2010)] with Spectral Difference discretization and a Runge-Kutta smoother, and in [Fidkowski *et al.* (2005)] with DG discretization and implicit smoothers of the block Jacobi and line relaxation type.

For multi$-p$ methods it has been, however, at least as popular to use a linear multilevel paradigm with implicit relaxation schemes, instead of the FAS approach. Applying multilevel techniques when solving Eq. (10) allows a straightforward interpretation as a preconditioner for linear systems. Using a suitable smoother for Eq. (10), one transfers the residual

$$r_m = -R(w_m^n) - \mathbb{M}_m \Delta w_m^n \tag{25}$$

to a lower order approximation, i.e. $r_{m_c} = \mathcal{I}_m^{m_c} r_m$. Subsequently one solves the error equations directly, i.e.

$$\mathbb{M}_{m_c} e_{m_c} = r_{m_c} \tag{26}$$

which may be done recursively using yet more levels. The corrected solution is then obtained as $w_m^+ = w_m^n + \mathcal{I}_{m_c}^m e_{m_c}$. In [Nastase and Mavriplis (2006)] the linear approach applied to a DG discretization was found superior in terms of runtime. A major advantage is certainly the reduced number of nonlinear residual evaluations, which are particularly costly in a higher order context. (Keep in mind that during a multi$-p$ iteration the mesh is fixed, so that the cost of evaluating the residual does not decrease as dramatically as with geometric multgigrid.)

For best results one ought to combine multi$-p$ and geometric multgrid. Recall that for nonlinear convection-dominated problems geometric multigrid aids through two mechanisms: firstly, the elimination of high-frequency error modes on successively coarser meshes; secondly, the propagation of error modes, and expulsion from the computational domain [Pierce and Giles (1997)]. While asymptotic convergence rates are dominated by high-frequency smoothing, early convergence is dominated by convection. Often one observes effectively converged output functionals, such as lift and drag coefficients at relatively high residual levels, before asymptotic convergence rates are reached. In this phase, geometric multigrid may be viewed primarily as an increase of the effective wave speed propagating error modes, which is, however, dependent on global coarsening. Since multi$-p$ methods do not provide such global coarsening, it is likely that best-practice multilevel solvers will still have to include geometric multigrid.

5.3. *Hybrid Multilevel Schemes*

It is certainly possible to use multigrid with different relaxation schemes on different mesh levels or levels of polynomial approximation. This leads to hybrid multilevel schemes. Depending on the constraints deemed important one may find very different "optimal" combinations. For example, in [Luo *et al.* (2006)] a multi$-p$ DG scheme is proposed that combines Shu's three-stage Runge-Kutta method, cf. Eq. (9), for polynomial levels of approximation $m > 0$, with implicit LU-SGS solves, cf. Eq. (14), for $m = 0$, with the primary concern being storage requirements.

A different method was proposed in [May *et al.* (2010)], where a damped Newton/GMRES implicit method is used for the highest level of polynomial approximation $m > 0$. Storage concerns are addressed with an optional matrix-free formulation. A geometric multigrid method with explicit multistage smoothing is used for the volume averages (i.e. for $m = 0$) between Newton iterations to accelerate the convection of the volume-averaged large-scale error modes. The smoothed volume averages replace the volume averages of the high-order relaxation. The rationale behind this is that experience indicates that error convection and expulsion is the primary mode of convergence, when considering integrated quantities, such as force coefficients. When using geometric multigrid methods, which accelerate the effective wave speed for error convection and expulsion, force coefficients are often essentially converged at rather high residual levels, when high-frequency errors still persist.

The method in [May *et al.* (2010)] is completed by a full multigrid (FMG) finite-volume start-up procedure. Algorithm 5.1 gives an example of a practical implementation of the overall approach. Let $RKMG(w; l, n)$ denote the application of n iterations of Runge-Kutta smoothing with $l-$level geometric multigrid. The mesh levels are identified by an indexed characteristic length h_l, where the coarsest mesh is indexed with $l = 1$, while the finest available mesh is indexed $l = L$. For easy reference we have denoted volume averages by an overbar, i.e. $\overline{w}_h$, while w_h indicates the solution at the current high-order polynomial level m. The implicit solves are denoted by $NK(w; m, n)$, where again, n is the number of iterations, and m is the level of polynomial approximation.

The first loop over the meshes defines the startup procedure. The computation starts on the coarsest grid using a finite-volume method with explicit multistage relaxation, and then proceeds up to the next finer grid when a sufficiently good approximation to the solution has been achieved.

Algorithm 5.1. Hybrid Multilevel with Full Multigrid

(1) Initialize $\overline{w}^0_{h_1}$ with free stream conditions
(2) For $l = 1, \ldots, L$, Do
(3) $\quad \overline{w}^n_{h_l} = \mathrm{RKMG}(\overline{w}^0_{h_l}; l, n_l)$
(4) $\quad$ if (l = L) exit
(5) $\quad \overline{w}^0_{h_{l+1}} = I^{h_{l+1}}_{h_l} \overline{w}^n_{h_l}$
(6) EndDo
(7) $w^0_{h_L} = Inject(\overline{w}^n_{h_L}; m)$
(8) For $n = 0, \ldots, N_{cyc}$, Do
(9) $\quad w^+_{h_L} = \mathrm{NK}(w^n_{h_L}; m, 1)$
(10) $\quad$ if (converged) exit
(11) $\quad \overline{w}^0_{h_L} = \mathcal{V}(w^+_{h_L})$
(12) $\quad \overline{w}^+_{h_L} = \mathrm{RKMG}(\overline{w}^0_{h_L}; L, n_{RK})$
(13) $\quad w^{n+1}_{h_L} = w^+_{h_L} - Inject(\overline{w}^0_{h_L} - \overline{w}^+_{h_L}; m)$
(14) EndDo

This is applied recursively, re-using all available coarser meshes with FAS multigrid, until the finest mesh is reached. The number of multigrid cycles n_l should be enough to attain reasonable convergence of integrated quantities, such as lift and drag, on each mesh level.

The result of the finite-volume relaxation procedure is used as initial guess for the damped Newton iteration acting on the high-order discretization in Algorithm 5.1, step 7. We define the injection operator $Inject(\overline{w}; m)$, which injects the volume average for the current approximation order m. Obviously the definition of this operator depends on the chosen basis and degrees of freedom. This operator is also used in the subsequent multilevel relaxation procedure. Because of the good start value provided by the initial full multigrid relaxation, only very mild damping for a few iterations has to be used to avoid start-up problems in the Newton-Krylov method on the highest level.

The main loop is over the combined Newton / GMRES and explicit smoothing operators. First the implicit iterator $NK(w; m, n)$ is applied. Note that usually $n = 1$, as shown in Algorithm 5.1, step 9. Subsequently the volume averages are extracted in step 11, where the operator is denoted $\mathcal{V}$. This is particularly easy for hierarchical bases that are often used with DG methods, e.g. [Dubiner (1991)]. It is easily accomplished also for the non-hierarchical Spectral Difference basis by (exact) numerical quadrature based on the solution nodes. Finally the explicit multigrid iterations are

performed for the volume averages, which produces updated values that replace the previous ones. Typically around $n_{RK} = 20$ iterations are used for the additional $RKMG$ smoothing between Newton iterations. Intermediate polynomial levels $0 < m_c < m$ are not used in the nonlinear multigrid cycles, but may be used within this framework under the linear multigrid paradigm, i.e. as a preconditioner for the linear systems, although incomplete LU factorizations also work effectively.

As a computational example, consider the inviscid flow test case summarized in Fig. 7 using the Spectral Difference Scheme with $m = 2$. Figure 8

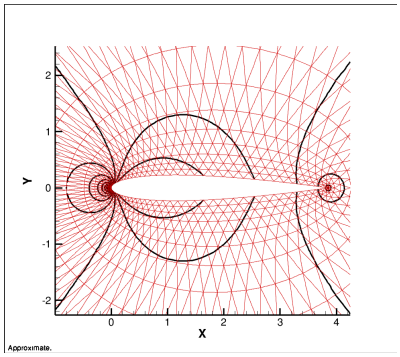

Fig. 7. Mach contours for inviscid flow around the NACA0012 profile. Free-stream Mach number $M_\infty = 0.3$, Angle of Attack $\alpha = 0^o$.

shows the convergence of the hybrid method, Algorithm 5.1, under mesh refinement in terms of the drag coefficient. Here the CFL number has been kept constant at CFL=550 to highlight the mesh independent convergence. Both nonlinear iterations, and linear iterations are shown, where the latter refers to the cumulative number of generated Krylov vectors. It can be seen that convergence of both degrades very severely for the single-grid method, while the convergence is nearly mesh independent for the hybrid mulitlevel method. The mesh sequence used for these computations is shown in Table 2 and Table 3 . More precisely, in Table 2 the three meshes used in the refinement study are summarized, while in Table 3 the multilevel data for the coarsest of these meshes is shown. For the finer meshes in Table 2 it should be understood that all previously defined coarser meshes are used recursively (with finite volume approximation). Thus the medium mesh uses a 4-level strategy, while the finest mesh uses 5 levels.

A similar test case has been computed for the flow conditions $M_\infty = 0.4$

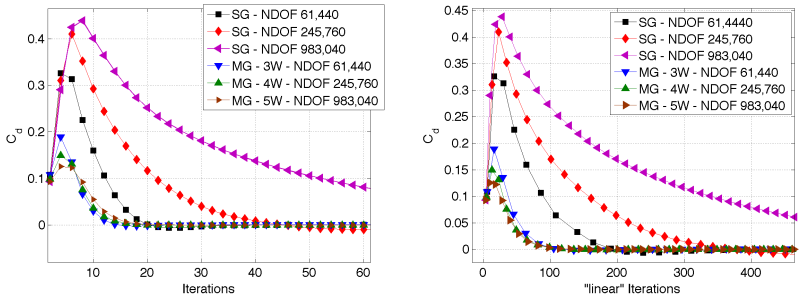

Fig. 8. Inviscid flow around the NACA0012 profile at free-stream Mach number $M_\infty = 0.3$, angle-of-attack $\alpha = 0^o$. Convergence of the single-grid method (SG), and the hybrid multilevel method, Algorithm 5.1, using 20 finite-volume multigrid cycles between Newton iterations. Krylov solver: GMRES(30) with ILU(2) preconditioning. Left: Convergence against outer, nonlinear iterations. Right: Convergence against linear iterations, i.e. number of generated Krylov vectors.

Table 2. Meshes used in h-refinement study.

Level	$NDOF$	Elements
fine	983,040	40,960
medium	245,760	10,240
coarse	61,440	2,560

and $\alpha = 5^o$, i.e. the case depicted in Fig. 1. The convergence in terms of the lift coefficient, plottet against CPU time, is shown in Fig. 9 for the three different meshes in Table 2.

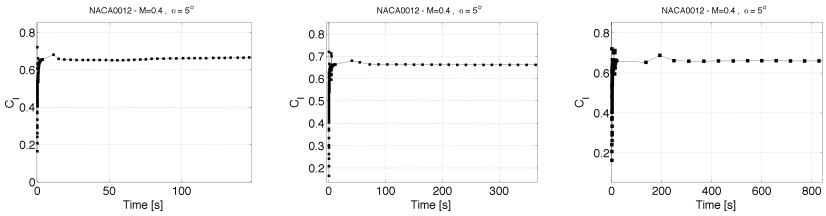

Fig. 9. Mesh refinement study. Inviscid flow around the NACA0012 profile at $M_\infty = 0.4$, $\alpha = 5°$. Convergence of the lift coefficient. Degrees of freedom: NDOF $= 61,440$ (left) , NDOF $= 245,760$ (middle) , NDOF $= 983,040$ (right).

Table 3. Meshes and Degrees of freedom used with the hybrid multilevel method on the coarsest mesh of Table 2.

Hybrid Multilevel					
Level	$NDOF$	m	Cells	CFL	Smoothing
4	61,440	2	2,560	550	Implicit
3	10,240	0	2,560	6	Explicit
2	2,560	0	640	6	Explicit
1	640	0	160	6	Explicit

6. Conclusion

We reviewed approaches to the solution of nonlinear systems arising from high-order spatial discretization in a CFD context. It is not advisable to end such a review with a clear recommendation on what method ought to be generally preferred, as such a choice is always problem-dependent. Readers familiar with best-practice low-order CFD methods will recognize the same trade-offs that have always existed: Restrictive stability restrictions with explicit methods that require fine-tuned convergence acceleration techniques, high memory requirement with implicit relaxation methods, and the problem of adequate preconditioning.

In the context of higher order discretization methods, however, these trade-offs are often more pronounced. Efficient solution needs to be defined in terms of available resources and objective of the calculation, which may inform the decision as to what relaxation scheme should be used. For example, solving the nonlinear equations to machine accuracy is a very different task compared with high-throughput computations that focus on convergence of output quantities to engineering levels of accuracy. Available resources, such as computer memory, may just dictate the choice of relaxation method. For example, the enormous storage requirements of Newton-Krylov methods for 3D computations may sometimes be prohibitive.

It must be said, that efficient solution of steady compressible flow problems to relatively modest levels of accuracy is still a domain dominated by the mature technology of standard best-practice lower order methods. However, solution methods that are well adapted to the unique environment of high-order discretization are an area of very active research, and the transition from model problems to more realistic applications is well underway. It is entirely possible that well designed hp-adaptive solvers will

be able to challenge the current status-quo in the near future.

References

Abgrall, R. and Roe, P. L. (2003). High order fluctuation schemes on triangular meshes, *J. Sci. Comp.* **19**, 1-3, pp. 3–36.

Atkins, H. L. and Helenbrook, B. T. (2005). Numerical evaluation of p-multigrid for the solution of discontinuous galerkin discretizations of diffusive equations, AIAA Paper 05-5110, American Institute of Aeronautics and Astronautics.

Barth, T. J. (1993). Recent developments in high order k-exact reconstruction on unstructured meshes, AIAA paper 93-0668, American Institute of Aeronautics and Astronautics.

Bassi, F. and Rebay, S. (1997). A high-order accurate discontinuous finite-element method for the numerical solution of the compressible Navier-Stokes equations, *J. Comp. Phys.* **131**, pp. 267–279.

Brandt, A. (1977). Multi-level adaptive solutions to boundary-value problems, *Math. Comp.* **31**, 138, pp. 333–390.

Cockburn, B. and Shu, C. W. (1988). TVB Runge-Kutta local projection Discontinuous Galerkin finite element method for conservation laws II: General framework, *Math. Comp.* **52**, 186, pp. 411–435.

Cockburn, B. and Shu, C. W. (2001). Runge-Kutta Discontinuous Galerkin methods for convection-dominated problems, *J. Sci. Comp.* **16**, 3, pp. 173–261.

Dubiner, M. (1991). Spectral methods on triangles and other domains, *J. Sci. Comput.* **6**, 4, pp. 345–390.

Dumbser, M. (2010). Arbitrary high order pnpm schemes on unstructured meshes for the compressible Navier-Stokes equations, *Computers and Fluids* **39**, 1, pp. 60–76.

Dumbser, M., Balsara, D. S., Toro, E. F. and Munz, C.-D. (2008). A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes, *J. Comp. Phys.* **227**, 18, pp. 8209–8253.

Fidkowski, K. J., Oliver, T. A., Lu, J. and Darmofal, D. L. (2005). p–Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations, *J. Comp. Phys.* **207**, pp. 92–113.

Gottlieb, S., Shu, C.-W. and Tadmor, E. (2001). Strong stability-preserving high-order time discretization methods, *SIAM Review* **43**, 1, pp. 89–112.

Hackbusch, W. (1994). *Iterative Solution of Large Sparse Systems of Equations*, *Applied Mathematical Sciences*, Vol. 95 (Springer-Verlag).

Harten, A. (1983). High-resolution schemes for hyperbolic conservation laws, *J. Comp. Phys.* **49**, 3, pp. 357–393.

Hesthaven, J. S. and Gottlieb, D. (1999). Stable spectral methods for conservation laws on triangles with unstructured grids, *Comput. Meth. Appl. Mech. Engrg.* **175**, pp. 361–381.

Hesthaven, J. S. and Warburton, T. (2007). *Nodal Discontinuous Galerkin Meth-*

ods: Algorithms, Analysis, and Applications, no. 54 in Texts in Applied Mathematics (Springer Verlag).

Huynh, H. T. (2007). A flux reconstruction approach to high-order schemes including discontinuous galerkin methods, AIAA Paper 07-4079, American Institute of Aeronautics and Astronautics.

Jameson, A. (1983). Solution of the Euler equations for two dimensional transonic flow by a multigrid method, *Appl. Math. Comp.* **13**, pp. 327–356.

Jameson, A. (1988). Computational transonics, *Comm. Pure Appl. Math.* **41**, 5, pp. 507–549.

Jameson, A. (1993). Computational algorithms for aerodynamic analysis and design, *Appl. Numer. Math.* **13**, 5, pp. 383–422.

Jameson, A. (1995). Analysis and design of numerical schemes for gas dynamcics 2: Artificial diffusion and discrete shock structure, *Int. J. Comp. Fluid. Dyn.* **5**, pp. 1–38.

Jameson, A. (2004). Aerodynamics, in E. Stein, R. De Borst and T. J. R. Hughes (eds.), *Encyclopedia of Computational Mechanics*, Vol. 3, chap. 11 (Wiley).

Jameson, A. (2009). A proof of the stability of the spectral difference method for all orders of accuracy, Report ACL 2009-1, Aerospace Computing Laboratory, Stanford University.

Jameson, A. and Caughey, D. A. (2001). How many steps are required to solve the Euler equations of steady compressible flow: In search of a fast solution algorithm, AIAA Paper 01-2673, American Institute of Aeronautics and Astronautics.

Jameson, A. and Yoon, S. (1987). Lower-upper implicit schemes with multiple grids for the Euler equations, *AIAA Journal* **25**, 7, pp. 929–935.

Karniadakis, G. E. and Sherwin, S. (2005). *Spectral/hp Element Methods for Computational Fluid Dynamics*, 2nd edn. (Oxford University Press).

Knoll, D. A. and Keyes, D. E. (2004). Jacobian-free newton–krylov methods: a survey of approaches and applications, *J. Comp. Phys.* **193**, pp. 357–397.

Kopriva, D. A. and Kolias, J. H. (1996). A conservative staggered-grid Chebyshev multidomain method for compressible flows, *J. Comp. Phys.* **125**, pp. 244–261.

Liu, Y., Vinokur, M. and Wang, Z. J. (2006). Spectral Difference method for unstructured grids I: Basic formulation, *J. Comp. Phys.* **216**, 2, pp. 780–801.

Luo, H., Baum, J. D. and Löhner, R. (2006). A p-multigrid Discontinuous Galerkin method for the Euler equations on unstructured grids, *J. Comp. Phys.* **211**, pp. 767–783.

Mavriplis, D. J. (1998). Multigrid strategies for viscous flow solvers on anisotropic unstructured meshes, *J. Comp. Phys.* **145**, 1, pp. 141–165.

Mavriplis, D. J. (2002). An assessment of linear versus nonlinear multigrid methods for unstructured mesh solvers, *J. Comp. Phys.* **175**, 1, pp. 302 – 325.

Mavriplis, D. J. and Venkatakrishan, V. (1995). Agglomeration multigrid for two-dimensional viscous flows, *Computers and Fluids* **24**, 5, pp. 553–570.

May, G. (2006). *A Kinetic Scheme for the Navier-Stokes Equations and High-Order Methods for Hyperbolic Conservation Laws*, Ph.D. thesis, Stanford

University, Stanford, CA 94305.

May, G., Iacono, F. and Jameson, A. (2010). A hybrid multilevel method for high-order discretization of the Euler equations on unstructured meshes, *J. Comp. Phys.* **229**, 10, pp. 3938–3956.

May, G. and Schöberl, J. (2010). Analysis of a spectral difference scheme with flux interpolation on raviart-thomas elements, AICES Technical Report 2010-04/8, Aachen Institute for Advanced Study in Computational Engineering Science.

Nastase, C. R. and Mavriplis, D. J. (2006). High-order Discontinuous Galerkin methods using an *hp*-multigrid approach, *J. Comp. Phys.* **213**, pp. 330–357.

Parsani, M., Van den Abeele, K., Lacor, C. and Turkel, E. (2010). Implicit LU-SGS algorithm for high-order methods on unstructured grid with p-multigrid strategy for solving the steady Navier-Stokes equations, *J. Comp. Phys.* **229**, 3, pp. 828–850.

Persson, P.-O. and Peraire, J. (2008). Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations, *SIAM J. Sci. Comput.* **30**, 6, pp. 2709–2733.

Pierce, N. A. and Giles, M. B. (1997). Preconditioned multigrid methods for compressible flow calculations on stretched meshes, *J. Comp. Phys.* **136**, pp. 425–445.

Premasuthan, S. (2010). *Towards an efficient and Robust High Order Accurate Flow Solver for Viscous Compressible Flows*, Ph.D. thesis, Stanford University, Stanford, CA 94305.

Roe, P. L. (1981). Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comp. Phys.* **43**, pp. 357–372.

Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*, 2nd edn. (Society for Industrial and Applied Mathematics).

Saad, Y. and Schultz, M. H. (1986). GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems, *SIAM J. Sci. Stat. Comp.*. **7**, pp. 856–869.

Shu, C.-W. (1988). Total-variation-diminishing time discretizations, *SIAM J. Sci. Stat. Comput.* **9**, 6, pp. 1073–1084.

Shu, C.-W. (2003). High-order finite difference and finite volume WENO schemes and discontinuous Galerkin methods for CFD, *Int. J. Comput. Fluid Dyn.* **17**, 2, pp. 107–118.

Shu, C.-W. and Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock capturing schemes, *J. Comp. Phys.* **77**, pp. 439–471.

Soulaimani, A., Salah, N. B. and Saad, Y. (2002). Enhanced GMRES acceleration techniques for some CFD problems, *Int. J. Comp. Fluid. Dyn.* **16**, 1, pp. 1–20.

Sun, Y. and Wang, Z. J. (2007). Efficient implicit non-linear LU-SGS approach for viscous flow computation using high-order Spectral Difference method, AIAA Paper 07-4322, American Institute of Aeronautics and Astronautics.

Swanson, R. C., Turkel, E. and Rossow, C. C. (2007). Convergence acceleration of Runge-Kutta schemes for solving the Navier-Stokes equations, *J. Comp. Phys.* **224**, 1, pp. 365–388.

Van den Abeele, K., Lacor, C. and Wang, Z. J. (2008). On the stability and accuracy of the spectral difference method, *J. Sci. Comput.* **37**, 2, pp. 162–188.

Van der Vorst, H. A. (2009). *Itertive Krylov Methods for Large Linear Systems* (Cambridge University Press).

Wang, Z. J. and Gao, H. (2009). A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids, *J. Comp. Phys.* **228**, 21, pp. 8161–8186.

Wang, Z. J., Liu, Y., May, G. and Jameson, A. (2007). Spectral Difference method for unstructured grids II: Extension to the Euler equations, *J. Sci. Comput.* **32**, 1, pp. 54–71.

Yoon, S. and Jameson, A. (1988). Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations, *AIAA Journal* **26**, 9, pp. 1025–1026.