Designing optimal smoothers for multigrid methods for unsteady flow problems

Philipp Birken*

University of Kassel, Kassel, 34132, Germany

We consider the solution of unsteady flow problems using multigrid methods. There, the typical solver comes from a dual time stepping approach, where an existing method designed for a steady problem is used for unsteady flows. This design mismatch leads to a significant decrease in convergence speed. Here, we will use the linear advection equation as a model problem to find optimal Runge-Kutta smoothers specifically for the unsteady case, based on a Fourier analysis. This leads to a significantly improved multigrid method.

I. Introduction

During the last decade, numerical methods for unsteady flows have garnered increasing attention. In a way, this can be attributed to a certain maturity reached by methods for steady flows. As was shown by Caughey and Jameson,[1] the solution of steady Euler flows is possible in three to five multigrid steps. Thus, steady two dimensional flows around airfoils can be solved on a PC in a matter of seconds. The solution of the steady RANS equations is more difficult and takes between 50 and 200 steps with a fast solver, which means that adequate methods for steady flows exist. Regarding unsteady flow phenomema, for a lot of applications, the interesting features are not on the scale of the fast acoustic eigenvalues, but on the scale of the convective eigenvalues. This makes implicit schemes for time integration much more interesting than explicit schemes, which are then severely restrained by the CFL condition. Usually, A-stable methods are employed. The applicability of these schemes is determined by the availability of fast solvers for the arising large nonlinear equation systems.

If we consider as target application three dimensional unsteady compressible viscous flows, it becomes apparent that a fast solver must have strong parallel scalability and that memory requirements must be low. The above mentioned multigrid method scales reasonably well and has low storage requirements. Using dual time stepping, it can be used for unsteady flows. However, it turns out that the convergence rate deteriorates significantly, because the multigrid method was finely tuned for the steady Euler equations Nevertheless, for unsteady Euler flows, we still obtain a reasonably fast method. When we go further away from steady Euler flows, namely to the unsteady Navier-Stokes equations, the dual time stepping multigrid method was observed to be very slow, in particular for turbulent flows on high aspect ratio grids. Note that this has been demonstrated in the context of discontinuous Galerkin methods as well.[2]

The alternative to multigrid is to use Newton's method, which requires the solution of large sparse linear equation systems, usually by preconditioned Krylov subspace methods like GMRES or BiCGSTAB. In particular Jacobian-free methods that circumvent computation and storage of the Jacobian are an attractive option, see the overview paper by Knoll and Keyes.[3] However, the preconditioner should be chosen appropriately Jacobian-free as well. Now, a lot of methods that work well on sequential machines like ILU or SGS do not scale well in parallel, resulting in multigrid appearing as an attractive preconditioner.

To improve the multigrid method for steady Euler for different equations or discretizations, a few approaches have been tried. Jameson and Hsu suggest to use one step of the ADI method, followed by few multigrid steps for the dual time problem,[4] which is similar to using one Newton step, followed by dual time stepping. Bijl and Carpenter on the other hand use k_1 dual time stepping up front, followed by k_2 steps of Newton's methods.[5] Both report an improvement in comparison to the base pure dual time stepping

*Scientific employee, Inst. of Mathematics, University of Kassel, Heinrich-Plett-Str. 40, 34132 Kassel, email: birken@mathematik.uni-kassel.de, AIAA Member.

scheme. Recently, Birken and Jameson proved that using multigrid as a nonlinear preconditioner is not a good choice.[6] Here, based on the observation that the deterioration of the multigrid convergence rate for unsteady flows is connected to the design of the method for a different problem, we will try to taylor the smoother in the multigrid method to unsteady flow problems.

Again the question arises which smoother to chose. Considering the demand for low storage and parallel scalability, explicit Runge-Kutta smoothers appear as an attractive option. Here, several parameters can be chosen to obtain a good smoother. For the steady Euler equations, this was done by Jameson[7] using a mix of analysis and intuition and for steady linear advection equation by van Leer et. al.,[8] who provided optimal coefficients for that equation, which also turned out to work well for the steady Euler equations. In the case of a discontinuous Galerkin method, Bassi et. al.[9] used the same methodology to come up with optimal coefficients for that discretization, again for the steady linear advection equation and demonstrated that these work reasonably well for the steady Euler equations.

Here, we will extend this methodology to unsteady problems, where we will use the linear advection equation as a model equation and discretize this with a finite volume scheme. To obtain optimal coefficients, the eigenvalues and eigenvectors of the discrete and continuous system are considered, giving rise to the notion of smooth and nonsmooth error components in this context. Then, an optimization problem is solved to obtain the coefficients of the smoother, in particular the coefficients of the RK method and its CFL number. The resulting schemes are then compared to the method of Van Leer for the steady Euler equations.

II. Governing equations and discretization

We consider the linear advection equation with $a > 0$ on the interval $x \in [0, 2]$ with periodic boundary conditions:

$$u_t + a u_x = 0. \tag{1}$$

An equidistant FV discretization for (1) with mesh width Δx leads to an evolution equation for the mean value in one cell i, located in the midpoint between two cell boundaries:

$$u_{i_t} + \frac{a}{\Delta x}(u_i - u_{i-1}) = 0.$$

Using implicit Euler with time step size Δt, the resulting linear system for the vector $\mathbf{u} = (u_1, ..., u_m)^T$ is

$$\begin{aligned} &\mathbf{u}^{n+1} - \mathbf{u}^n + \frac{a\Delta t}{\Delta x}\mathbf{B}\mathbf{u}^{n+1} = 0 \\ &\Leftrightarrow \mathbf{u}^n - \mathbf{A}\mathbf{u}^{n+1} = 0 \end{aligned} \tag{2}$$

where

$$\mathbf{A} = \left(I + \frac{\nu}{\Delta x}\mathbf{B}\right) \tag{3}$$

with $\nu = a\Delta t$ and

$$\mathbf{B} = \begin{pmatrix} 1 & & & & -1 \\ -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix}.$$

If we consider nonperiodic boundary conditions, the entry in the upper right corner of $\mathbf{B}$ becomes zero. Otherwise, nothing is changed.

III. Basic multigrid method

The multigrid method we use here is based on agglomeration, which corresponds best to finite volume discretizations. Thus, the restriction and prolongation are given by

$$\mathbf{R} = 0.5 \begin{pmatrix} 1 & 1 & & & & \\ & & 1 & 1 & & \\ & & & \ddots & \ddots & \\ & & & & 1 & 1 \end{pmatrix} \text{ and } \mathbf{P} = \begin{pmatrix} 1 & & & & \\ 1 & & & & \\ & 1 & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & & 1 \\ & & & & 1 \end{pmatrix}.$$

The coarse grid matrix is obtained by discretizing the problem on that grid. We use a V-cycle and presmoothing only. On the coarsest level, the smoother is applied instead of the usual direct solve, since this better corresponds to the Full Approximation scheme used for the nonlinear equations. Thus we obtain the scheme:

Function MG($\mathbf{x}_l, \mathbf{b}_l, l$)

- $\mathbf{x}_l = \mathbf{S}_l^{\nu_1}(\mathbf{x}_l, \mathbf{b}_l)$ (Presmoothing)
- if ($l > 0$)
  - $\mathbf{r}_{l-1} = \mathbf{R}_{l-1,l}(\mathbf{b}_l - \mathbf{A}_l\mathbf{x}_l)$ (Restriction)
  - $\mathbf{v}_{l-1} = 0$
  - Call $MG(\mathbf{v}_{l-1}, \mathbf{r}_{l-1}, l-1)$ (Computation of the coarse grid correction)
  - $\mathbf{x}_l = \mathbf{x}_l + \mathbf{P}_{l,l-1}\mathbf{v}_{l-1}$ (Correction via Prolongation)
- end if

As smoothers, we consider s-stage low-storage explicit Runge-Kutta schemes of the form

$$\begin{aligned} u_0 &= u_n \\ u_j &= u_n + \alpha_j \Delta t^* f(u_{j-1}), \quad j = 1, ..., s-1 \\ u_{n+1} &= u_n + \Delta t^* f(u_{s-1}), \end{aligned}$$

where the α_j and Δt^* are free parameters. We make the common consistency requirement that $\alpha_j \in [0, 1]$. The differential equation resulting from a dual time stepping approach to (2) is a hyperbolic equation with source terms in pseudo time t^*:

$$\mathbf{u}_{t^*} = \mathbf{u}^n - \mathbf{u}(t^*) - \frac{\nu}{\Delta x}\mathbf{B}\mathbf{u}(t^*), \quad \mathbf{u}(t_0^*) = \mathbf{u}^n. \tag{4}$$

One step of the RK smoother thus consists of performing one step of the RK scheme for the solution of the above equation (4).

IV. Optimizing the smoother

The eigenvectors of the matrix A from (3) are discrete forms of the functions $e^{ix\Theta}$ for various Θ and the eigenvalues are given by

$$\lambda(\Theta) = -1 - \frac{\nu}{\Delta x}(1 - e^{-i\Theta}). \tag{5}$$

If nonperiodic boundary conditions are used, the matrix becomes lower triangular and all eigenvalues are equal to $-1 - \frac{\nu}{\Delta x}$. Now, on the coarse grid, we can represent functions with $\Theta \in [-\pi/2, \pi/2]$. Thus, the smoother has to take care of error components with $|\Theta| \in [\pi/2, \pi]$.

For a linear problem, an explicit s-stage RK smoother can be described by its stability polynomial P_s of degree s as

$$u_{n+1} = P_s(\Delta t^* A)u_n$$

with Δt^* the time step in pseudotime. For example, an explicit 2-stage RK scheme is given by

$$P_2(z) = 1 + z + \alpha_1 z^2 = (z - z_1)(z - z_2), \tag{6}$$

$$z_{1/2} = -\frac{1}{2\alpha_1} \pm \frac{i}{2\alpha_1}\sqrt{4\alpha_1 - 1}.$$

For a 3-stage scheme we have two free parameters α_1 and α_2 and the polynomial

$$P_3(z) = 1 + z + \alpha_2 z^2 + \alpha_1\alpha_2 z^3 = 1 + z + \alpha_2 z^2 + bz^3 \tag{7}$$

with $b = \alpha_1\alpha_2$.

Due to the linearity, it is sufficient to look at $P_s(\Delta t^*\lambda(\Theta))$ with (see (5))

$$\Delta t^*\lambda(\Theta) = -\Delta t^* - \frac{\nu\Delta t^*}{\Delta x}(1 - e^{-i\Theta}).$$

Possible parameters of the smoother are the pseudo time step size Δt^* and the coefficients of the RK method. Now, $\nu = a\Delta t$ is fixed during the multigrid iteration, but Δx not. Furthermore, the pseudo time step is restricted by a CFL condition based on ν. Thus, instead of optimizing for Δt^*, we define the pseudo time step on each grid level as

$$\Delta t^* = c\Delta x$$

and optimize for $c := \Delta t^*/\Delta x$. Now we have

$$z(\Theta, c; \nu, \Delta x) := \Delta t^*\lambda(\Theta) = -c\Delta x - \nu c + \nu c e^{-i\Theta}. \tag{8}$$

With

$$e^{-i\Theta} = \cos(-\Theta) + i\sin(-\Theta) = \cos(\Theta) - i\sin(\Theta)$$

we obtain

$$z(\Theta, c; \nu, \Delta x) = -c\Delta x - \nu c + \nu c\cos(\Theta) - i\nu c\sin(\Theta).$$

In the end, given ν and Δx, we have to solve an optimization problem where we look at the modulus of the maximal value of the smoother for $|\Theta| \in [\pi/2, \pi]$ and then try to minimize that over the parameters α_j and c. Using symmetry of P_s and equivalenty looking at the square of the modulus, we obtain

$$\min_{c, P_s} \max_{|\Theta| \in [\pi/2, \pi]} |P_s(z(\Theta, c; \nu, \Delta x))|^2. \tag{9}$$

Note that due to the dependence of the optimal coefficients on ν and Δx, there is no unique optimal smoother for all problems and the optimal smoother might be different for different time steps.

For the 2-stage scheme (6), we have due to $\mathrm{Re}\, z_1 = -\frac{1}{2\alpha}$ and $|z_1| = \frac{1}{\sqrt{\alpha}}$:

$$\begin{aligned} |P_2(z)|^2 = |(z - z_1)(z - \bar{z_1})|^2 = |z^2 - 2z\mathrm{Re}z_1 + |z_1|^2|^2 = |z^2 + z/\alpha + 1/\alpha|^2 \\ = |(\mathrm{Re}z)^2 - (\mathrm{Im}z)^2 + \mathrm{Re}z/\alpha + 1/\alpha + i2\mathrm{Re}z\mathrm{Im}z + i\mathrm{Im}z/\alpha|^2 \\ = \left((\mathrm{Re}z)^2 - (\mathrm{Im}z)^2 + \mathrm{Re}z/\alpha + 1/\alpha\right)^2 + (2\mathrm{Re}z\mathrm{Im}z + \mathrm{Im}z/\alpha)^2 \end{aligned}$$

Similar computations for the 3-stage scheme (7) lead to

$$\begin{aligned} |P_3(z)|^2 \quad = \quad & (1 + \mathrm{Re}z + \alpha_2\mathrm{Re}z^2 - \alpha_2\mathrm{Im}z^2 + b\mathrm{Re}z^3 - 3b\mathrm{Re}z\mathrm{Im}z^2)^2 + \\ & (\mathrm{Im}z + 2\alpha_2\mathrm{Re}z\mathrm{Im}z - b\mathrm{Im}z^3 + 3b\mathrm{Re}z^2\mathrm{Im}z)^2. \end{aligned}$$

It turns out that for these functions, the final form of (9) is too difficult to solve exactly. Therefore, we discretize the parameter space and compute an approximate solution. This requires a bounded region, which is already the case for Θ and the α_j, which are between 0 and 1. As for c, we know that the explicit RK scheme has a bounded stability region, therefore we just chose an upper bound for c and are satisfied, if the optimal value for c is not on the boundary. As an example, $\max_{|\Theta| \in [\pi/2, \pi]} |P_2(z(\Theta, c; 25/120, 1/24))|$ is shown as a function of c and α_1 in figure 1. Note that the optimal c is on the boundary, meaning that the choice $c \in [0, 2]$ here is reasonable.

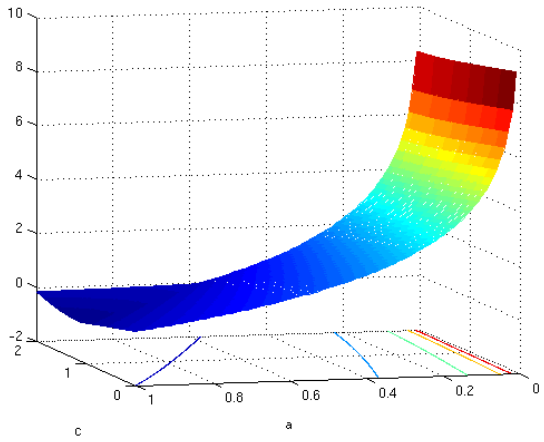

Figure 1. Function $\max_{|\Theta|\in[\pi/2,\pi]} |P_2(z(\Theta, c; 25/120, 1/24))|$.

For the 2-stage scheme, we chose a grid of $100 \times 200 \times 100$ for the parameter space $\alpha_1 \times c \times t$ and we make the restriction $c \in [0, 2]$. The optimization gives results presented in table 1 (left). As can be seen, the parameter α does not depend on Δx, whereas there is a weakly linear dependence of c on Δx. We choose $\alpha = 1$ and $c = 1.13$ for the new RK-2-smoother.

For the 3-stage scheme, we have one more parameter, which increases the dimension of the computational grid, which is why we chose $50 \times 50 \times 250 \times 100$ for the space $\alpha_1 \times \alpha_2 \times c \times t$. As a restriction for c, we put $c \in [0, 7]$. The results can be see in table 1 (right). This time, there is a weak dependence of all the parameters on Δx. We chose $\alpha_1 = 0.15$, $\alpha_2 = 0.4$ and $c = 6.18$ for the RK-3 smoother.

Table 1. Results of optimization for 2-stage scheme (left) and 3-stage scheme (right), $\nu = 25/120$

	α	c	Opt-value	α_1	α_2	c	Opt-value
Van Leer et. al.	1/3	0.48					
$\Delta x = 1/24$	1.	1.13	0.5630	0.15	0.4	6.18	0.014894
$\Delta x = 1/12$	1.	1.03	0.5628	0.15	0.4	5.56	0.013523
$\Delta x = 1/6$	1.	0.87	0.5626	0.11765	0.34	6.52	0.0075749

V. Numerical results

We test this on two problems with $\Delta x = 1/24$ on the finest level and $a = 25/12$. As initial conditions, we use a step function with values 5 and 1, as well as the function $\sin(\pi x)$. We then perform one time step with $\Delta t = 0.1$, meaning that $\nu = 25/120$. As a reference, the method of Van Leer et. al.[8] is used, which has $\alpha = 1/3$ and a CFL number of 1. This CFL number is based on a, thus giving $\Delta t^* = \Delta x/a$ or otherwise put $c^{ref} = 1/a = 0.48$. All computations are performed using MATLAB.

The results for the step function are shown in figure 2, whereas the results for the sine data can be seen in figure 3. On the left, the initial data is always shown in blue, and in red the exact solution, as well as the numerical one. Since the space discretization is of first order and the time integration method is implicit Euler, the results are very diffusive. However, the multigrid method converges linearly to the exact solution of the discrete equations. The convergence curves are always shown on the right for the method of Van Leer et al and the new optimized RK-2 and RK-3 method. As can be seen, the optimized methods are significantly faster than the reference method. In particular, the reference method reduces the error by about 1.15 per iteration, whereas the optimized RK-2 smoother by 1.37 and the RK-3 smoother by 5.6. As for the computational effort, the reference method and the RK-2 smoother have exactly the same computational cost, whereas the RK-3 smoother mainly needs one additional matrix vector multiplication. Since these form the main work of the method, the RK-3 smoother is about 50% more costly than the RK-2 smoother.

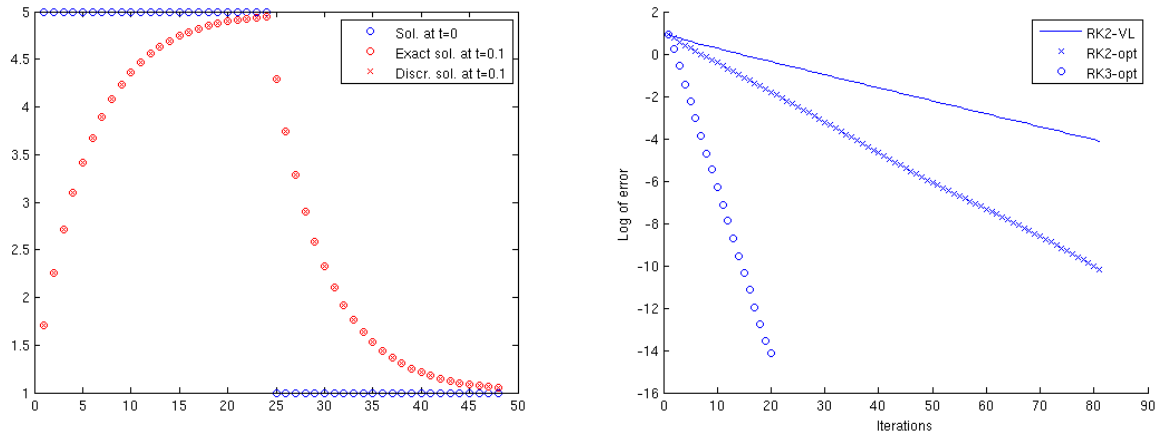

Figure 2. Initial solution and discrete and numerical solution after one time step (left) and convergence plots for different methods (right) for step function initial data.

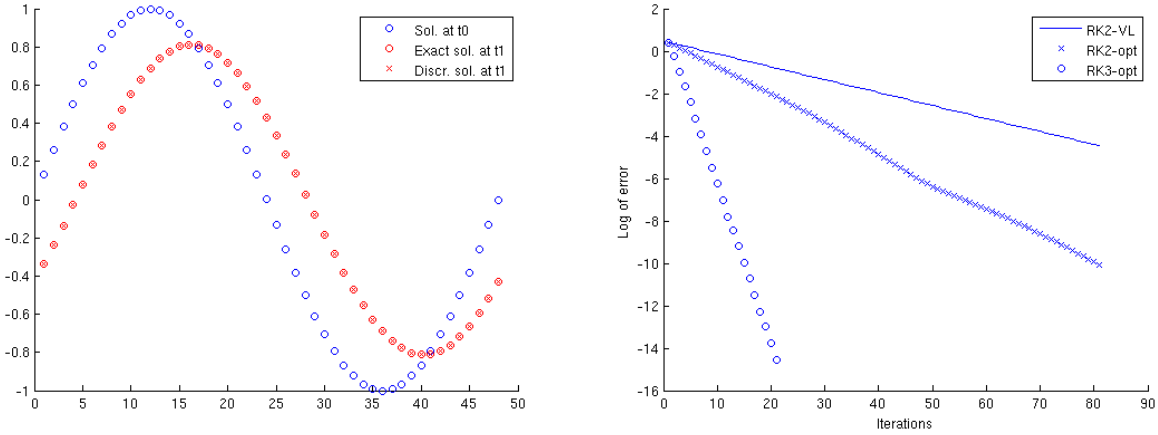

Figure 3. Initial solution and discrete and numerical solution after one time step (left) and convergence plots for different methods (right) for sine initial data.

Now, we are going to apply the method to the step function case with nonperiodic boundary, to see if the different eigenvalues lead to problems. As can be see in figure 4, this is not the case and the convergence rate for all methods is almost unchanged.

VI. Conclusions

We developed optimal explicit 2- and 3-stage Runge-Kutta smoothers for the unsteady linear advection equation. These were then demonstrated to significantly improve convergence speed, compared to using a method designed for steady state. This shows that it does pay to optimize for unsteady flows instead of reusing the method for steady flows. The optimal smoother does depend on the problem parameters, but only weakly on the mesh width, whereas the dependence on ν has to be tested in future work. Furthermore, to find methods that perform well for unsteady Euler or even Navier-Stokes equations, we will persue the use of more complex model equations for future research.

Acknowledgements

This work was supported by the german research foundation (DFG) as part of the collaborative research area SFB/TR TRR30, project C2.

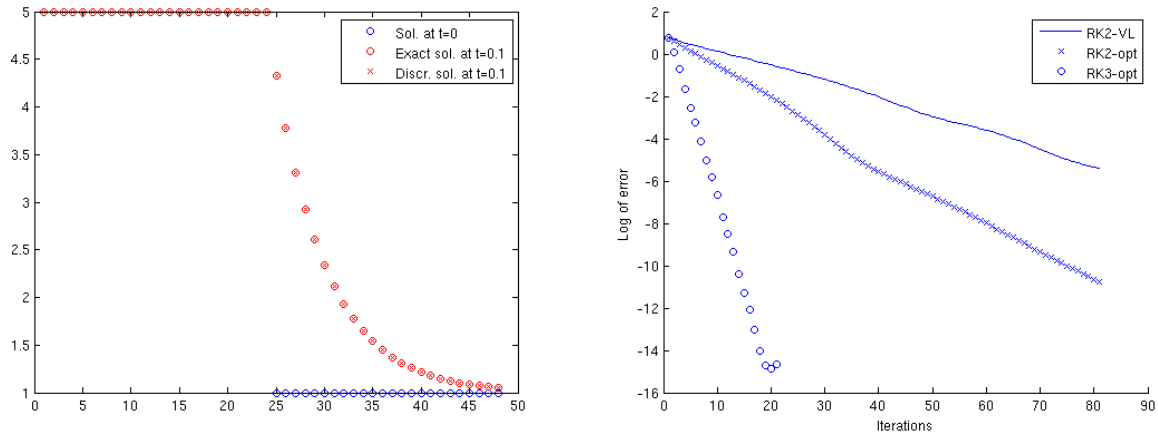

Figure 4. Initial solution and discrete and numerical solution after one time step (left) and convergence plots for different methods (right) for step function initial data with nonperiodic boundary conditions.

References

[1]Caughey, D. A. and Jameson, A., "How Many Steps are Required to Solve the Euler Equations of Steady Compressible Flow: In Search of a Fast Solution Algorithm," *AIAA-Paper*, Vol. 2001-2673, 2001.

[2]Luo, H., Baum, J. D., and Löhner, R., "A p-multigrid discontinuous Galerkin method for the Euler equations on unstructured grids," *J. Comp. Phys.*, Vol. 211, 2006, pp. 767–783.

[3]Knoll, D. and Keyes, D., "Jacobian-free Newton-Krylov methods: a survey of approaches and applications," *J. Comp. Phys.*, Vol. 193, 2004, pp. 357–397.

[4]Hsu, J. and Jameson, A., "An Implicit-Explicit Hybrid Scheme for Calculating Complex Unsteady Flows," *AIAA Paper 2002-0714*, 2002.

[5]Bijl, H. and Carpenter, M. H., "Iterative Solution Techniques for Unsteady Flow Computation using Higher Order Time Integration Techniques," *Int. J. Numer. Meth. Fluids*, Vol. 47, 2005, pp. 857–862.

[6]Birken, P. and Jameson, A., "On Nonlinear Preconditioners in Newton-Krylov-Methods for Unsteady Flows," *Int. J. Num. Meth. Fluids*, Vol. 62, 2010, pp. 565–573.

[7]Jameson, A., "Aerodynamics," *Encyclopedia of Computational Mechanics, Volume 3: Fluids*, edited by E. Stein, R. de Borst, and T. J. Hughes, John Wiley and Sons, 2004, pp. 325–406.

[8]van Leer, B., Tai, C.-H., and Powell, K. G., "Design of Optimally Smoothing Multi-Stage Schemes for the Euler Equations," *AIAA Paper*, Vol. 89-1933-CP, 1989, pp. 40–59.

[9]Bassi, F., Ghidoni, A., and Rebay, S., "Optimal Runge-Kutta smoothers for the *p*-multigrid discontinuous Galerkin solution of the 1D Euler equations," *J. Comp. Phys.*, 2010.