Proceedings of ALGORITMY 2009
pp. 91–100

# A MULTILEVEL DISCONTINUOUS GALERKIN METHOD FOR THE COMPRESSIBLE NAVIER-STOKES EQUATIONS

F. PRILL , M. LUKÁČOVÁ-MEDVIĎOVÁ , AND R. HARTMANN

**Key words.** Discontinuous Galerkin Method, Multigrid Method, Navier-Stokes Equations.

**AMS subject classifications.** 35Q30, 65F10, 65M22, 65M55, 65N30, 65N35.

**1. Introduction.** For about a decade the discontinuous Galerkin method (dGFEM) has become popular for the discretization of the nonlinear conservation laws of fluid dynamics. To the desirable properties of this high-order finite element method belong the flexibility in handling unstructured triangulations of complex geometries, mesh adaptation and freedom in the choice of the polynomial basis. In addition, the dGFEM can be viewed as a generalization of the finite volume method. For details we refer to the recent studies [2, 3, 11] and the references cited therein.

As a major drawback, the discrete problems that arise from the dGFEM discretization consist of a relatively large number of unknowns. Therefore iterative methods that require the assembly of the stiffness matrix remain limited to low-order discretizations. Further difficulties occur in the context of computational fluid dynamics, such as anisotropies induced by the operator as well as by the computational meshes. Thus there is the need for a scalable, robust and memory efficient iterative solver for the dGFEM discretization.

The purpose of this paper is to present a solution algorithm for nonlinear problems arising for the dGFEM discretization that takes these difficulties into account. We propose a multiplicative coupling of two different methods. The first part is constituted by a nonlinear multilevel iteration that exploits the nested hierarchy of spaces of varying polynomial degree $p$. The second component is a Newton-multigrid method with the smoothed aggregation technique for the solution of the nonlinear low-order problems. The combination of both parts yields a robust multilevel solver that does not require the explicit introduction of a sequence of coarse meshes. We demonstrate the feasibility of the approach for laminar two-dimensional compressible flow.

Nonlinear multigrid approaches for the dGFEM have been previously described and demonstrated, see for example [8]. We know only one previous work on a smoothed aggregation coarse space for a domain decomposition preconditioner for the dGFEM [12]. The application to an algebraic type multigrid method and the combination with a multi-$p$ method for compressible flow are the focus of the current work of the authors [13].

**2. Continuous and discrete problems.** We consider two different model problems. Firstly, we consider as usual a scalar linear model problem. By this we hope to get some insight for the solution of the nonlinear system of the compressible Navier-Stokes equations. We explicitly state the dGFEM discretization for the scalar advection-diffusion equation in the following, whereas for the Navier-Stokes equations, we refer to [3, 11].

**2.1. Scalar model problem.** The model problem under consideration is the linear advection-diffusion equation on a polyhedral domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$:

$$-\varepsilon \Delta u + \beta \cdot \nabla u = f \quad \text{in } \Omega. \tag{2.1}$$

We assume $f \in L^2(\Omega)$, $\varepsilon \geq 0$ and incompressible, Lipschitz-continuous wind $\beta = \{\beta_i\}_{i=1}^d : \mathbb{R}^d \to \mathbb{R}^d$. The boundary $\Gamma = \partial\Omega$ is partitioned into sets $\Gamma_-$, $\Gamma_+$, $\Gamma_\mathrm{D}$, $\Gamma_\mathrm{N}$ denoting the inflow and outflow part of the boundary, respectively, and the Dirichlet and Neumann part of the remaining edges. Boundary conditions are imposed as

$$u = g_\mathrm{D} \text{ on } \Gamma_\mathrm{D} \cup \Gamma_-, \quad \mathbf{n} \cdot (\nabla u) = g_\mathrm{N} \text{ on } \Gamma_\mathrm{N}.$$

For the well-posedness of this problem, see for example [14].

**2.2. Compressible Navier-Stokes equations.** For laminar compressible flow the conservative state is given by $\mathbf{u} = [\rho, \rho\mathbf{v}, \rho E]^T$ with $\rho$ denoting the density, $\mathbf{v}$ the velocity vector and $E$ the specific total energy. The Navier-Stokes equations are formulated in the Cartesian coordinate system as

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathcal{F}^c(\mathbf{u}) - \mathcal{F}^v(\mathbf{u}, \nabla\mathbf{u})) = \mathbf{0} \quad \text{in } \Omega \subset \mathbb{R}^d \tag{2.2}$$

with convective and viscous fluxes $\mathcal{F}^c(\mathbf{u}) = (\mathbf{f}_1^c, \ldots, \mathbf{f}_d^c)^T$ and $\mathcal{F}^v(\mathbf{u}, \nabla\mathbf{u}) = (\mathbf{f}_1^v, \ldots, \mathbf{f}_d^v)^T$, respectively. They are given by

$$\mathbf{f}_s^c(\mathbf{u}) = (\rho v_s, \rho v_1 v_s + \delta_{1s} p, \ldots, \rho v_d v_s + \delta_{ds} p, (\rho E + p) v_s)^T, \quad s = 1, \ldots, d$$

with $p = (\gamma - 1)\left[\rho E - \frac{1}{2}\|\mathbf{v}\|_2^2\right]$ denoting the pressure, the Poisson adiabatic constant $\gamma = 1.4$ (dry air), and

$$\mathbf{f}_s^v(\mathbf{u}, \nabla\mathbf{u}) = (0, \tau_{s1}, \ldots, \tau_{sd}, \sum_{i=1}^d \tau_{si} v_i + \mathcal{K} \frac{\partial T}{\partial x_s})^T, \quad s = 1, \ldots, d.$$

Here $T$ denotes the temperature function, $\mathcal{K}$ the heat conduction coefficient and $\tau$ is the viscous stress tensor. The Navier-Stokes equations are complemented by suitable Dirichlet and Neumann boundary conditions, see, for example, the monograph by Feistauer [7] for details. We use the Vijayasundaram numerical flux function. Since we are interested in steady solutions, the variable $t > 0$ plays the role of a pseudo-time variable and the temporal accuracy is irrelevant. When introducing the linearized equations below, we will describe its role as a globalization strategy.

**2.3. Discontinuous Galerkin discretization.** For the dGFEM discretization, we assume that $\Omega$ can be subdivided into a mesh $\mathcal{T}_h$, consisting of quadrilaterals (elements) $\kappa_j \neq \emptyset$, $j = 1, \ldots, n_t$, of characteristic size $h > 0$.

We consider a parametric discretization with piecewise discontinuous finite elements. Let $n_c$ stand for the number of equation components, i. e. $n_c = 1$ for the scalar model problem and $n_c = d + 2$ for the Navier-Stokes equations. The global discrete function space is given by

$$V_h^p := \{v \in \left[L^2(\Omega)\right]^{n_c} : v\big|_\kappa \circ \sigma_\kappa \in [\mathcal{Q}_p(\hat{\kappa})]^{n_c} \ \ \forall \kappa \in \mathcal{T}_h\},$$

where $\sigma_\kappa : \hat{\kappa} \to \kappa$ denotes a sufficiently smooth mapping from a reference element $\hat{\kappa} := [-1, 1]^d$ to the triangulation and $\mathcal{Q}_p(\hat{\kappa})$ is the set of polynomials of degree less

than or equal to $p$ in each variable. By $n_{h,p} = n_c(p+1)^d \operatorname{card} \mathcal{T}_h$ we denote the dimension of the vector space $V_h^p$.

The statement of the discrete problem requires some notation for the inter-element face terms [2]. We define the set $\mathcal{E}$ of all intersections between neighboring elements of the partition and between elements and the boundary $\Gamma$. Furthermore, we define the set of interior faces $\mathcal{E}_{\text{int}} = \{e \in \mathcal{E} : e \subset \Omega\}$ and its union $\Gamma_{\text{int}} := \{\mathbf{x} \in \Omega : \exists e \in \mathcal{E}_{\text{int}} \text{ with } \mathbf{x} \in e\}$.

As functions $v \in V_h^p$ are double-valued on $\mathbf{x} \in \Gamma_{\text{int}}$, we define $v_\kappa^+$, $\kappa \in \mathcal{T}_h$, to be the inner trace of $v$ on $\partial\kappa$. The outer trace $v_\kappa^-$ is defined as the inner trace $v_\kappa^+$ relative to the neigboring element. With this notation the jump and mean value of $v \in \left[H^1(\mathcal{T}_h)\right]^{n_c}$ on an edge $e \in \mathcal{E}_{\text{int}}$ are given by

$$[\![v]\!]_e = v_\kappa^+ \otimes \mathbf{n}_\kappa^+ + v_\kappa^- \otimes \mathbf{n}_\kappa^-, \quad \{v\}_e = \frac{1}{2}\left(v_\kappa^+ + v_\kappa^-\right),$$

where $\mathbf{n}_\kappa^\pm$ is the outer unit normal vector. In the following we will omit the subscript $(\cdot)_e$ for the sake of simplicity.

We also introduce a jump notation for the inflow part $\partial_-\kappa$ of an element boundary that is defined as $\partial_-\kappa = \{\mathbf{x} \in \partial\kappa \ : \ \beta(\mathbf{x}) \cdot \mathbf{n} < 0\}$. Then the jump of $u$ across $\partial_-\kappa \setminus \Gamma$ is defined by $[v]_\kappa := v_\kappa^+ - v_\kappa^-$. At the boundary $\Gamma$ the jump and average operators have to be modified, see [2].

For the scalar model problem (2.1) the discrete problem is given as follows: Find $u_h \in V_h^p$ such that

$$\mathcal{L}(u_h, v_h) \equiv B_1(u_h, v_h) + B_2(u_h, v_h) = \ell(v_h) \quad \forall v_h \in V_h^p \tag{2.3}$$

with bilinear forms $B_1$, $B_2$ corresponding to the diffusion and convection part, respectively. The latter uses a standard upwind term and is given by

$$\begin{aligned} B_2(u,v) &:= (\beta \cdot \nabla_h u, v)_\Omega - \sum_{\kappa \in \mathcal{T}_h} \int_{\partial_-\kappa} (\beta \cdot \mathbf{n})[u] v^+ \, ds, \\ \ell(v) &:= -\sum_{\kappa \in \mathcal{T}_h} \int_{\partial_-\kappa \cap (\Gamma_- \cup \Gamma_{\text{D}})} (\beta \cdot \mathbf{n}) g_{\text{D}} v^+ \, ds + (f, v)_\Omega, \end{aligned}$$

where we used the notation $\nabla_h$ for the broken gradient $(\nabla_h u)\big|_\kappa = \nabla\left(u\big|_\kappa\right)$, $\kappa \in \mathcal{T}_h$.

The formulation of the elliptic operator is the following:

$$\begin{aligned} B_1(u,v) := \varepsilon(\nabla_h u, \nabla_h v)_\Omega &- \varepsilon\langle [\![u]\!], \{\nabla_h v\}\rangle_{\Gamma \cup \Gamma_{\text{int}}} \\ &- \varepsilon\langle \{\nabla_h u\}, [\![v]\!]\rangle_{\Gamma \cup \Gamma_{\text{int}}} - \eta_e \{r_e([\![u]\!])\}. \end{aligned}$$

Here we have chosen the (modified) scheme of Bassi and Rebay [3], where $\eta_e$, $e \in \mathcal{E}_{\text{int}}$, is a stabilizing constant and $r_e$ denotes a local lifting operator which is defined in a weak manner by

$$\int_\Omega r_e(\varphi) \cdot \tau \, d\mathbf{x} = -\varepsilon \int_e \varphi \cdot \{\tau\} \, ds \quad \forall \tau \in [V_h^p]^d.$$

Various other consistent and stable dGFEM have been collected in [2].

A semi-linear form $\mathcal{L}(\mathbf{u}_h, \mathbf{v}_h)$ which is vector-valued, however similar in structure, can be derived for the compressible flow problem (2.2) via reformulation as a first order

system and the substitution of stabilizing numerical fluxes [3, 11]. With a basis in $V_h^p$ denoted by $\{\phi_\ell^p\}_{1\leq\ell\leq n_{h,p}}$ and an isomorphism $P_p : \mathbb{R}^{n_{h,p}} \to V_h^p$, $P_p\mathbf{x} = \sum_{i=1}^{n_{h,p}} x_i\phi_i^p$, the discrete problem is then given in terms of coefficient vectors as

$$L_p(\mathbf{x}) = \mathbf{f}_p, \tag{2.4}$$

where $L_p : \mathbb{R}^{n_{h,p}} \to \mathbb{R}^{n_{h,p}}$, $L_p(\mathbf{x})_j = \mathcal{L}(P_p\mathbf{x}, \phi_j^p)$, $j = 1, \ldots, n_{h,p}$. The right-hand side vector is expressed with the dual functional $P^*$ as $\mathbf{f}_p = P^*\ell \in \mathbb{R}^{n_{h,p}}$.

**3. The *hp*-multilevel method.** We are now concerned with the solution of equation (2.4). Algebraic type Newton-multigrid methods have been applied to discontinuous Galerkin discretizations on unstructured meshes with promising results [13]. However, for higher polynomial degrees and finer meshes the number of unknowns is too large for a standard Newton-type approach. A matrix-free Newton-Krylov method could be employed which uses finite differences and avoids the assembly of the complete Jacobian matrix $\underline{A}_{h,p} \in \mathbb{R}^{n_{h,p}\times n_{h,p}}$. Still, the question of an optimal preconditioner for the unstructured triangulation would remain open.

An alternative is the direct application of a multigrid (MG) method to the nonlinear system of equations. This requires a hierarchy of nonlinear discrete problems of varying complexity. The high-order dGFEM provides a nested sequence of vector spaces $V_h^p$, $V_h^{p-1}$, ... of decreasing polynomial degree in a natural way. The approach gives rise to a family of discrete nonlinear operators $L_p$, $L_{p-1}$, ..., $L_{p_{\min}}$.

Still, there is a drawback in this approach, since for many applications of moderate degree of approximation the number of levels is limited and the direct solution $\mathbf{x} \in \mathbb{R}^{n_{h,p_{\min}}}$ on the coarsest level $p_{\min}$ remains expensive. A remedy is to employ an $h$-optimal, algebraic type Newton-multigrid for the low-order solution on the unstructured mesh together with the nonlinear multigrid.

**3.1. Nonlinear multigrid.** As intergrid transfer operators in the nested level hierarchy $V_h^p, V_h^{p-1}, \ldots$, we choose the prolongation by natural injection $\iota_l : V_h^{l-1} \to V_h^l$, and the canonical restriction operator $\iota_l^*$, where the matrix formulation is given by $\tilde{I}_{l-1}^l = P_l^{-1}P_{l-1}$ and $\tilde{I}_l^{l-1} = (\tilde{I}_{l-1}^l)^T$. The dGFEM allows for arbitrary basis functions, thus a hierarchical basis $\{\phi_\ell^{l-1}\}_\ell \subset \{\phi_\ell^l\}_\ell$ can be employed which makes the injection particularly simple and memory efficient in implementation.

A major difference between the nonlinear multigrid algorithm and its linear counterpart is the need for a restricted nonlinear state vector. For this we take the orthogonal projection onto the space $V_h^{l-1}$ with respect to the $L^2$-norm. The corresponding restriction matrix is denoted by $\hat{I}_l^{l-1}$.

Further we define a nonlinear smoothing iteration

$$\mathbf{v}_l \mapsto \mathcal{S}_l(\mathbf{v}_l) := \mathbf{v}_l + \gamma\underline{W}_l^{-1}\left(\mathbf{f}_l - L_l(\mathbf{v}_l)\right) \tag{3.1}$$

with a damping factor $\gamma > 0$ and preconditioner matrix $\underline{W}_l$, which will be described in more detail in Section 3.3. We now state the

**Nonlinear V-cycle Multigrid Algorithm.** Given $\mathbf{f}_l \in \mathbb{R}^{n_{h,l}}$ and an initial guess $\mathbf{u}_{l,0} \in \mathbb{R}^{n_{h,l}}$, the output $\mathrm{NMG_V}(l, \mathbf{f}_l, \mathbf{u}_{l,0}; \nu, \mu)$ of the nonlinear V-cycle algorithm is an approximate solution of (2.4) obtained recursively as follows.

For $l = p_{\min}$, we take $\mathrm{NMG_V}(p_{\min}, \mathbf{f}_l, \mathbf{u}_{l,0}; \nu, \mu)$ to be $L_{p_{\min}}^{-1}(\mathbf{f}_l)$.

For $l > p_{\min}$, we obtain $\mathrm{NMG_V}(l, \mathbf{f}_l, \mathbf{u}_{l,0}; \nu, \mu)$ in three steps.

1. (Pre-Smoothing)
   Apply the block Jacobi scheme (3.1) $\nu$ times to compute $\mathbf{u}_{l,\nu} \in \mathbb{R}^{n_{h,l}}$.

2. (Coarse Grid Correction) Restrict the solution approximation $\mathbf{u}_{l-1,0} = \hat{I}_l^{l-1}\mathbf{u}_{l,\nu}$. Compute the right-hand side

$$\mathbf{f}_{l-1} = \tilde{I}_l^{l-1}\mathbf{f}_l + \left[L_{l-1}(\mathbf{u}_{l-1,0}) - \tilde{I}_l^{l-1}L_l(\mathbf{u}_{l,\nu})\right],$$

and solve the coarse grid equation $L_{l-1}(\mathbf{v}) = \mathbf{f}_{l-1}$ using the $(l-1)$-st level V-cycle with $\mathbf{u}_{l-1,0}$ as the initial guess. Then transfer the coarse level error back to the $l$-th level, and update the solution approximation. In total,

$$\mathbf{u}_{l,\nu+1} = \mathbf{u}_{l,\nu} + \tilde{I}_{l-1}^{l}\left[\mathrm{NMG_V}(l-1, \mathbf{f}_{l-1}, \mathbf{u}_{l-1,0}; \nu, \mu) - \mathbf{u}_{l-1,0}\right].$$

3. (Post-Smoothing)
Apply the block Jacobi scheme (3.1) $\mu$ times to compute $\mathbf{u}_{l,\nu+\mu+1}$.

Finally, set $\mathrm{NMG_V}(l, \mathbf{f}_l, \mathbf{u}_{l,0}; \nu, \mu) = \mathbf{u}_{l,\nu+\mu+1}$.

A nested iteration strategy $\mathrm{NMG_V}(p_{\min}, \cdot)$, $\mathrm{NMG_V}(p_{\min}+1, \cdot)$, ... can be employed to obtain an initial solution approximation $\mathbf{u}_{p,0}$. Experiments indicate that the method scales with the polynomial degree $p$, cf. Section 4. A closer look at the method, however, reveals that the computational cost of the nonlinear smoothing iteration (3.1) is not independent of $p$. Nevertheless, in practice this scaling issue remains rather theoretical for moderate degrees of approximation, e. g. $p \leq 10$.

**3.2. Newton-multigrid solver.** The multigrid algorithm $\mathrm{NMG_V}$ poses a nonlinear low-order equation

$$L_{p_{\min}}(\mathbf{v}) = \mathbf{f}_{p_{\min}}, \mathbf{v} \in V_h^p, \tag{3.2}$$

that remains a challenging problem especially on unstructured meshes.

We choose a Newton-multigrid iteration to solve (3.2), i. e. we approximate the solution $\mathbf{u} \in V_h^{p_{\min}}$ with a sequence of linearized problems. More precisely, we linearize equation (2.2) with the semi-implicit Euler scheme,

$$\underline{A}\mathbf{d}^i \equiv \left[\frac{1}{\Delta t}\underline{M} + \underline{A}_{h,p_{\min}}(\mathbf{u}^n)\right]\mathbf{d}^i = \mathbf{b}^i, \quad \mathbf{u}^{i+1} := \mathbf{u}^i + s\mathbf{d}^i,\ i = 1, 2, \ldots,$$

where $\underline{A}_{h,p_{\min}}(\mathbf{u}^n)$ is the coarse-level Jacobian, $\underline{M}$ is the block diagonal mass matrix and $\mathbf{b}^i := \mathbf{f}_{p_{\min}} - L_{p_{\min}}(\mathbf{u}^i)$. Further, $s \in \mathbb{R}^+$ is a suitable damping factor and $\Delta t$ is a time step which is increased during the nonlinear iteration. This ensures that a standard Newton scheme is recovered after an initial transient period. Each linearized problem $\underline{A}\mathbf{d}^i = \mathbf{b}^i$ is solved only approximately by a few V-cycles of a linear multigrid method, resulting in an inexact Newton method with a linear rate of convergence.

We resort to the smoothed aggregation (SA) approach [15] to construct a sequence of coarse vector spaces $M_1 \subset M_2 \subset \cdots \subset M_J \equiv \mathbb{R}^{n_{h,p_{\min}}}$, since a natural hierarchy of spatial discretizations is not available for unstructured tesselations. The basic idea of the SA variational multigrid method is to partition the elements of the mesh $\mathcal{T}_h$ into small disjoint clusters $a_m$, $m = 1, \ldots, n_k$, and to construct the (coefficient vector) spaces $M_k$ from the basis of characteristic functions $\chi_{a_m}$, whose energy norm is improved by a suitable smoothing polynomial. Additionally, we get a family of coarse grid operators and right-hand side vectors $\underline{A}_k$, $\mathbf{b}_k$, $k = 1, \ldots, J$.

More precisely, we employ a Petrov-Galerkin SA technique [9, 13], i. e. we set the coarse level operators $\underline{A}_{k-1}$ recursively as

$$\underline{A}_{k-1} = I_k^{k-1}\underline{A}_k(\underline{Id} - \lambda_k^{-1}\underline{A}_k)I_{k-1}^k, k = J, \ldots, 1, \quad \underline{A}_J := \underline{A},$$

where $I_k^{k-1} \in \mathbb{R}^{n_{k-1} \times n_k}$ denotes the trivial zero-one restriction onto the agglomerates, $I_{k-1}^k$ the corresponding prolongation, $\lambda_k$ is an approximation of the spectral radius $\varrho(\underline{A}_k)$ obtained via the Lanczos method, and $\underline{Id}$ is the identity matrix. We state the

**Linear V-cycle Multigrid Algorithm.** Given a right-hand side $\mathbf{b}_l \in M_l$ and an initial solution $\mathbf{x}_l \in M_l$, the iterate $\mathrm{MG_V}(l, \mathbf{b}_l, \mathbf{x}_l; \bar{\mu}, \bar{\nu})$ of the nonsymmetric *linear* V-cycle algorithm is obtained as follows.

For $l = 1$, perform a direct solution $\mathrm{MG_V}(l, \mathbf{b}_l, \mathbf{x}_l) := \underline{A}_1^{-1}\mathbf{b}_1$.

For $l > 1$, take $\mathrm{MG_V}(l, \mathbf{b}_l, \mathbf{x}_l; \bar{\mu}, \bar{\nu})$ as the result $\mathbf{x}_l$ of the following steps.

1. Perform $\bar{\nu}$ iterations of $\mathbf{x}_l \leftarrow (\underline{Id} - \underline{W}_l\underline{A}_l)\,\mathbf{x}_l + \underline{W}_l\mathbf{b}_l$ (Pre-Smoothing).
2. Set $\mathbf{b}_{l-1} = I_l^{l-1}\,(\mathbf{b}_l - \underline{A}_l\mathbf{x}_l)$ and compute $\mathbf{x}_l \leftarrow \mathbf{x}_l + I_{l-1}^l\mathbf{x}_{l-1}$, where $\mathbf{x}_{l-1} := \mathrm{MG_V}(l-1, \mathbf{b}_{l-1}, \mathbf{0})$ (Coarse Grid Correction).
3. Perform $\bar{\mu}$ iterations of $\mathbf{x}_l \leftarrow (\underline{Id} - \underline{W}_l\underline{A}_l)\,\mathbf{x}_l + \underline{W}_l\mathbf{b}_l$ (Post-Smoothing).

If $L_l$ denotes a linear operator, the nonlinear multigrid $\mathrm{NMG_V}$ coincides with this algorithm. The algorithm may be used as a preconditioner for a Krylov subspace method like GMRES to increase robustness. For elliptic problems on unstructured grids, the regularity-free theory of Bramble [6] can be applied to prove the optimality of the algorithm. However, for advection-diffusion systems convergence theory is much less developed even in the linear case.

**3.3. Block-implicit smoothing iteration.** Finally, we describe the smoothing iteration (3.1) in more detail. The block Jacobi iteration is known to be efficient in damping out the oscillatory part of the error for the classical geometric multigrid. We assume that the same holds true for the nonlinear multigrid algorithm, as well as for the aggregation multigrid method discussed in Section 3.2.

For the problems (2.1), (2.2), where advection terms are present, the blocks are formed as follows. First, the elements of $\mathcal{T}_h$ are organized as sequences (*lines*). Only face integrals between elements of a common line are considered. This corresponds to a block-diagonal approximation of the Jacobian $\underline{A}_{h,l} = \partial_{\mathbf{u}} L_l$,

$$\underline{W}_l^{-1} := \mathrm{blockdiag}\left(\underline{A}_{h,l,i}^{-1}\right)_{i=1,\ldots,N},$$

where each $\underline{A}_{h,l,i}$ is a block tridiagonal system and $N$ denotes the number of lines. It is observed experimentally, and has been supported by local Fourier analysis, that lines should follow strong couplings in the stiffness matrix, where by *strong coupling* we mean a measure for interdependencies between the elements of the triangulation. We employ a heuristic similar to Okusanya [8] to determine the blocks of the iteration matrices $\underline{W}_l$. Figure 4.1 shows an example decomposition in the vicinity of a NACA 0012 airfoil.

For difficult cases the stability of the nonlinear smoother can be improved by applying a sequence of smoothing steps with varying damping parameters. The resulting semi-iterative method is interpreted as an $s$-stage Runge-Kutta scheme

$$\begin{aligned}
&\mathbf{v}_{l,0} := \mathbf{v}_l,\\
&\left[(\alpha_i\Delta t)^{-1}\underline{M} + \underline{W}_l\right](\mathbf{v}_{l,i} - \mathbf{v}_{l,i-1}) =\\
&\qquad (\alpha_i\Delta t)^{-1}\underline{M}(\mathbf{v}_l - \mathbf{v}_{l,i-1}) - (\mathbf{f}_l - L_l(\mathbf{v}_{l,i-1}))\,, \quad i = 1,\ldots,s,\\
&\mathbf{v}_l \leftarrow \mathbf{v}_{l,s}.
\end{aligned}$$

For the laminar flow example in Section 4.2 we choose $s = 5$ coefficients from [4] as $\{\alpha_i\}_{i=1,\ldots,5} = \left\{\frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, 1\right\}$.

**4. Numerical results.** We now turn to the numerical experiments. For the sake of simplicity we have applied the simple V-cycle forms of the proposed algorithms and restricted ourselves to the case of hexahedral meshes and straight boundary geometries. The computational results were obtained using the DLR `PADGE` code [10] on a parallel computer with distributed memory. CPU time results are not included in the description of the numerical tests because no attempt was made to optimize the performance of the current solver for the problems considered.

**4.1. Linear model problem.** We consider (2.1) and choose the parameters of a three-dimensional example problem from [5]. The vector field $\beta$, which is locally varying in the computational domain $\Omega = (0,1)^3$ and not grid-aligned, is given by

$$\beta(\mathbf{x}) = \frac{w(\mathbf{x})\mathrm{Pe}}{\sqrt{3}}\,[1,1,1]^T, \text{ where } w(\mathbf{x}) := \begin{cases} 1 & x_1 + x_2 - 2x_3 \le 0 \\ 0 & x_1 + x_2 - 2x_3 \ge 1 \\ (x_1 + x_2 - 2x_3)^2 & \text{elsewhere} \end{cases}$$

and $\mathrm{Pe} := 10$ denotes the Péclet number. We further assume isotropic diffusion $\varepsilon = 1$ and a constant right hand side $f = 1$. The boundary conditions are of the Dirichlet type with

$$g_{\mathrm{D}} := \begin{cases} 1 & 0 \le x_1, x_2 \le \frac{1}{2}, x_3 = 0 \\ 0 & \text{elsewhere.} \end{cases}$$

Figure 4.2 shows the contour lines of the approximate solution of equation (2.1) on a slice along the main diagonal. The computational domain contains areas of diffusion as well as of dominating convection.

For the convergence study we use uniform partitionings of the domain into tensor product elements, though the algorithm does not exploit this regular structure. The considered grid sizes and polynomial degrees are $h \in \left\{\frac{1}{8}, \frac{1}{16}, \frac{1}{32}\right\}$, $p \in \{0, \ldots, 3\}$, i.e. the number of unknowns $n_{h,p}$ grows up to $2.1 \cdot 10^6$. The behavior of the two different solver components is demonstrated by separate computations.

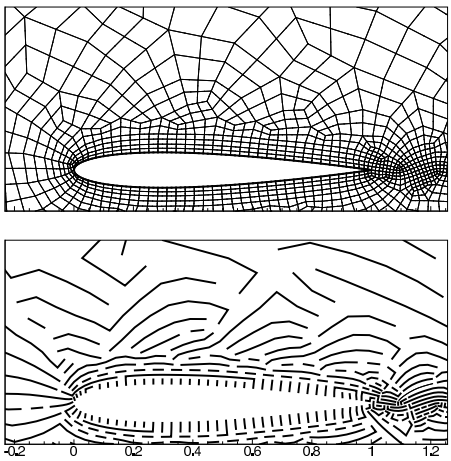


FIGURE 4.1. *NACA0012 test case. Computational mesh (zoom) and example of a typical line decomposition for viscous flow.*

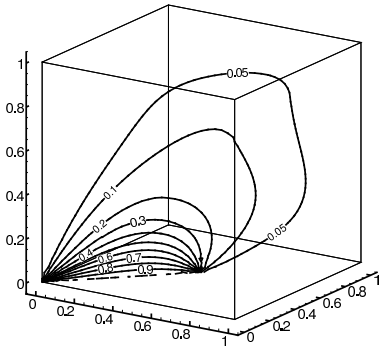


FIGURE 4.2. *Linear advection-diffusion test case. Contour lines of the solution on a slice along the main diagonal.*

**Smoothed Aggregation Multigrid.** Table 4.1a shows the result of GMRES-accelerated V-cycles with 1 pre- and post-smoothing step. The table contains both the average residual reduction in the $\ell_2$-norm, $r_{avg} := \left(\frac{\|\mathbf{r}^N\|_2}{\|\mathbf{r}^0\|_2}\right)^{\frac{1}{N}}$, and the condition number of the preconditioned system estimated during the Krylov iteration. When looking at each row separately, one observes that the results reflect the expected behavior of a multigrid method fairly well, i. e. the number of iterations is more or less bounded. This holds true despite the presence of an advection term, where optimal convergence is no longer covered by variational multigrid theory. On the finest level with bilinear elements, however, a slight deviation from the exact boundedness occurs. We interpret this as a result of the heuristic agglomeration strategy. On the other hand, the convergence of the SA multigrid is getting slow with increasing polynomial degree.

**Multi-$p$ method.** We now investigate the behavior of the multi-$p$ method for the linear test case. The V-cycle with 1 pre- and post-smoothing step (element-wise Jacobi smoother with damping $\gamma := 0.75$) is applied up to a polynomial degree $p = 3$ with a coarse level problem for $p_{\min} = 0$. Table 4.1b contains the average residual convergence rate $r_{avg}$, where a fixed number of 20 smoothing steps is performed to approximate the solution of the coarse level equation (3.2). Additionally Table 4.1b shows the convergence rates for the case that problem (3.2) is solved up to machine accuracy (denoted by $1(\epsilon)$, $2(\epsilon)$ $3(\epsilon)$).

The convergence rates clearly indicate that the number of V-cycles does not increase with the polynomial degree. On the other hand the algorithm does not scale with the grid size, if the coarse level problem is not solved accurately. Even when computing the coarse level problem to machine accuracy, a slight $h$-dependence is observed on the relatively coarse meshes. Further computations however show that this effect disappears on finer grids. Thus the multi-$p$ algorithm constitutes a feasible approach to solve high-order numerical systems, provided that a scalable coarse level solver is at hand. This motivates the combination of both algorithms in the next example.

**4.2. Flow past NACA 0012 airfoil.** In the second test case the subsonic flow past a NACA 0012 airfoil was computed for the Mach number $\mathrm{Ma}_\infty = 0.5$, with the Reynolds number $\mathrm{Re} = 500$ and an angle of attack $\alpha = 2°$. The walls of the airfoil are adiabatic and we choose freestream conditions $\mathbf{u}_{p_{\min},0} \in \mathbb{R}^{n_{h,l}}$ as the initial guess. On the far field boundary, which is of circular shape with a radius of 50 chord lengths, subsonic inflow and outflow conditions are prescribed, cf. [7]. The conditions for this simple but well investigated numerical experiment are identical to the ADIGMA MTC3S test case [1]. Figure 4.1 shows the unstructured computational mesh that contains 2304 hexahedral elements and was constructed based on data from [1]. Equation (2.2) was discretized using discontinuous elements with a piecewise quadratic boundary.

The computation employed a nested sequence of nonlinear V-cycles up to a uniform polynomial degree $p = 5$ with ca. $3.3 \cdot 10^5$ unknowns. A single Newton step was performed on the coarse level $p_{\min} = 0$ to solve equation (3.2) with 10 linear multigrid cycles. The resulting Mach number distribution at the leading and the trailing edge of the airfoil is depicted in Figure 4.3. In terms of the flow physics the resolution is overly fine, however, it is the purpose of this numerical test to show the convergence behavior of the iterative method which proves to be sufficiently stable.

Instead of employing a relative stop criterion in each stage of the nested multilevel

iteration, a fixed number of 7 V-cycles with 2 pre- and post-smoothing steps was performed. Figure 4.4 shows a graph of the (continuous) $L^2(\Omega)$ norm of the residual $\mathbf{r} := \mathbf{f}_p - L_p(\mathbf{x})$ with respect to the V-cycle iteration index. The absolute residual measured in the Euclidean norm converges in about 35 V-cycles ($tol \leq 10^{-8}$), 14 steps of them are performed on the fine level $p = 5$. Additionally, the plot contains the plots of lower-order approximations iterating to full convergence (dotted lines). The CFL number of the linearized implicit smoother is adapted according to the development of the residual. As a consequence the convergence of the multilevel algorithm accelerates during the solution process.

**Acknowledgements.** F. Prill and R. Hartmann acknowledge the partial financial support of both the President's Initiative and Networking Fund of the Helmholtz Association of German Research Centres and the European project ADIGMA [1]. The research of M. Lukáčová was supported partially by the European Graduate School Differential Equations with Applications in Science and Engineering (DEASE), MEST-CT-2005-021122. The authors gratefully acknowledge these supports.

## REFERENCES

[1] *ADIGMA. Adaptive higher-order variational methods for aerodynamic applications in industry.* Specific Targeted Research Project no. 30719 supported by European Commission. `http://www.dlr.de/as/en/Desktopdefault.aspx/tabid-2035`.

[2] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM J. Numer. Anal., 39 (2002), pp. 1749–1779.

[3] F. Bassi, A. Crivellini, S. Rebay, and M. Savini, *Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and $k-\omega$ turbulence model equations*, Comput. Fluids, 34 (2005), pp. 507–540.

[4] F. Bassi, A. Ghidoni, S. Rebay, and P. Tesini, *High-order accurate p-multigrid discontinuous Galerkin solution of the Euler equations*, Int. J. Numer. Methods Fluids, (2008).

[5] J. Bey, *Finite-Volumen- und Mehrgitterverfahren für elliptische Randwertprobleme*, PhD thesis, Tübingen: Univ. Tübingen, Mathematische Fakultät, 1997.

[6] J. H. Bramble, *Multigrid methods.*, vol. 294 of Pitman Research Notes in Mathematics Series, Longman Scientific & Technical, Harlow, 1993.

[7] M. Feistauer, J. Felcman, and I. Straškraba, *Mathematical and computational methods for compressible flow.*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2003.

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

[9] H. Guillard, A. Janka, and P. Vaněk, *Analysis of an Algebraic Petrov-Galerkin Smoothed Aggregation Multigrid Method*, Appl. Numer. Math., 58 (2006), pp. 1861–1874.

[10] R. Hartmann, J. Held, T. Leicht, and F. Prill, `PADGE`*, Parallel Adaptive Discontinuous Galerkin Environment.* Technical reference, DLR, Braunschweig, 2008.

[11] R. Hartmann and P. Houston, *An Optimal Order Interior Penalty Discontinuous Galerkin Discretization of the Compressible Navier-Stokes Equations*, J. Comput. Phys., 227 (2008), pp. 9670–9685.

[12] C. Lasser and A. Toselli, *Overlapping preconditioners for discontinuous Galerkin approximations of second order problems*, in Thirteenth International Conference on Domain Decomposition Methods, N. Debit, M. Garbey, R. Hoppe, J. Périaux, D. Keyes, and Y. Kuznetsov, eds., 2001.

[13] F. Prill, M. L. Medvidova, and R. Hartmann, *Smoothed Aggregration Multigrid for the Discontinuous Galerkin Method.* submitted to SIAM J. Sci. Comp., 2008.

[14] H.-G. Roos, M. Stynes, and L. Tobiska, *Numerical methods for singularly perturbed differential equations. Convection-diffusion and flow problems.*, vol. 24 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 1996.

[15] P. Vaněk, M. Brezina, and J. Mandel, *Convergence of algebraic multigrid based on smoothed aggregation*, UCD/CCD Report 126, University of Colorado at Denver, 1998.

| $p$ \ $h$ | $\frac{1}{8}$ | $\frac{1}{16}$ | $\frac{1}{32}$ |
|---|---|---|---|
| 0 | 0.004 (1.04) | 0.016 (1.07) | 0.020 (1.09) |
| 1 | 0.387 (4.50) | 0.444 (5.11) | 0.492 (5.92) |
| 2 | 0.404 (5.20) | 0.426 (5.30) | 0.437 (5.48) |
| 3 | 0.664 (23.94) | 0.694 (20.74) | 0.702 (25.32) |

(a) SA multigrid algorithm.

| $p$ \ $h$ | $\frac{1}{8}$ | $\frac{1}{16}$ | $\frac{1}{32}$ |
|---|---|---|---|
| | – | – | – |
| 1 | 0.60 | 0.67 | 0.86 |
| 2 | 0.57 | 0.65 | 0.85 |
| 3 | 0.55 | 0.64 | 0.84 |
| $1(\epsilon)$ | 0.60 | 0.60 | 0.60 |
| $2(\epsilon)$ | 0.57 | 0.58 | 0.58 |
| $3(\epsilon)$ | 0.54 | 0.57 | 0.57 |

(b) Multi-$p$ method.

TABLE 4.1

*Separate convergence study of the two solver components for the linear test case. The tables show the average reduction of the residual w. r. t. the $\ell_2$-norm. The left table also contains the condition number estimates of the preconditioned system (in parentheses).*

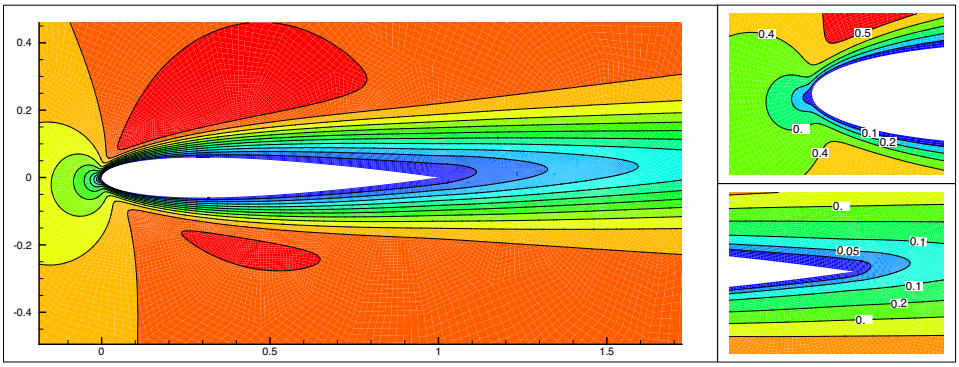


FIGURE 4.3. *NACA0012 test case. Mach number distribution with details around the leading and trailing edges (right).*

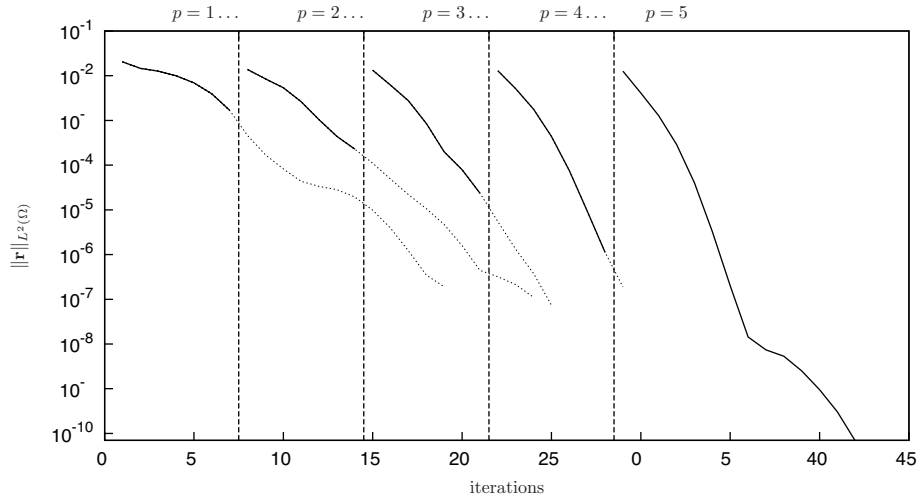


FIGURE 4.4. *NACA0012 test case. Residual convergence history.*