Computers and Fluids 163 (2018) 50–53

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Technical note

The Montecinos–Balsara ADER-FV polynomial basis: Convergence properties & extension to non-conservative multidimensional systems

Haran Jackson

Cavendish Laboratory, JJ Thomson Ave, Cambridge CB3 0HE, UK

ARTICLE INFO

Article history:
Received 20 October 2017
Revised 18 December 2017
Accepted 28 December 2017
Available online 29 December 2017

Keywords:
ADER
Finite Volume
Galerkin
Eigenvalues
Convergence

ABSTRACT

Hyperbolic systems of PDEs can be solved to arbitrary orders of accuracy by using the ADER Finite Volume method. These PDE systems may be non-conservative and non-homogeneous, and contain stiff source terms. ADER-FV requires a spatio-temporal polynomial reconstruction of the data in each space-time cell, at each time step. This reconstruction is obtained as the root of a nonlinear system, resulting from the use of a Galerkin method. It was proved in [7] that for traditional choices of basis polynomials, the eigenvalues of certain matrices appearing in these nonlinear systems are always 0, regardless of the number of spatial dimensions of the PDEs or the chosen order of accuracy of the ADER-FV method. This guarantees fast convergence to the Galerkin root for certain classes of PDEs.

In Montecinos and Balsara [8] a new, more efficient class of basis polynomials for the one-dimensional ADER-FV method was presented. This new class of basis polynomials, originally presented for conservative systems, is extended to multidimensional, non-conservative systems here, and the corresponding property regarding the eigenvalues of the Galerkin matrices is proved.

1. Background

ADER-FV methods were first devised by Toro and collaborators (see Toro et al. [11] and also Toro and Titarev [12], Titarev and Toro [10]). Dumbser et al. [2] obviated the need for the cumbersome analytic work required by the Cauchy–Kowalewski procedure by use of a Galerkin predictor. Although ADER-FV methods have been very successful in solving a large variety of different hyperbolic systems (e.g. see Dumbser et al. [3], Balsara et al. [1], Hidalgo and Dumbser [6], Zanotti and Dumbser [13]), they remain relatively computationally expensive.

Montecinos and Balsara [8] have proposed a new, more efficient class of basis polynomials. While the method was given for conservative, one-dimensional systems in the original paper, it is extended here to general non-conservative, multidimensional systems.

2. Extension of the Montecinos–Balsara formulation

Take a non-homogeneous, non-conservative, hyperbolic system of the form:

$$\frac{\partial \boldsymbol{Q}}{\partial t} + \nabla \cdot \vec{\boldsymbol{F}}(\boldsymbol{Q}) + \vec{\boldsymbol{B}}(\boldsymbol{Q}) \cdot \nabla \boldsymbol{Q} = \boldsymbol{S}(\boldsymbol{Q}) \tag{1}$$

where $\boldsymbol{Q}$ is the vector of conserved variables, $\vec{\boldsymbol{F}} = (\boldsymbol{F_1}, \boldsymbol{F_2}, \boldsymbol{F_3})$ and $\vec{\boldsymbol{B}} = (B_1, B_2, B_3)$ are respectively the conservative nonlinear fluxes and matrices corresponding to the purely non-conservative components of the system, and $\boldsymbol{S}(\boldsymbol{Q})$ is the algebraic source vector.

Define spatial variables $x^{(1)}$, $x^{(2)}$, $x^{(3)}$. Take the space-time cell $C = [x_{i_1}^{(1)}, x_{i_1+1}^{(1)}] \times [x_{i_2}^{(2)}, x_{i_2+1}^{(2)}] \times [x_{i_3}^{(3)}, x_{i_3+1}^{(3)}] \times [t_n, t_{n+1}]$. Define the scaled spatial and temporal variables:

$$\chi^{(k)} = \frac{x^{(k)} - x_{i_k}^{(k)}}{x_{i_k+1}^{(k)} - x_{i_k}^{(k)}} \tag{2a}$$

$$\tau = \frac{t - t_n}{t_{n+1} - t_n} \tag{2b}$$

Thus, C becomes:

$$\left(\chi^{(1)}, \chi^{(2)}, \chi^{(3)}, \tau\right) \in [0,1]^4 \tag{3}$$

By rescaling $\vec{\boldsymbol{F}}, \vec{\boldsymbol{B}}, \boldsymbol{S}$ by the appropriate constant factors, and defining $\tilde{\nabla} = (\partial_{\chi^{(1)}}, \partial_{\chi^{(2)}}, \partial_{\chi^{(3)}})$, within C Eq. (1) becomes:

$$\frac{\partial \boldsymbol{Q}}{\partial \tau} + \tilde{\nabla} \cdot \vec{\boldsymbol{F}}(\boldsymbol{Q}) + \vec{\boldsymbol{B}}(\boldsymbol{Q}) \cdot \tilde{\nabla} \boldsymbol{Q} = \boldsymbol{S}(\boldsymbol{Q}) \tag{4}$$

A basis $\{\psi_0, \ldots, \psi_N\}$ of P_N and inner product $\langle \cdot, \cdot \rangle$ are now required to produce a polynomial reconstruction of $\boldsymbol{Q}$ within C. Traditionally, this basis has been chosen to be either nodal ($\psi_i(\alpha_j) =$

E-mail address: hj305@cam.ac.uk

https://doi.org/10.1016/j.compfluid.2017.12.018
0045-7930/

δ_{ij} where $\{\alpha_0, \dots, \alpha_N\}$ are a set of nodes, e.g. the Gauss–Legendre abscissae - see Dumbser et al. [2]), or modal (e.g. the Legendre polynomials - see Balsara et al. [1]).

Montecinos and Balsara [8] take the following approach. $\langle \cdot, \cdot \rangle$ is taken to be the usual integral product on [0, 1]. Supposing that $N = 2n+1$ for some $n \in \mathbb{N}$, Gauss–Legendre nodes $\{\alpha_0, \dots, \alpha_n\}$ are taken. The basis $\Psi = \{\psi_0, \dots, \psi_N\} \subset P_N$ is taken with the following properties for $i = 0, \dots, n$:

$$\begin{cases} \psi_i(\alpha_j) = \delta_{ij} & \psi_i'(\alpha_j) = 0 \\ \psi_{n+1+i}(\alpha_j) = 0 & \psi_{n+1+i}'(\alpha_j) = \delta_{ij} \end{cases} \tag{5}$$

Define the following subsets:

$$\Psi^0 = \{\psi_i : 0 \le i \le n\} \tag{6a}$$

$$\Psi^1 = \{\psi_i : n+1 \le i \le 2n+1\} \tag{6b}$$

The WENO method (as used in Dumbser et al. [5]) produces an order-N polynomial reconstruction $w(\chi^{(1)}, \chi^{(2)}, \chi^{(3)})$ of the data at time t_n in $[x_{i_1}^{(1)}, x_{i_1+1}^{(1)}] \times [x_{i_2}^{(2)}, x_{i_2+1}^{(2)}] \times [x_{i_3}^{(3)}, x_{i_3+1}^{(3)}]$. It is used as initial data in the problem of finding the Galerkin predictor. Taking representation $w = w_{abc}\psi_a(\chi^{(1)})\psi_b(\chi^{(2)})\psi_c(\chi^{(3)})$ we have for $0 \le i, j, k \le n$:

$$w_{ijk} = w(\alpha_i, \alpha_j, \alpha_k) \tag{7a}$$

$$w_{(n+i+1)jk} = \partial_{\chi^{(1)}} w(\alpha_i, \alpha_j, \alpha_k) \tag{7b}$$

$$w_{i(n+j+1)k} = \partial_{\chi^{(2)}} w(\alpha_i, \alpha_j, \alpha_k) \tag{7c}$$

$$w_{ij(n+k+1)} = \partial_{\chi^{(3)}} w(\alpha_i, \alpha_j, \alpha_k) \tag{7d}$$

Take the following temporal nodes, where $\tau_1, \dots, \tau_N$ are the usual Legendre–Gauss nodes on [0, 1] and $\tau_0 = 0$ or $\tau_0 = 1$ if we are performing a continuous Galerkin / discontinuous Galerkin reconstruction, respectively:

$$\{\tau_0, \dots, \tau_N\} \tag{8}$$

Define $\Phi = \{\phi_0, \dots, \phi_N\} \subset P_N$ to be the set of Lagrange interpolating polynomials on the temporal nodes. We now define the spatio-temporal polynomial basis $\Theta = \Phi \otimes \Psi \otimes \Psi \otimes \Psi = \{\theta_\beta\}$ for $0 \le \beta \le (N+1)^4 - 1$. Define subsets $\Theta^{\iota\xi\kappa} = \Phi \otimes \Psi^\iota \otimes \Psi^\xi \otimes \Psi^\kappa = \{\theta_\mu^{\iota\xi\kappa}\}$ where $\iota, \xi, \kappa \in \{0, 1\}$ for $0 \le \mu \le (N+1)(n+1)^3 - 1$.

Denoting the Galerkin predictor by $\boldsymbol{q}$, take the following set of approximations:

$$\boldsymbol{Q} \approx \theta_\beta \boldsymbol{q}_\beta = \theta_\mu^{\iota\xi\kappa} \boldsymbol{q}_\mu^{\iota\xi\kappa} \tag{9a}$$

$$\vec{\boldsymbol{F}}(\boldsymbol{Q}) \approx \theta_\beta \vec{\boldsymbol{F}}_\beta = \theta_\mu^{\iota\xi\kappa} \vec{\boldsymbol{F}}_\mu^{\iota\xi\kappa} \tag{9b}$$

$$\vec{\boldsymbol{B}}(\boldsymbol{Q}) \cdot \tilde{\nabla}\boldsymbol{Q} \approx \theta_\beta \boldsymbol{B}_\beta = \theta_\mu^{\iota\xi\kappa} \boldsymbol{B}_\mu^{\iota\xi\kappa} \tag{9c}$$

$$\boldsymbol{S}(\boldsymbol{Q}) \approx \theta_\beta \boldsymbol{S}_\beta = \theta_\mu^{\iota\xi\kappa} \boldsymbol{S}_\mu^{\iota\xi\kappa} \tag{9d}$$

for some coefficients $\boldsymbol{q}_\beta, \vec{\boldsymbol{F}}_\beta, \boldsymbol{B}_\beta, \boldsymbol{S}_\beta$. The *nodal basis representation* is used for the coefficients of Θ^{000}:

$$\vec{\boldsymbol{F}}_\mu^{000} = \vec{\boldsymbol{F}}(\boldsymbol{q}_\mu^{000}) \tag{10a}$$

$$\boldsymbol{B}_\mu^{000} = B_1(\boldsymbol{q}_\mu^{000})\boldsymbol{q}_\mu^{100} + B_2(\boldsymbol{q}_\mu^{000})\boldsymbol{q}_\mu^{010} + B_3(\boldsymbol{q}_\mu^{000})\boldsymbol{q}_\mu^{001} \tag{10b}$$

$$\boldsymbol{S}_\mu^{000} = \boldsymbol{S}(\boldsymbol{q}_\mu^{000}) \tag{10c}$$

In general, we have:

$$\vec{\boldsymbol{F}}_\mu^{\iota\xi\kappa} = \partial_\chi^\iota \partial_\upsilon^\xi \partial_\zeta^\kappa (\vec{\boldsymbol{F}}(\boldsymbol{Q})) \tag{11a}$$

$$\boldsymbol{B}_\mu^{\iota\xi\kappa} = \partial_\chi^\iota \partial_\upsilon^\xi \partial_\zeta^\kappa (\vec{\boldsymbol{B}}(\boldsymbol{Q}) \cdot \tilde{\nabla}\boldsymbol{Q}) \tag{11b}$$

$$\boldsymbol{S}_\mu^{\iota\xi\kappa} = \partial_\chi^\iota \partial_\upsilon^\xi \partial_\zeta^\kappa (\boldsymbol{S}(\boldsymbol{Q})) \tag{11c}$$

where the right-hand-side is evaluated at the nodal point corresponding to μ. The full expressions are omitted here for brevity's sake, but note that for a one-dimensional system:

$$\boldsymbol{F}_1{}_\mu^{100} = \frac{\partial \boldsymbol{F}(\boldsymbol{q}_\mu^{000})}{\partial \boldsymbol{Q}} \cdot \boldsymbol{q}_\mu^{100} \tag{12a}$$

$$\begin{aligned} \boldsymbol{B}_\mu^{100} = &\left(\frac{\partial B_1(\boldsymbol{q}_\mu^{000})}{\partial \boldsymbol{Q}} \cdot \boldsymbol{q}_\mu^{100} \right) \cdot \boldsymbol{q}_\mu^{100} + B_1(\boldsymbol{q}_\mu^{000}) \\ &\cdot \left(\frac{\partial^2 \theta_\kappa^{000}(\chi_\mu, \tau_\mu)}{\partial \chi^2} \boldsymbol{q}_\mu^{000} + \frac{\partial^2 \theta_\kappa^{100}(\chi_\mu, \tau_\mu)}{\partial \chi^2} \boldsymbol{q}_\mu^{100} \right) \end{aligned} \tag{12b}$$

$$\boldsymbol{S}_\mu^{100} = \frac{\partial \boldsymbol{S}(\boldsymbol{q}_\mu^{000})}{\partial \boldsymbol{Q}} \cdot \boldsymbol{q}_\mu^{100} \tag{12c}$$

where χ_μ, τ_μ are the spatial and temporal coordinates where $\theta_\mu^{100} = 0$ and $\partial_\chi \theta_\mu^{100} = 1$. Note that $\frac{\partial B_1}{\partial \boldsymbol{Q}}$ is a rank 3 tensor.

Consider functions f, g of the following form:

$$f(\tau, \chi^{(1)}, \chi^{(2)}, \chi^{(3)}) = f_\tau(\tau) f_1(\chi^{(1)}) f_2(\chi^{(2)}) f_3(\chi^{(3)}) \tag{13a}$$

$$g(\tau, \chi^{(1)}, \chi^{(2)}, \chi^{(3)}) = g_\tau(\tau) g_1(\chi^{(1)}) g_2(\chi^{(2)}) g_3(\chi^{(3)}) \tag{13b}$$

Define the following integral operators:

$$[f, g]^t = f_\tau(t) g_\tau(t) \langle f_1, g_1 \rangle \langle f_2, g_2 \rangle \langle f_3, g_3 \rangle \tag{14a}$$

$$\{f, g\} = \langle f_\tau, g_\tau \rangle \langle f_1, g_1 \rangle \langle f_2, g_2 \rangle \langle f_3, g_3 \rangle \tag{14b}$$

Multiplying (9b) by test function θ_α, using the polynomial approximations for $\boldsymbol{Q}, \vec{\boldsymbol{F}}, \vec{\boldsymbol{B}}, \boldsymbol{S}$, and integrating over space and time gives:

$$\left\{ \theta_\alpha, \frac{\partial \theta_\beta}{\partial \tau} \right\} \boldsymbol{q}_\beta = \{\theta_\alpha, \theta_\beta\} (\boldsymbol{S}_\beta - \boldsymbol{B}_\beta) - \left\{ \theta_\alpha, \frac{\partial \theta_\beta}{\partial \chi^{(k)}} \right\} \boldsymbol{F}_{k\beta} \tag{15}$$

2.1. The discontinuous Galerkin method

This method of computing the Galerkin predictor allows solutions to be discontinuous at temporal cell boundaries, and is also suitable for stiff source terms. Integrating (15) by parts in time gives:

$$\begin{aligned} &\left([\theta_\alpha, \theta_\beta]^1 - \left\{ \frac{\partial \theta_\alpha}{\partial \tau}, \theta_\beta \right\} \right) \boldsymbol{q}_\beta \\ &= [\theta_\alpha, \boldsymbol{w}]^0 + \{\theta_\alpha, \theta_\beta\} (\boldsymbol{S}_\beta - \boldsymbol{B}_\beta) - \left\{ \theta_\alpha, \frac{\partial \theta_\beta}{\partial \chi^{(k)}} \right\} \boldsymbol{F}_{k\beta} \end{aligned} \tag{16}$$

where $\boldsymbol{w}$ is the reconstruction obtained at the start of the time step with the WENO method. Take the following ordering:

$$\theta_{(N+1)^3 h + (N+1)^2 i + (N+1) j + k}(\tau, \chi, \upsilon, \zeta) = \phi_h(\tau) \psi_i(\chi) \psi_j(\upsilon) \psi_k(\zeta) \tag{17}$$

where $0 \le h, i, j, k \le N$. Thus, define the following:

$$U_{\alpha\beta} = \left[\theta_\alpha, \theta_\beta\right]^1 - \left\{\frac{\partial\theta_\alpha}{\partial\tau}, \theta_\beta\right\} = \left(R^1 - M^{\tau,1}\right) \otimes (M^\chi)^3 \tag{18a}$$

$$V^k_{\alpha\beta} = \left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\chi^{(k)}}\right\} = M^\tau \otimes (M^\chi)^{k-1} \otimes M^{\chi,1} \otimes (M^\chi)^{3-k} \tag{18b}$$

$$\boldsymbol{W}_\alpha = \left[\theta_\alpha, \Psi_\gamma\right]^0 \boldsymbol{w}_\gamma = R^0 \otimes (M^\chi)^3 \tag{18c}$$

$$Z_{\alpha\beta} = \left\{\theta_\alpha, \theta_\beta\right\} = M^\tau \otimes (M^\chi)^3 \tag{18d}$$

where $\left\{\Psi_\gamma\right\} = \Psi \otimes \Psi \otimes \Psi$ and:

$$\begin{cases} M^\tau_{ij} = \langle\phi_i, \phi_j\rangle & M^{\tau,1}_{ij} = \langle\phi_i', \phi_j\rangle \\ M^\chi_{ij} = \langle\psi_i, \psi_j\rangle & M^{\chi,1}_{ij} = \langle\psi_i, \psi_j'\rangle \\ R^1_{ij} = \phi_i(1)\phi_j(1) & \boldsymbol{R^0_i} = \phi_i(0) \end{cases} \tag{19}$$

Thus:

$$U_{\alpha\beta}\boldsymbol{q}_\beta = \boldsymbol{W}_\alpha + Z_{\alpha\beta}\left(\boldsymbol{S}_\beta - \boldsymbol{B}_\beta\right) - V^{(k)}_{\alpha\beta}\boldsymbol{F}_{k\beta} \tag{20}$$

Take the definitions:

$$\begin{cases} D = (M^\chi)^{-1}M^{\chi,1} \\ E = \left(R^1 - M^{\tau,1}\right) \end{cases} \tag{21}$$

Noting that $E\boldsymbol{1} = \boldsymbol{R^0}$, we have, by inversion of U:

$$\boldsymbol{q} = \left(\boldsymbol{1} \otimes I^3\right)\boldsymbol{w} + \left(E^{-1}M^\tau \otimes I^3\right)(\boldsymbol{S} - \boldsymbol{B}) - \left(E^{-1}M^\tau \otimes I^{k-1} \otimes D \otimes I^{3-k}\right)\boldsymbol{F}_k \tag{22}$$

Thus, we have:

$$\boldsymbol{q}_{hijk} = \boldsymbol{w}_{ijk} + \left(E^{-1}M^\tau\right)_{hm}\left(\boldsymbol{S}_{mijk} - \boldsymbol{B}_{mijk}\right) - \left(E^{-1}M^\tau\right)_{hm}\left(D_{in}(\boldsymbol{F}_1)_{mnjk} + D_{jn}(\boldsymbol{F}_2)_{mink} + D_{kn}(\boldsymbol{F}_3)_{mijn}\right) \tag{23}$$

Note then that $\boldsymbol{q}^{\iota\xi\kappa}$ is a function of $\boldsymbol{S}^{\iota\xi\kappa}, \boldsymbol{B}^{\iota\xi\kappa}, \overrightarrow{\boldsymbol{F}}$:

$$\boldsymbol{q}^{\iota\xi\kappa} = \mathcal{F}(\boldsymbol{S}^{\iota\xi\kappa}) + \mathcal{F}(\boldsymbol{B}^{\iota\xi\kappa}) + \mathcal{G}_{\iota\xi\kappa}(\overrightarrow{\boldsymbol{F}}^{000}, \ldots, \overrightarrow{\boldsymbol{F}}^{111}) \tag{24}$$

where $\mathcal{F}, \mathcal{G}_{\iota\xi\kappa}$ are linear functions. Note in turn that, by (11c):

$$\boldsymbol{S}^{\iota\xi\kappa} = \mathcal{H}\left(\bigcup_{(0,0,0) \le (a,b,c) \le (\iota,\xi,\kappa)} \boldsymbol{q}^{abc}\right) \tag{25}$$

where $\mathcal{H}$ is a nonlinear function.

In the case of stiff source terms, the following Picard iteration procedure can be used to solve (23), as adapted from Montecinos and Balsara [8]:

$$\left(\boldsymbol{q}^{\iota\xi\kappa}\right)_{m+1} = \mathcal{F}\left(\mathcal{H}\left(\left(\boldsymbol{q}^{\iota\xi\kappa}\right)_{m+1} \cup \bigcup_{\substack{(0,0,0) \le (a,b,c) \le (\iota,\xi,\kappa) \\ (a,b,c) \ne (\iota,\xi,\kappa)}} \left(\boldsymbol{q}^{abc}\right)_m\right)\right) + \mathcal{F}((\boldsymbol{B}^{\iota\xi\kappa})_m) + \mathcal{G}_{\iota\xi\kappa}((\overrightarrow{\boldsymbol{F}}^{000})_m, \ldots, (\overrightarrow{\boldsymbol{F}}^{111})_m) \tag{26}$$

2.2. The continuous Galerkin method

This method of computing the Galerkin predictor is not suitable for stiff source terms, but is less computationally expensive and ensures continuity across temporal cell boundaries. The first $N+1$ elements of $\boldsymbol{q}$ are fixed by imposing the following condition:

$$\boldsymbol{q}(\chi, 0) = \boldsymbol{w}(\chi) \tag{27}$$

For $\boldsymbol{v} \in \mathbb{R}^{(N+1)^2}$ and $X \in M_{(N+1)^2,(N+1)^2}(\mathbb{R})$, let $\boldsymbol{v} = (\boldsymbol{v^0}, \boldsymbol{v^1})$ and $X = \begin{pmatrix} X^{00} & X^{01} \\ X^{10} & X^{11} \end{pmatrix}$ where $\boldsymbol{v^0}$, X^{00} are the components relating solely to the first $N+1$ components of $\boldsymbol{v}$. We only need to find the latter components of $\boldsymbol{q}$, and thus, from (15), we have:

$$\left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\tau}\right\}^{11}\boldsymbol{q}^1_\beta = \left\{\theta_\alpha, \theta_\beta\right\}^{11}\left(\boldsymbol{S}^1_\beta - \boldsymbol{B}^1_\beta\right) - \left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\chi^{(k)}}\right\}^{11}\boldsymbol{F}^1_{k\beta} + \left\{\theta_\alpha, \theta_\beta\right\}^{10}\left(\boldsymbol{S}^0_\beta - \boldsymbol{B}^0_\beta\right) - \left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\chi^{(k)}}\right\}^{10}\boldsymbol{F}^0_{k\beta} \tag{28}$$

Define the following:

$$U_{\alpha\beta} = \left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\tau}\right\}^{11} \tag{29a}$$

$$V^k_{\alpha\beta} = \left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\chi^{(k)}}\right\}^{11} \tag{29b}$$

$$\boldsymbol{W}_\alpha = \left\{\theta_\alpha, \theta_\beta\right\}^{10}\left(\boldsymbol{S}_\beta - \boldsymbol{B}_\beta\right)^0 - \left\{\theta_\alpha, \frac{\partial\theta_\beta}{\partial\chi^{(k)}}\right\}^{10}\boldsymbol{F}^0_{k\beta} \tag{29c}$$

$$Z_{\alpha\beta} = \left\{\theta_\alpha, \theta_\beta\right\}^{11} \tag{29d}$$

Thus:

$$U_{\alpha\beta}\boldsymbol{q}^1_\beta = \boldsymbol{W}_\alpha + Z_{\alpha\beta}\left(\boldsymbol{S}^1_\beta - \boldsymbol{B}^1_\beta\right) - V^k_{\alpha\beta}\boldsymbol{F}^1_{k\beta} \tag{30}$$

Note that, as with the discontinuous Galerkin method, $\boldsymbol{W}$ has no dependence on the degrees of freedom in $\boldsymbol{q}$. As the source terms are not stiff, the following iteration is used:

$$U_{\alpha\beta}\left(\boldsymbol{q}^1_\beta\right)_{m+1} = \boldsymbol{W}_\alpha + Z_{\alpha\beta}\left(\left(\boldsymbol{S}^1_\beta\right)_m - \left(\boldsymbol{B}^1_\beta\right)_m\right) - V^k_{\alpha\beta}\left(\boldsymbol{F}^1_{k\beta}\right)_m \tag{31}$$

3. Convergence properties

In Jackson [7] it was proved that for traditional choices of polynomial bases, the eigenvalues of $U^{-1}V^i$ are all 0 for any $N \in \mathbb{N}$, for $i = 1, 2, 3$. This implies that in the conservative, homogeneous case ($\overrightarrow{\boldsymbol{B}} = \boldsymbol{S} = \boldsymbol{0}$), owing to the Banach Fixed Point Theorem, existence and uniqueness of a solution are established, and convergence to this solution is guaranteed. As noted in Dumbser and Zanotti [4], in the linear case it is implied that the iterative procedure converges after at most $N+1$ iterations. A proof of this result for the Montecinos–Balsara polynomial basis class is now provided here. For the theory in linear algebra required for this section, please consult a standard textbook on the subject, such as Nering [9].

Take the definitions (19), (21). Consider that:

$$U^{-1}V^k = E^{-1}M^\tau \otimes I^{k-1} \otimes D \otimes I^{3-k} \tag{32}$$

Therefore:

$$\left(U^{-1}V^k\right)^m = \left(E^{-1}M^\tau\right)^m \otimes \left(I^{k-1}\right)^m \otimes D^m \otimes \left(I^{3-k}\right)^m \tag{33}$$

A matrix X is nilpotent ($X^k = 0$ for some $k \in \mathbb{N}$) if and only if all its eigenvalues are 0. Note that $U^{-1}V^k$ is nilpotent if $D^m = 0$ for some $m \in \mathbb{N}$.

Note that if $p \in P_N$ then $p = a_j \psi_j$ for some unique coefficient vector $\boldsymbol{a}$. Thus, taking inner products with ψ_i, we have $\langle \psi_i, \psi_j \rangle a_j = \langle \psi_i, p \rangle$ for $i = 0, \ldots, N$. This produces the following result:

$$p = a_j \psi_j \Leftrightarrow \boldsymbol{a} = (M^{\chi})^{-1} \boldsymbol{x}, \; x_i = \langle \psi_i, p \rangle \tag{34}$$

Taking $\boldsymbol{a} \in \mathbb{R}^{N+1}$, define:

$$p = a_0 \psi_0 + \ldots + a_N \psi_N \in P_N \tag{35}$$

Note that:

$$\left(M^{\chi,1} \boldsymbol{a} \right)_i = \left\langle \psi_i, \psi_0' \right\rangle a_0 + \ldots + \left\langle \psi_i, \psi_N' \right\rangle a_N = \left\langle \psi_i, p' \right\rangle \tag{36}$$

Thus, by (34):

$$\left((M^{\chi})^{-1} M^{\chi,1} \boldsymbol{a} \right)_i \psi_i = (D\boldsymbol{a})_i \psi_i = p' \tag{37}$$

By induction:

$$(D^m \boldsymbol{a})_i \psi_i = p^{(m)} \tag{38}$$

for any $m \in \mathbb{N}$. As $p \in P_N$, $D^{N+1} \boldsymbol{a} = \boldsymbol{0}$. As $\boldsymbol{a}$ was chosen arbitrarily, $D^{N+1} = 0$. No specific choice has been made for $N \in \mathbb{N}$ and thus the result holds in general.

Thus, in the case that $\overrightarrow{\boldsymbol{B}} = \boldsymbol{S} = \boldsymbol{0}$, existence and uniqueness of a solution are established, and convergence to this solution is guaranteed for the iterative solution to (20) in the discontinuous Galerkin case, and (30) in the continuous Galerkin case.

Acknowledgements

I acknowledge financial support from the EPSRC Centre for Doctoral Training in Computational Methods for Materials Science under grant EP/L015552/1.

References

[1] Balsara DS, Rumpf T, Dumbser M, Munz CD. Efficient, high accuracy ADER-WENO schemes for hydrodynamics and divergence-free magnetohydrodynamics. J Comput Phys 2009;228:2480–516.
[2] Dumbser M, Balsara DS, Toro EF, Munz CD. A unified framework for the construction of one-step finite volume and discontinuous galerkin schemes on unstructured meshes. J Comput Phys 2008;227(18):8209–53.
[3] Dumbser M, Peshkov I, Romenski E, Zanotti O. High order ADER schemes for a unified first order hyperbolic formulation of continuum mechanics: viscous heat-conducting fluids and elastic solids. J Comput Phys 2016;314:824–62.
[4] Dumbser M, Zanotti O. Very high order pnpm schemes on unstructured meshes for the resistive relativistic mhd equations. J Comput Phys 2009;228:6991–7006.
[5] Dumbser M, Zanotti O, Hidalgo A, Balsara DS. Ader-weno finite volume schemes with space-time adaptive mesh refinement. J Comput Phys 2013;248:257–86.
[6] Hidalgo A, Dumbser M. ADER Schemes for nonlinear systems of stiff advection-diffusion-reaction equations. J Sci Comput 2011;48:173–89.
[7] Jackson H. On the eigenvalues of the ADER-WENO galerkin predictor. J Comput Phys 2017;333:409–13.
[8] Montecinos GI, Balsara DS. A cell-centered polynomial basis for efficient galerkin predictors in the context of ADER finite volume schemes. The one-dimensional case. Comput Fluids 2017;156:220–38.
[9] Nering ED. Linear algebra and matrix theory. John Wiley & Sons; 1970.
[10] Titarev VA, Toro EF. Ader: arbitrary high order godunov approach. J Sci Comput 2002;17(1):609–18.
[11] Toro E, Millington R, Nejad L. Towards very high order godunov schemes. In: Godunov methods. Springer; 2001. p. 907–40.
[12] Toro E, Titarev V. Solution of the generalized riemann problem for advection–reaction equations. In: Proceedings of the royal society of London A: Mathematical, Physical and Engineering Sciences, Vol. 458. The Royal Society; 2002. p. 271–81.
[13] Zanotti O, Dumbser M. Efficient conservative ADER schemes based on WENO reconstruction and space-time predictor in primitive variables. Comput Astrophys Cosmol 2016;3:1.