INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **47**:1197–1203
Published online 12 January 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.900

Implicit finite element discretizations based on the flux-corrected transport algorithm

Matthias Möller[*,†], D. Kuzmin and S. Turek

Institute of Applied Mathematics, University of Dortmund, Vogelpothsweg 87, D-44227 Dortmund, Germany

SUMMARY

The flux-corrected transport (FCT) methodology is generalized to implicit finite element schemes and applied to the Euler equations of gas dynamics. For scalar equations, a local extremum diminishing scheme is constructed by adding artificial diffusion so as to eliminate negative off-diagonal entries from the high-order transport operator. To obtain a nonoscillatory low-order method in the case of hyperbolic systems, the artificial viscosity tensor is designed so that all off-diagonal blocks of the discrete Jacobians are rendered positive semi-definite. Compensating antidiffusion is applied within a fixed-point defect correction loop so as to recover the high accuracy of the Galerkin discretization in regions of smooth solutions. All conservative matrix manipulations are performed edge-by-edge which leads to an efficient algorithm for the matrix assembly. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: high-resolution schemes; flux limiters; finite elements; compressible flows

1. INTRODUCTION

The concepts of FCT can be traced back to the celebrated SHASTA scheme proposed by Boris and Book [1] about three decades ago. Their algorithm was superseded by Zalesak's multi-dimensional limiter [2] and carried over to finite elements by Löhner *et al.* [3]. However, it remained entirely explicit and is thus subject to the CFL condition which becomes very restrictive if adaptive mesh refinement is employed.

In a series of recent publications [4, 5], we generalized the FCT paradigm to implicit finite element discretizations. The underlying nonoscillatory positivity-preserving scheme fulfils the local extremum diminishing (LED) criterion. Since all mass-conserving modifications rely solely on the magnitude, sign and position of nonzero matrix coefficients, this approach can be characterized as algebraic flux correction (AFC). To tap the potential of implicit discretizations

[*] Correspondence to: Matthias Möller, Institute of Applied Mathematics (LS III), University of Dortmund, Vogelpothsweg 87, D-44227 Dortmund, Germany.

[†] E-mail: matthias.moeller@math.uni-dortmund.de

Received 27 April 2004
Accepted 29 November 2004

to operate at arbitrarily large time steps, we present an iterative flux limiting strategy. The rejected antidiffusion is 'recycled' step-by-step so as to prevent the FEM–FCT scheme from becoming increasingly diffusive at large time steps. The proposed algorithm can be readily integrated into existing codes as a supplement to the matrix assembly.

2. DISCRETE UPWINDING

Consider the generic conservation law $\partial u/\partial t = -\nabla\cdot(\mathbf{v}u)$ discretized in space by the 'lumped-mass' Galerkin FEM. As a result, we obtain an ODE system of the form

$$M_{\mathrm{L}}\frac{\mathrm{d}u}{\mathrm{d}t} = Ku \quad \text{or} \quad m_i\frac{\mathrm{d}u_i}{\mathrm{d}t} = \sum_{j\neq i} k_{ij}(u_j - u_i) + \delta_i u_i, \quad \delta_i = \sum_j k_{ij} \tag{1}$$

where M_{L} denotes the lumped mass matrix and K stands for the discrete transport operator. If all coefficients k_{ij} were nonnegative for $i \neq j$, then the semi-discretized scheme would be local extremum diminishing in the absence of the term $\delta_i u_i$, which vanishes for divergence-free velocity fields and is responsible for a physical growth of local extrema otherwise. The operator K can be turned into its LED counterpart L by adding artificial diffusion D which is constructed so as to eliminate all negative off-diagonal entries [4]

$$L = K + D \quad \text{where} \quad d_{ii} = -\sum_{j\neq i} d_{ij}, \quad d_{ij} = \max\{0, -k_{ij}, -k_{ji}\} = d_{ji} \tag{2}$$

This modification proves to be conservative, since discrete diffusion operators possess zero row/column sums and thus can be decomposed into a sum of antisymmetric internodal fluxes

$$(Du)_i = \sum_{j\neq i} f_{ij}, \quad \text{where} \quad f_{ij} = d_{ij}(u_j - u_i) = -f_{ji} \tag{3}$$

In a practical implementation, we initialize $L := K$ and apply artificial diffusion edge-by-edge for each pair of nodes i and j whose basis functions have overlapping supports

$$\begin{aligned} l_{ii} &:= l_{ii} - d_{ij} \quad & l_{ij} &:= l_{ij} + d_{ij} \\ l_{ji} &:= l_{ji} + d_{ij} \quad & l_{jj} &:= l_{jj} - d_{ij} \end{aligned} \tag{4}$$

This 'discrete upwinding' technique carries over to multi-dimensions and yields the least diffusive *linear* LED scheme. Since linear monotonicity-preserving methods can be at most first-order accurate, the excessive artificial diffusion is to be removed by applying *nonlinear* antidiffusion so as to recover the discretization of high order in regions of smooth solutions.

3. ALGEBRAIC FLUX CORRECTION OF FCT TYPE

After an implicit time discretization ($0 < \theta \leqslant 1$), we obtain the nonlinear algebraic system

$$M_{\mathrm{L}}\frac{u^{n+1} - u^n}{\Delta t} = \theta L u^{n+1} + (1-\theta)Lu^n + f(u^{n+1}, u^n) \tag{5}$$

Successive approximations to the end-of-step solution u^{n+1} can be computed, e.g. by the fixed-point defect correction scheme. Its two-step implementation is as follows

$$A\Delta u^{(m+1)} = r^{(m)} \quad m = 0, 1, 2, \ldots \tag{6}$$

$$u^{(m+1)} = u^{(m)} + \Delta u^{(m+1)} \quad u^{(0)} = u^n \tag{7}$$

where $r^{(m)} = b^{(m+1)} - Au^{(m)}$ denotes the residual for the mth cycle and A is a 'preconditioner' which should be easy to invert. In general, a system of the form $Au^{n+1} = Bu^n$ preserves the positivity of the old solution u^n if B has no negative entries and A is a so-called M-matrix. By construction, the evolution operator of the underlying linear LED scheme $A = M_{\mathrm{L}} - \theta\Delta tL$ enjoys this property and thus provides an excellent preconditioner for system (6).

3.1. Basic FEM–FCT algorithm

The load vector $b^{(m+1)} = b^n + f(u^{(m)}, u^n)$, where $b^n = [M_{\mathrm{L}} + (1-\theta)\Delta tL]u^n$ is the right-hand side of the low-order scheme, would satisfy the above positivity constraint if the antidiffusive correction $f(u^{(m)}, u^n)$ was omitted. To retain as much antidiffusion as possible without generating new extrema or accentuating existing ones, the raw antidiffusive fluxes are to be multiplied by appropriate correction factors. As a result, the right-hand side for the equation at hand is to be redefined as follows [5]:

$$b_i^{(m+1)} = b_i^n + \sum_{j \neq i} \alpha_{ij}^{(m)} f_{ij}^{(m)}, \quad 0 \leqslant \alpha_{ij}^{(m)} \leqslant 1 \tag{8}$$

The scaling factors $\alpha_{ij}^{(m)}$ are estimated by Zalesak's limiter [2] making use of an intermediate solution $\tilde{u}^n = M_{\mathrm{L}}^{-1} b^n$ which is supposed to be positivity-preserving. For a detailed description of the limiting process and the positivity proof, the interested reader is referred to Reference [5].

3.2. Iterative FEM–FCT algorithm

Unfortunately, implicit FCT schemes of this form suffer from a loss of accuracy at large time steps. As a remedy, an iterative limiting strategy was introduced in Reference [5], which differs from the algorithm presented above in that the previously accepted antidiffusion is taken into account and only the rejected portion of the antidiffusive flux needs to be dealt with at subsequent defect correction steps. To this end, the provisional solution $\tilde{u}^{(m)} = M_{\mathrm{L}}^{-1} b^{(m)}$ is updated at each outer iteration and the scaling factors are applied to the difference between the raw antidiffusion and all previous corrections $\Delta f_{ij}^{(m)} = f_{ij}^{(m)} - g_{ij}^{(m)}$. Finally, the limited antidiffusive fluxes are added to the sum of their predecessors and inserted into the right-hand side

$$g_{ij}^{(m+1)} = g_{ij}^{(m)} + \alpha_{ij}^{(m)} \Delta f_{ij}^{(m)} \quad \text{and} \quad b_i^{(m+1)} = b_i^{(m)} + \sum_{j \neq i} \alpha_{ij}^{(m)} \Delta f_{ij}^{(m)} \tag{9}$$

As the iteration process continues, more and more antidiffusion can be built into the intermediate solution. At the same time, the task of the flux limiter simplifies, because the remainder $\Delta f_{ij}^{(m)}$ shrinks and a larger percentage of it can be accepted.

4. EULER EQUATIONS

Compressible flows are governed by the Euler equations which represent a system of conservation laws for the mass, momentum and energy of an inviscid fluid. Let U denote the vector of conservative variables and $\mathbf{F} = (F^1, F^2, F^3)$ stand for the triple of fluxes for each co-ordinate direction. Then the hyperbolic PDE system can be written in divergence form as

$$\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{F} = 0 \quad \text{where} \quad U = \begin{bmatrix} \rho \\ \rho \mathbf{v} \\ \rho E \end{bmatrix} \quad \text{and} \quad \mathbf{F} = \begin{bmatrix} \rho \mathbf{v} \\ \rho \mathbf{v} \otimes \mathbf{v} + p\mathcal{I} \\ \rho H \mathbf{v} \end{bmatrix} \tag{10}$$

By application of the chain rule, the Euler equations can be expressed in terms of the Jacobian matrices $\mathbf{A} = (A^1, A^2, A^3)$, which gives an equivalent quasi-linear formulation

$$\frac{\partial U}{\partial t} + \mathbf{A} \cdot \nabla U = 0 \quad \text{where} \quad \mathbf{A} \cdot \nabla U = \sum_{d=1}^{3} A^d \frac{\partial U}{\partial x_d}, \quad A^d = \frac{\partial F^d}{\partial U} \tag{11}$$

4.1. Discretization and matrix assembly

As has been already mentioned in the Introduction, *implicit* finite element methods are still rarely used in compressible flow simulations. Therefore, matrix assembly for the Euler equations has received little attention in the literature. In what follows, we show how it can be implemented building on Roe's linearization technique for hyperbolic systems [6].

Discretizing the divergence form (11) in space yields an ODE system as follows:

$$M_C \frac{\mathrm{d}U}{\mathrm{d}t} = KU \quad \text{or} \quad \sum_j M_{ij} \frac{\mathrm{d}U_j}{\mathrm{d}t} = -\sum_{j \neq i} \mathbf{c}_{ij} \cdot (\mathbf{F}_j - \mathbf{F}_i) \tag{12}$$

where M_C is the block-diagonal mass matrix and the coefficients $\mathbf{c}_{ij} = \int_\Omega \varphi_i \nabla \varphi_j \, \mathrm{d}\mathbf{x}$ possess zero column sums. To obtain K, which is a discrete counterpart of the operator $-\mathbf{A} \cdot \nabla$ for Equation (11), we need to eliminate the dependent variables $\mathbf{F}_j$ in favour of the unknowns U_j.

In his pioneering work on approximate Riemann solvers [6], Roe showed that

$$\mathbf{F}_j - \mathbf{F}_i = \hat{\mathbf{A}}_{ij}(U_j - U_i) \tag{13}$$

where $\hat{\mathbf{A}}_{ij}$ is the Jacobian tensor $\mathbf{A}$ evaluated at a special set of density-averaged variables.

As a result, the right-hand side of Equation (12) can be assembled edge-by-edge considering the differences between the values of the conservative variables/fluxes at nodes i and j

$$(KU)_i \longleftarrow \mathbf{c}_{ij} \cdot (\mathbf{F}_i - \mathbf{F}_j) = \mathbf{c}_{ij} \cdot \hat{\mathbf{A}}_{ij}(U_i - U_j) \tag{14}$$

$$(KU)_j \longleftarrow \mathbf{c}_{ji} \cdot (\mathbf{F}_j - \mathbf{F}_i) = \mathbf{c}_{ji} \cdot \hat{\mathbf{A}}_{ij}(U_j - U_i) \tag{15}$$

Let us introduce the so-called cumulative Roe matrices which are defined as follows [5]:

$$\begin{aligned} A_{ij} &= \mathbf{a}_{ij} \cdot \hat{\mathbf{A}}_{ij} \\ B_{ij} &= \mathbf{b}_{ij} \cdot \hat{\mathbf{A}}_{ij} \end{aligned} \quad \text{where} \quad \begin{aligned} \mathbf{a}_{ij} &= \frac{\mathbf{c}_{ij} - \mathbf{c}_{ji}}{2} \\ \mathbf{b}_{ij} &= \frac{\mathbf{c}_{ij} + \mathbf{c}_{ji}}{2} \end{aligned} \tag{16}$$

The symmetric part B_{ij} equals zero unless both nodes are located on the boundary so that only the antisymmetric part A_{ij} of the cumulative Roe matrix needs to be evaluated for interior edges. The contribution of the edge ij to (14)–(15) reads

$$(KU)_i \longleftarrow (A_{ij} + B_{ij})(U_i - U_j) \tag{17}$$

$$(KU)_j \longleftarrow (A_{ij} - B_{ij})(U_i - U_j) \tag{18}$$

This representation leads to a very efficient edge-based algorithm for assembling the right-hand side of Equation (12), since the coefficients $\mathbf{c}_{ij}$ remain constant as long as the mesh is fixed and thus can be generated once and for all at the beginning of the simulation. The sparsity graph of the global matrix K depends solely on the underlying mesh and on the type of approximation. As in the scalar case, it contains nonzero entries only for those pairs of nodes i and j whose basis functions have overlapping supports.

If the global element matrix itself (or parts of it) rather than its product with the solution vector U needs to be assembled, this can be accomplished by evaluating the four 5×5 blocks

$$K_{ii} = A_{ij} + B_{ij} \quad K_{ij} = -A_{ij} - B_{ij} \tag{19}$$

$$K_{ji} = A_{ij} - B_{ij} \quad K_{jj} = -A_{ij} + B_{ij} \tag{20}$$

edge-by-edge and scattering their entries to the positions with indices i and j in the corresponding blocks of the operator K.

4.2. Design of artificial viscosities

To a large extent, the ability of a high-resolution scheme to withstand the formation of wiggles depends on the quality of the underlying low-order method. Following the strategy for scalar equations, we perform mass lumping and replace the original Galerkin discretization (12) by

$$M_{\mathrm{L}} \frac{\mathrm{d}U}{\mathrm{d}t} = LU \quad \text{or} \quad m_i \frac{\mathrm{d}U_i}{\mathrm{d}t} = \sum_{j \neq i} L_{ij}(U_j - U_i) \tag{21}$$

where L is the low-order Jacobian operator. Recall that its scalar counterpart was derived by conservative elimination of negative off-diagonal entries from the high-order operator. As a generalization of the LED principle to hyperbolic systems, all off-diagonal matrix blocks L_{ij} need to be positive semi-definite (that is, their eigenvalues should be nonnegative) [5].

To construct a nonoscillatory low-order scheme for the compressible Euler equations, we add tensorial artificial viscosity D_{ij} and remove the symmetric part of the cumulative Roe matrix

$$L_{ii} = A_{ij} - D_{ij} \quad L_{ij} = -A_{ij} + D_{ij} \tag{22}$$

$$L_{ji} = A_{ij} + D_{ij} \quad L_{jj} = -A_{ij} - D_{ij} \tag{23}$$

The global matrix assembly can be adopted from the previous section. The missing boundary terms B_{ij} belong to the antisymmetric raw antidiffusive fluxes which read

$$F_{ij} = -\left(M_{ij} \frac{\mathrm{d}}{\mathrm{d}t} + D_{ij} + B_{ij}\right)(U_j - U_i) = -F_{ji} \tag{24}$$

It remains to design the matrix D_{ij} so as to comply with the generalized LED constraint.

Due to the hyperbolicity of the Euler equations, there exists a diagonal matrix Λ_{ij} of real eigenvalues and a regular matrix R_{ij} of right eigenvectors such that

$$A_{ij} = |\mathbf{a}_{ij}| R_{ij} \Lambda_{ij} R_{ij}^{-1} \quad \text{where} \quad |\mathbf{a}_{ij}| = \sqrt{\mathbf{a}_{ij} \cdot \mathbf{a}_{ij}} \tag{25}$$

As a generalization of Roe's approximate Riemann solver [6], all negative eigenvalues of the coefficient blocks L_{ij} and L_{ji} can be eliminated by setting the artificial viscosity D_{ij} as follows:

$$D_{ij} = |A_{ij}| = |\mathbf{a}_{ij}| R_{ij} |\Lambda_{ij}| R_{ij}^{-1} \tag{26}$$

In the FEM–FCT framework, it is worthwhile to add scalar dissipation $D_{ij} = d_{ij}\mathrm{I}$ which is proportional to the spectral radius of the Roe matrix $d_{ij} = |\mathbf{a}_{ij}| \max_i |\lambda_i|$. This choice of D_{ij} is much more efficient and excessive diffusion is removed by the flux limiter anyway. Flux correction is performed as in the scalar case using a transformation to the local characteristic variables or a synchronization of the correction factors for the conservative ones [3, 5].

5. NUMERICAL EXAMPLES

To illustrate the performance of the algorithm, consider two test problems for convection-dominated flows proposed by LeVeque [7]. The results depicted in Figure 1 were computed on a uniform mesh of 128×128 bilinear elements using the Crank–Nicolson time-stepping scheme.

In Figure 2, the generalized FEM–FCT scheme has been applied to the classical shock tube problem of gas dynamics (left) and to a steady supersonic channel flow ($M_\infty = 2.5$) past a wedge of 15° (right). As expected, the results produced by the low-order method (top) are unacceptably smeared due to excessive diffusion. In contrast, the oblique shock is resolved

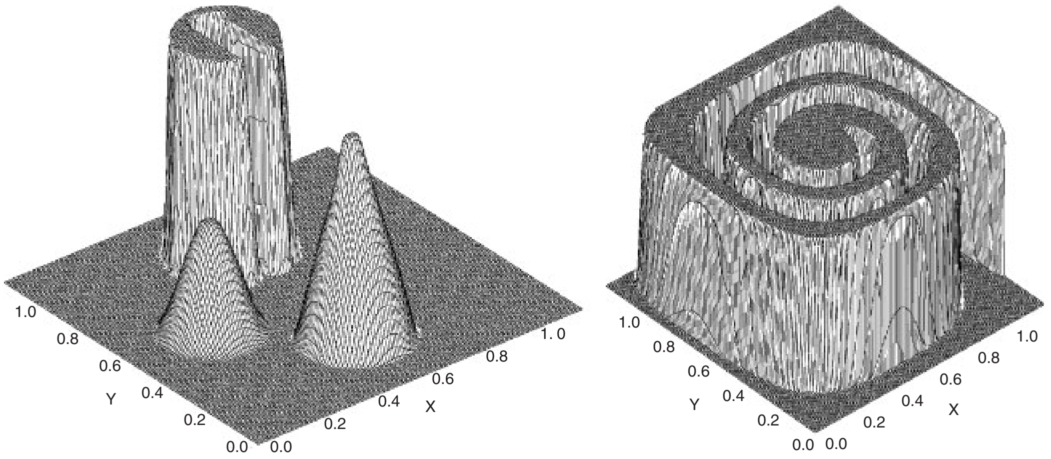

Figure 1. Solid body rotation (left) and swirling flow (right).

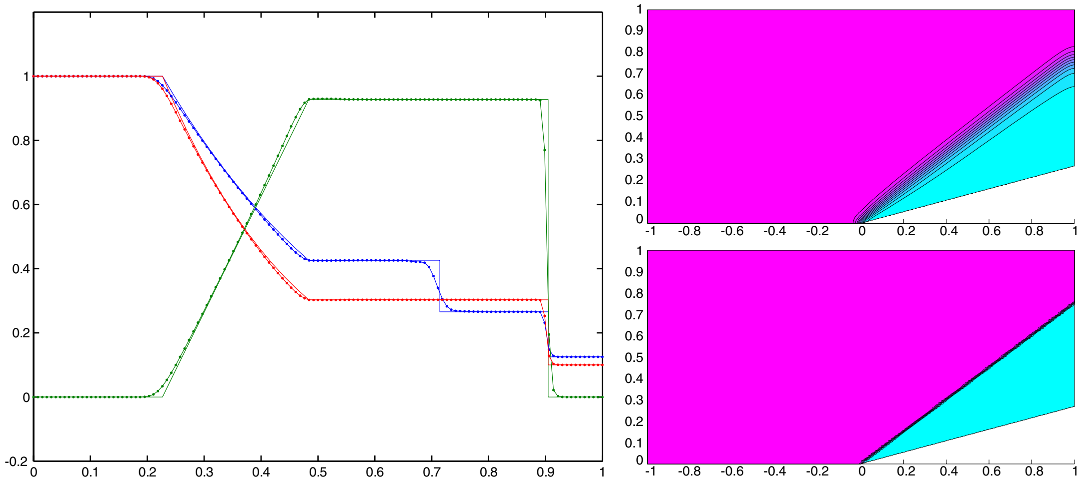

Figure 2. Shock tube problem (left) and compression corner (right).

accurately by the new FEM–FCT scheme (bottom). A detailed description of the algorithm for the Euler equations and additional examples can be found in Reference [5].

REFERENCES

1. Boris JP, Book DL. Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works. *Journal of Computational Physics* 1973; **11**:38–69.
2. Zalesak ST. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics* 1979; **31**:335–362.
3. Löhner R, Morgan K, Peraire J, Vahdati M. Finite element flux-corrected transport (FEM–FCT) for the Euler and Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1987; **7**:1093–1109.
4. Kuzmin D, Turek S. Flux correction tools for finite elements. *Journal of Computational Physics* 2002; **175**: 525–558.
5. Kuzmin D, Möller M. Algebraic flux correction. Part I–II. In *Flux-Corrected Transport*: *Principles*, *Algorithms*, *and Applications*, Kuzmin D, Löhner R, Turek S (eds). Springer: Berlin, 2005.
6. Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
7. LeVeque RJ. High-resolution conservative algorithms for advection in incompressible flow. *SIAM Journal on Numerical Analysis* 1996; **33**:627–665.