INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2001; **35**: 903–938

Positivity preserving pointwise implicit schemes with application to turbulent compressible flat plate flow

Lyon W. J. Lanerolle*,[1]

School for Marine Science and Technology, University of Massachusetts-Dartmouth, New Bedford, MA, U.S.A.

SUMMARY

A family of positivity preserving pointwise implicit schemes applicable to source term dominated problems is constructed, where the minimum order of spatial accuracy is one and the maximum is three. It is designed for achieving steady state numerical solutions and is constructed through the analysis of appropriate model problems, where the convective fluxes for the higher-order members are prescribed by the Chakravarthy–Osher family of total variation diminishing (TVD) schemes. Multidimensionality is facilitated by operator splitting. Numerical experimentation confirms the stability, convergence, accuracy, positivity, and computational efficiency associated with the proposed schemes. These schemes are ideally suited to solving the low-Reynolds number turbulent k–ϵ equations for which the positivity of k and ϵ and the presence of stiff source terms are critical issues. Hence, using a finite volume formulation of these schemes, the low-Reynolds number Chien k–ϵ turbulence model is implemented for a flat plate geometry and a series of turbulent flow (steady state) computations are carried out to demonstrate the positivity, robustness, and reliability of the algorithm. The free-stream and initial k and ϵ values are specified in a very simple manner. Algorithm convergence acceleration is achieved using Multigrid techniques. The k–ϵ model flow predictions are shown to be in agreement with empirical profiles. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: flat plate; k–ϵ model; point implicit; positivity preserving; pseudo-time stepping; source terms

1. INTRODUCTION

The numerical approximation of convection terms arising in partial differential equations (PDEs) is still an unresolved issue in the field of computational fluid dynamics (CFD). The discretization of these terms should be so as to yield stable, accurate, and ideally monotonicity preserving numerical schemes. The simplest example of an accurate discretization for

* Correspondence to: School for Marine Science and Technology, University of Massachusetts-Dartmouth, 706 South Rodney French Boulevard, New Bedford, MA 02744, U.S.A. Tel.: +1 508 9106310; fax: +1 508 9998197.

[1] E-mail: llanerolle@umassd.edu

Received 30 November 1999
Revised 13 June 2000

convection is to use central differencing, but it will lead to the numerical solutions containing grid-based oscillations in the high gradient regions and in the vicinity of extrema. For strongly coupled systems of equations and those that are source term dominated, these oscillations can show growth in an unbounded fashion leading to blow-up. Godunov [1] showed that all monotone linear schemes can, at best, be only of first-order spatial accuracy. For central differenced schemes, numerical stability and the damping of oscillations can be achieved with the use of appropriate artificial dissipation models. However, the level of dissipation that needs to be included in a numerical algorithm (and its variation locally) to guarantee oscillation-free solutions is not known in advance and excessive amounts will degrade the solution quality. On the other hand, the use of a spatially first-order accurate monotonicity preserving scheme will include excessive numerical dissipation in the solutions generated—particularly in situations such as high-Reynolds number turbulent fluid flows involving geometries where the flow is not in alignment with the mesh. One key requirement in turbulence modeling is to be able to distinguish the differences in the solutions of alternative turbulence models. Grid refinement can be used to reduce numerical errors but this technique is not always practical in situations such as in flows over complex geometries and in three-dimensional flows (due to computational storage requirements and expense). Hence, an attractive option is to consider higher-order non-linear convection algorithms, which contain ‘built-in’ numerical dissipation and which also do not generate unphysical oscillations.

Harten [2] set up a mathematical framework for the construction of non-linear higher-order monotonicity preserving upwind schemes based on the total variation diminishing (TVD) concept. TVD schemes are based on the principle that a discrete convective flux consists of a first-order accurate monotone flux an a set of correction terms with the effects of the latter limited (usually referred to as flux-limited) in such a manner so as to render the discretization monotonicity preserving; when these correction terms are not limited, the resulting convective flux is spatially high order (for example, second- or third-order) but non-monotone. In the smooth parts of numerical solutions, the correction terms are not limited, which results in the underlying higher-order schemes being used. However, in regions with steep gradients or exterma, these terms are limited so as to suppress the creation of any grid-based oscillations and hence in such regions, TVD schemes are of lower spatial accuracy than in smooth solution regions.

The process by which the above-stated correction terms are limited is not unique and hence has led to the development of many TVD schemes. An in-depth survey of various TVD schemes is given in References [3,4].

The above discussion applies only in one space dimension to linear and non-linear scalar equations and also to systems of linear equations. Genuinely multidimensional TVD schemes are quite sophisticated and not easy to construct—in particular, invoking of the flux-limiting procedure in multidimensions and maintaining high accuracy; some examples of such schemes are given in References [5–7]. Therefore, what is commonly done in *steady state* calculations is to apply the one-dimensional TVD schemes to each of the co-ordinate directions separately (operator splitting), which is also the approach adopted in the present work. There is, however, no mathematical theory to guarantee that this approach leads to oscillation-free solutions, although it has been observed to be effective in inviscid flow computations [8].

TVD schemes have been primarily constructed for homogeneous hyperbolic PDEs and although the inclusion of source terms in practice can be straightforward, there is no formal theory regarding the monotone properties of such schemes. Moreover, the effectiveness of TVD schemes for source term dominated problems is often not adequately described in the literature and therefore one of the primary objectives in the present paper is to analyze this. Source terms must also be dealt with effectively—particularly in stiff problems without which instability and slow convergence can result; in this paper, pointwise implicit (or point implicit) methods are used. Point implicit methods solve a given system of PDEs at each grid point in an implicit fashion by treating the source terms implicitly and the unknown variables at other points explicitly. The resulting pointwise systems, when they are non-linear, can be solved with the Newton–Raphson method (if good enough starting guesses are available). These schemes have the advantage of being quicker than solving the global system of equations but nevertheless take full account of the force of the source terms. Steady state numerical solutions are achieved by time marching the unsteady solutions.

The k–ϵ system of equations used to model turbulent flow is an ideal example where monotonicity preserving numerical schemes are required for the discretization of convection, because k and ϵ are essentially positive-valued variables but grid-based oscillations can drive them negative. However, in the literature, the use of TVD schemes for the solution of the k–ϵ equations is not very common; some instances of their use are given in References [9–13] and only the first three of these references consider compressible flows. Further, in all of these references, with the exception of Reference [9], the near-wall turbulence is modeled using either wall functions or a separate low-Reynolds number model. Low-Reynolds number k–ϵ models, which are valid throughout the whole of the flow field, are hence preferred, but in the vicinity of solid boundaries the source terms exhibit stiffness [14]; although this can be a severe restriction, pointwise implicit methods are well suited to handling such source terms. In the present research it is the Chien low-Reynolds number k–ϵ model that is employed [15–17]. With regard to accuracy, Reference [9] employs a second-order accurate TVD scheme and Reference [11] a third-order accurate one. In Reference [10] it is the Chakravarthy–Osher family of TVD schemes (also given in References [4,8,18,19]) that is used and it will be employed in the present research too in order to perform turbulent flow computations because: (i) a family of high-order accurate TVD schemes can be generated by merely varying a single parameter and (ii) it also includes a third-order accurate TVD scheme, which would be the ideal candidate to use. In general, advection schemes that are accurate to odd orders are dissipative and dissipative truncation errors help towards damping-out oscillations. Therefore, a spatially third-order accurate positivity preserving (i.e., k and ϵ remain ≥ 0) pointwise implicit method would be an ideal scheme to use in the numerical solution of the Chien low-Reynolds number k–ϵ equations.

An outline of this paper is as follows. First, the alternative first-order accurate point implicit methods are developed and the conditions for stability and positivity preservation are established. These schemes are then extended to higher-order spatial accuracy using the Chakravarthy–Osher family of TVD schemes and numerical experimentation is performed for a source term dominated one-dimensional scalar and a two-dimensional scalar hyperbolic model problem. The extension of this analysis to a two-dimensional equation pair is presented thereafter. Finally, the numerical algorithm in a finite volume formulation is employed to compute fully and partly (steady state) turbulent compressible flows over a flat plate with the Chien low-Reynolds number k–ϵ model.

2. POINT IMPLICIT METHODOLOGY

Point implicit methods solve a given system of PDEs at each grid point in an implicit fashion by treating the source terms implicitly and the unknown variables at other points (from the discretization of convection and diffusion terms) explicitly. The resulting pointwise equations when they are non-linear can be solved with the Newton–Raphson method (if good enough starting guesses are available). As it is *steady state* solutions that are of interest, they are achieved by pseudo-time stepping the unsteady equations with the time derivatives discretized in a forward Euler manner: as shown in Reference [14] it is advantageous to pseudo-time step to steady state than to solve the steady equations directly. There are two iteration processes:

(a) an outer iteration, which involves the updating of the nodewise variables in time over the whole mesh and
(b) an inner iteration at each grid point to solve the pointwise equations.

As it is positive-valued solutions that will be sought (because the ultimate aim is to solve the k–ϵ equations and k and ϵ are essentially positive quantities), computable restrictions for the pseudo-time steps are determined in order to ensure:

1. the existence of a positive-valued solution for the outer iterations at each grid point (ideally a single positive solution but otherwise multiple positive solutions);
2. the numerical stability and positivity preservation of the algorithm for the outer iterations at each grid point; and
3. that the solution of the local point equations (even when using the Newton method for non-linear equations) generates positive-valued solutions.

3. ONE-DIMENSIONAL POINT IMPLICIT SCHEMES

Spatially first-order accurate positivity preserving, point implicit schemes are described below. In Section 3.2 they are extended to higher spatial accuracy.

3.1. First-order accurate point implicit schemes

The convection terms will be discretized using first-order accurate upwind differences and the diffusion terms will be approximated by standard central differences. Then, three different basic point implicit schemes can be constructed by discretizing the convection–diffusion terms $aU_x - bU_{xx}$ (for the case $a, b > 0$) so that on a grid of uniform spacing Δx and at a node j and pseudo-time level $n+1$

$$aU_x - bU_{xx}\big|_j = \frac{a(U_j^n - U_{j-1}^n)}{\Delta x} - \frac{b(U_{j+1}^n - 2U_j^n + U_{j-1}^n)}{\Delta x^2} \tag{1}$$

or

$$aU_x - bU_{xx}\big|_j = \frac{a(U_j^{n+1} - U_{j-1}^n)}{\Delta x} - \frac{b(U_{j+1}^n - 2U_j^{n+1} + U_{j-1}^n)}{\Delta x^2} \tag{2}$$

or

$$aU_x - bU_{xx}\big|_j = \frac{a(U_j^{n+1} - U_{j-1}^{n+1})}{\Delta x} - \frac{b(U_{j+1}^n - 2U_j^{n+1} + U_{j-1}^{n+1})}{\Delta x^2} \tag{3}$$

which will be called explicit, semi-implicit and Gauss–Seidel convection–diffusion respectively. For the model problem

$$[U_t] + aU_x - bU_{xx} = S(U), \quad a, b > 0 \tag{4}$$

with a linear source term $S(U) = sU$ such that $s \lesseqgtr 0$, the resulting numerical schemes with $v = a\cdot\Delta t_j/\Delta x$, $\mu = b\cdot\Delta t_j/\Delta x^2$ (Δt_j is the pseudo-time step at node j) will satisfy modified maximum principles (modified to include source term contributions—see Reference [14]) for explicit convection–diffusion when (*i*) $v + 2\mu \leq 1$ and if $s > 0$ then also (ii) $1 - \Delta t_j s > 0$ and for the semi-implicit and Gauss–Seidel discretizations when $1 + v + 2\mu - \Delta t_j s > 0$. These relations enable the determination of upper bounds for time steps that will ensure the satisfaction of the modified maximum principles.

Although the time step restrictions are the same, the Gauss–Seidel discretization (which exploits the latest information arising by sweeping the grid) is expected to show faster convergence to the steady state (see, for example, Reference [20]) than the semi-implicit scheme but it is only effective when the wind and grid sweeping directions are coincident, which is a severe restriction.

3.1.1. One-dimensional scalar model problem. The estimation of the pseudo-time step restrictions to satisfy conditions (1)–(3) of Section 2 will be demonstrated via the model problem

$$[U_t] + aU_x - bU_{xx} = cU(1 - U); \quad a, b, c \text{ (global constants)} > 0 \tag{5}$$

which, like the k–ϵ equations (assuming uniqueness and existence), has two constant steady states: a trivial root $U = 0$ and also another positive root $U = 1.0$. This is an autonomous equation [21–23] and it can be shown that the former is an unstable root and the latter is an asymptotically stable one. Because the analytic solution to Equation (5), in the absence of convection–diffusion with $U(t = 0) = U_0 > 0$, is

$$U(t) = \frac{U_0\,\mathrm{e}^{ct}}{(1 - U_0) + U_0\,\mathrm{e}^{ct}} > 0 \quad \forall t \in \Re^+ \tag{6}$$

the aim is to maintain the positiveness of U with a positivity preserving numerical discretization for the convection–diffusion terms—because only oscillations due to the discretization of convection can cause negative values in $U(x, t)$.

The time step restrictions are derived in Appendix A and the analysis shows that the Gauss–Seidel discretization is expected to provide the fastest convergence to the steady state, but it is very restrictive for real problems as explained above. The semi-implicit method through numerical experimentation has been shown [14] to exhibit convergence no better than the explicit method. It is also not elementary to construct spatially higher-order accurate schemes based on the semi-implicit and Gauss–Seidel schemes, and hence hereafter only the explicit discretization will be considered.

3.2. Higher-order accurate point implicit schemes

The first-order accurate convective flux will now be replaced by a higher-order monotone flux from the Chakravarthy–Osher family of TVD schemes. This family of schemes and its implementation have been adequately described in References [8,10,14,18,19,24] and it suffices to state here that different members of the family are generated by varying a single parameter ϕ. The resulting unlimited schemes and their spatial accuracy for a single space dimension are summarized in Table I.

As the aim of the present research is to use the most accurate discretization for the convection terms, it will be attempted to employ the spatially third-order accurate scheme, which results when $\phi = 1/3$. The following objectives would like to be achieved:

- to find the most effective means of prescribing the values of variables in the fictitious grid points (due to the extended convection stencil)—which are beyond the boundary points;
- to study the convergence and the global accuracy of the limited and unlimited TVD schemes; and
- to analyze whether the source terms adversely affect the performance of the TVD schemes.

It is vital that the unlimited TVD scheme (with $\phi = 1/3$) is implemented in such a manner that it is as close as possible to being third-order accurate; otherwise the accuracies of the limited schemes will also be affected.

Table I. The Chakravarthy–Osher family of TVD schemes (TE, truncation error; $\nu_{\max}$, maximum permitted CFL number).

ϕ	Name of scheme	TE	$\nu_{\max}$
1/3	Third-order	$\mathcal{O}(\Delta x^3)$	2/5
-1	Fully upwind	$+\frac{1}{3}\frac{\partial^3 f}{\partial x^3}\cdot(\Delta x)^2$	2/3
0	Fromm's	$+\frac{1}{12}\frac{\partial^3 f}{\partial x^3}\cdot(\Delta x)^2$	1/2
1/2	Low TE second-order	$-\frac{1}{24}\frac{\partial^3 f}{\partial x^3}\cdot(\Delta x)^2$	1/3
$-1/3$	No name	$+\frac{1}{6}\frac{\partial^3 f}{\partial x^3}\cdot(\Delta x)^2$	4/7

The discretization of the time derivatives and the diffusion terms remain unchanged (from before). It is clear that the time step analysis performed for the first-order accurate convective flux extends naturally to the present case, with only the restriction for the positivity preservation of the algorithm requiring the modification where $\nu + 2\mu < \nu_{\max}$ (Table I).

3.3. *Numerical experiments*

To validate the above analysis, the following model problem is studied:

$$[U_t] + aU_x = c(x)U(1 - U); \quad a \text{ (constant)} > 0 \tag{7}$$

for $x \in \Omega = [0, 1]$ with $c(x) = 1 - 4\pi \cos(4\pi x)$ and the Dirichlet boundary condition $U(0, t) = 0.5$; the initial conditions are $U(x, 0) = 0.5\ \forall x \in \Omega$. The function $c(x)$ was included in order to ensure that the analytic steady state solution to this equation

$$U(x) = \frac{e^{\hat{c}(x)/a}}{1 + e^{\hat{c}(x)/a}} \tag{8}$$

where $\hat{c}(x) = \int c(x)\, dx = x - \sin(4\pi x)$, was of a non-monotone character.

Details regarding the numerical solution of this equation will now be discussed. The fictitious grid around the computational domain is defined by linearly extrapolating the x values from within the domain, which ensures the smooth distribution of the grid points. The variables at the fictitious nodes are assigned their values by extrapolating from within the domain by using one of the following as exemplified for a node $N + 1$ (where $x > 1$):

$$\text{zeroth-order:} \quad U_{N+1} = U_N \tag{9}$$

$$\text{first-order:} \quad U_{N+1} = 2U_N - U_{N-1} \tag{10}$$

$$\text{second-order:} \quad U_{N+1} = 3U_N - 3U_{N-1} + U_{N-2} \tag{11}$$

$$\text{third-order:} \quad U_{N+1} = 4U_N - 6U_{N-1} + 4U_{N-2} - U_{N-3} \tag{12}$$

As the numerical scheme (with $\phi = 1/3$) should ideally be third-order accurate, it is not necessary to use extrapolations as high as third order, which is only included for comparison purposes. Generally, for a numerical method (applied to a mixed initial boundary value problem) that is spatially accurate to order p, the use of extrapolation to the order of $p - 1$ is sufficient to maintain the order of accuracy [25].

The results of numerical experiments performed using the above model problem on uniform grids with $a = 1.0$ are given in Figures 1–3 and Table II. For the Newton method, the starting guesses used to compute variables at time level $n + 1$ were the values of the variables at time level n (previous time values).

The convergence history (on a mesh having 101 grid points) for the unlimited TVD scheme with $\phi = 1/3$ shows that the numerical algorithm is convergent for all of the different

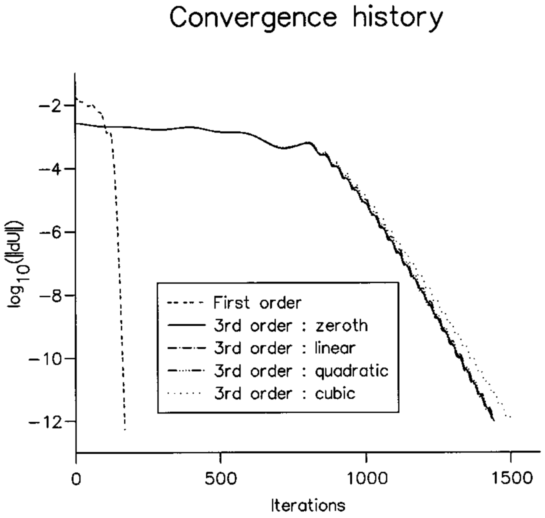

Figure 1. One-dimensional hyperbolic model problem: convergence history for various extrapolations. ($\|dU\| := \|U^{n+1} - U^n\|_2$, n is the pseudo-time level).

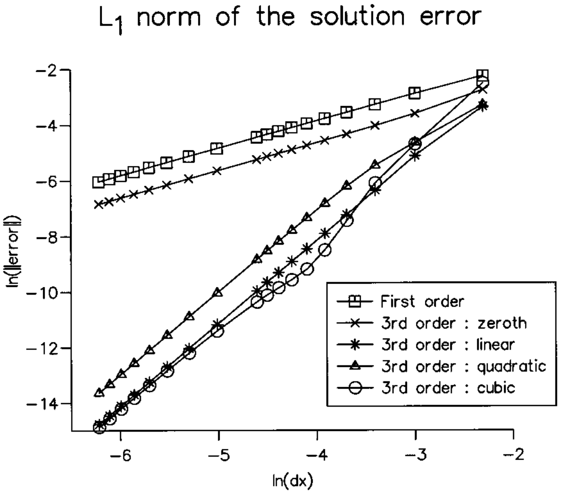

Figure 2. One-dimensional hyperbolic model problem: plot of L_1 norms of the global error for various extrapolations.

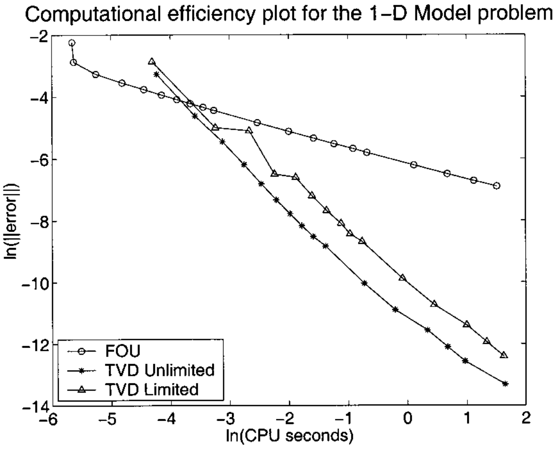

Figure 3. One-dimensional hyperbolic model problem: computational efficiency plot showing the cost of error reduction for the first-order and third-order (unlimited and limited with quadratic extrapolation) methods.

Table II. Global accuracies of the unlimited TVD scheme with $\phi = 1/3$ for the one-dimensional model problem with different extrapolations.

Extrapolation	Global accuracy
None (FOU)	$(\Delta x)^{0.98}$
Zeroth	$(\Delta x)^{0.99}$
First	$(\Delta x)^{2.95}$
Second	$(\Delta x)^{2.99}$

extrapolations and they have similar convergence profiles and rates: this scheme requires more iterations than the first-order upwind (FOU) scheme due to the stricter Courant–Friedrich–Lewy (CFL) restrictions. The global accuracies associated with each of the extrapolations (computed using a least-squares fit on the straight line regions of the L_1 error plots) are summarized in Table II and it is clear that, for the unlimited schemes, both linear and quadratic extrapolations yield a nearly third-order accurate scheme, with the latter producing a better result. The L_1 error plots also serve to show that although on the coarsest grid the errors from different methods and extrapolations are of similar value, the rate of fall of error with grid spacing is greatest for the higher-order methods. Furthermore, the computational efficiency plot in Figure 3 indicates that upon grid refinement, the third-order method (with second-order extrapolation) with and without flux limiting is significantly more efficient at reducing errors than the first-order method, thus making it a more attractive method to use. As expected, the limited higher-order scheme exhibits errors of larger magnitude than the unlimited version.

Table III, in which the results of the limited and unlimited computations with the Chakravarthy–Osher family of TVD schemes are summarized, indicates that in all cases the limited and unlimited algorithms are convergent and their convergence rates (for each value of ϕ) are similar. The limited schemes are of higher global accuracy than the unlimited ones and such a behavior is also documented in Reference [18]. For $\phi = 1/3$ the limited and unlimited schemes (with quardratic extrapolation) are nearly third-order accurate, while for other values of ϕ the schemes (with linear extrapolation) are post-second-order accurate. Numerical experiments also revealed that the time step restrictions for the inner (Newton) iteration were not required to be enforced; therefore they are not employed hereafter.

Hence, it is possible to conclude that the present implementation of the Chakravarthy–Osher family of TVD schemes for the solution of the model problem (7) has yielded a convergent algorithm for all admissible values of ϕ, even with the presence of a non-linear source term. For $\phi = 1/3$ with quadratic extrapolation, both the limited and unlimited schemes are nearly spatially third-order accurate. The numerical algorithm converges to the desired solution (positive valued, attractive solution), thereby indicating the positivity property of the algorithm.

4. TWO-DIMENSIONAL POINT IMPLICIT SCHEMES

The first-order accurate explicit discretization of the convection terms on a uniform mesh with mesh spacing Δx in the x-direction and Δy in the y-direction at node (i, j) are now given by the following three point stencils:

1. $a_1 > 0$ and $a_2 > 0$: $$[a_1 U_x + a_2 U_y]^n_{i,j} = a_1 \frac{U^n_{i,j} - U^n_{i-1,j}}{\Delta x} + a_2 \frac{U^n_{i,j} - U^n_{i,j-1}}{\Delta y} \tag{13}$$

2. $a_1 < 0$ and $a_2 > 0$: $$[a_1 U_x + a_2 U_y]^n_{i,j} = a_1 \frac{U^n_{i+1,j} - U^n_{i,j}}{\Delta x} + a_2 \frac{U^n_{i,j} - U^n_{i,j-1}}{\Delta y} \tag{14}$$

3. $a_1 < 0$ and $a_2 < 0$: $$[a_1 U_x + a_2 U_y]^n_{i,j} = a_1 \frac{U^n_{i+1,j} - U^n_{i,j}}{\Delta x} + a_2 \frac{U^n_{i,j+1} - U^n_{i,j}}{\Delta y} \tag{15}$$

4. $a_1 > 0$ and $a_2 < 0$: $$[a_1 U_x + a_2 U_y]^n_{i,j} = a_1 \frac{U^n_{i,j} - U^n_{i-1,j}}{\Delta x} + a_2 \frac{U^n_{i,j+1} - U^n_{i,j}}{\Delta y} \tag{16}$$

For the higher-order monotone discretization of the convection terms, the operator splitting approach is adopted, where the one-dimensional formulations are applied in each of the spatial directions separately. The diffusion terms are also constructed by applying the one-dimensional discretizations in each of the spatial directions separately.

When moving from one to two dimensions, the character of the pointwise discrete equations remains unchanged and hence the previously described pseudo-time step analysis is also valid here with the exception that the restriction for the positivity preservation of the explicit algorithm is now given by $v_x + v_y + 2\mu_x + 2\mu_y < v_{\max}$, where

Table III. Summary of results for the one-dimensional scalar hyperbolic model problem (number of cells = number of nodes − 1).

Scheme		Iterations			L_1 error			Global accuracy
ϕ	State	60 cells	80 cells	100 cells	60 cells	80 cells	100 cells	
FOU	—	112	140	169	1.94×10^{-2}	1.47×10^{-2}	1.18×10^{-2}	$(\Delta x)^{0.98}$
1/3	Unlimited	755	886	1026	6.52×10^{-4}	2.82×10^{-4}	1.46×10^{-4}	$(\Delta x)^{2.93}$
1/3	TVD limited	713	854	990	7.47×10^{-4}	3.05×10^{-4}	1.70×10^{-4}	$(\Delta x)^{2.91}$
−1	Unlimited	432	539	644	1.88×10^{-3}	1.05×10^{-3}	6.67×10^{-4}	$(\Delta x)^{2.03}$
−1	TVD limited	456	559	684	1.98×10^{-3}	9.86×10^{-4}	6.37×10^{-4}	$(\Delta x)^{2.03}$
0	Unlimited	510	618	725	5.41×10^{-4}	2.88×10^{-4}	1.79×10^{-4}	$(\Delta x)^{2.17}$
0	TVD limited	507	601	704	6.28×10^{-4}	3.22×10^{-4}	2.03×10^{-4}	$(\Delta x)^{2.22}$
1/2	Unlimited	1129	1273	1427	2.59×10^{-4}	1.38×10^{-4}	3.60×10^{-5}	$(\Delta x)^{2.16}$
1/2	TVD limited	993	1184	1343	3.54×10^{-4}	1.64×10^{-4}	1.19×10^{-4}	$(\Delta x)^{2.16}$
−1/3	Unlimited	415	516	616	9.86×10^{-4}	5.41×10^{-4}	3.41×10^{-4}	$(\Delta x)^{2.08}$
−1/3	TVD limited	417	520	638	1.03×10^{-3}	5.49×10^{-4}	3.49×10^{-4}	$(\Delta x)^{2.12}$

$$\nu_x := a_1 \frac{\Delta t}{\Delta x}, \quad \nu_y := a_2 \frac{\Delta t}{\Delta y}, \quad \mu_x := b_1 \frac{\Delta t}{\Delta x^2}, \quad \mu_y := b_2 \frac{\Delta t}{\Delta y^2} \tag{17}$$

and b_1, b_2 are diffusion coefficients in the x- and y-directions respectively, and $\nu_{\max}$ is as in Table I.

4.1. Numerical experiments

Consider the model problem

$$[U_t] + a_1 U_x + a_2 U_y = c(x, y)U(1 - U), \quad a_1, a_2 \text{ (constant)} > 0 \tag{18}$$

in $(x, y) \in \Omega' = [0, 1] \times [0, 1]$ with $c(x, y) = 1 - 4\pi \cos(4\pi x) - 4\pi \cos(4\pi y)$ and the boundary conditions

$$U(0, y, t) = \frac{e^{F(y)/a_2}}{1 + e^{F(y)/a_2}} \quad \text{for } 0 \leq y \leq 1 \tag{19}$$

$$U(x, 0, t) = \frac{e^{F(x)/a_1}}{1 + e^{F(x)/a_1}} \quad \text{for } 0 \leq x \leq 1 \tag{20}$$

where $F(z) = 0.5z - \sin(4\pi z)$. The initial conditions are taken to be $U(x, y, 0) = U(0, 0, t) = 0.5$. This problem has the analytic steady state solution

$$U(x, y) = \frac{e^{F(x)/a_1 + F(y)/a_2}}{1 + e^{F(x)/a_1 + F(y)/a_2}} \tag{21}$$

which clearly does not exhibit monotone behavior.

Numerical experiments with this model problem were conducted for $a_1 = 1.0$ and $a_2 = 1/\sqrt{2}$ and the results are presented in Figures 4–6 and Table IV. Here too, the Newton method was employed as in the one-dimensional model problem.

The convergence histories (on a 101 × 101 grid) show that the unlimited TVD algorithm with $\phi = 1/3$ is convergent for all of the different extrapolations used and that they have similar convergence rates. The global accuracies (computed as for the one-dimensional model problem) for the different extrapolations presented in Table IV show that in order to maintain an accuracy close to third order, linear extrapolation (unlike for the 1-D model problem) is not sufficient and at least quadratic extrapolation is required. Third-order extrapolation too is seen to give the desired results producing smaller errors than the other extrapolations (perhaps due to a smaller error constant) and such an observation has also been reported by Hayase *et al.* [26] for a comparison between second- and third-order accurate boundary conditions used in a QUICK scheme. The L_1 error plot shows that even on the coarsest grid, the error from the higher-order methods is significantly smaller than the first-order method and that the higher-order methods (upon grid refinement) enable the errors to fall at a sharper rate. The computational efficiency plot in Figure 6 indicates that the third-order method (unlimited and

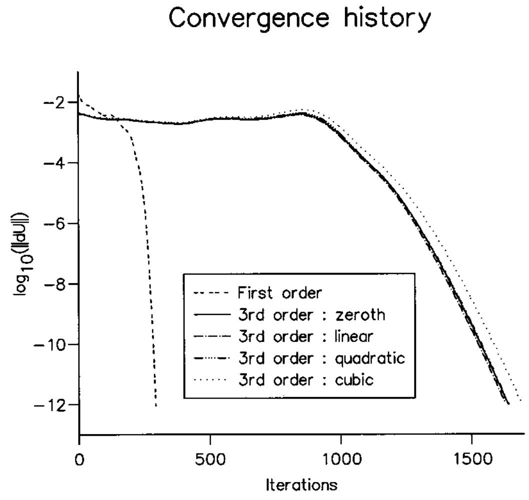

Figure 4. Two-dimensional hyperbolic model problem: convergence history for various extrapolations. ($\|dU\| := \|U^{n+1} - U^n\|_2$, n is the pseudo-time level).

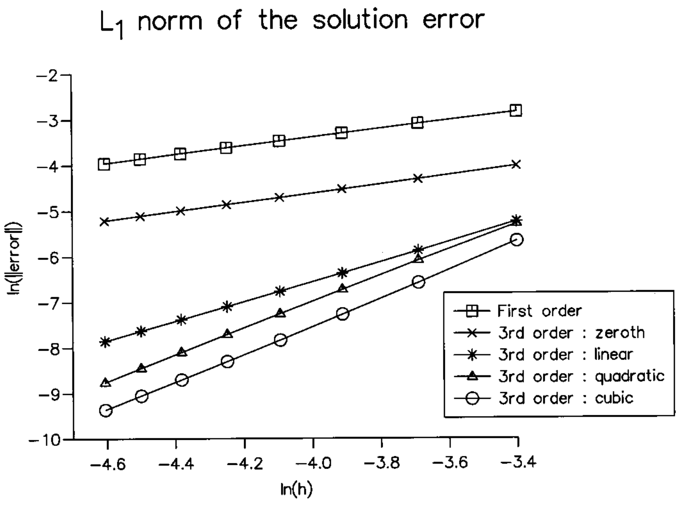

Figure 5. Two-dimensional hyperbolic model problem: plot of L_1 norms of the global error for various extrapolations.

limited with quadratic extrapolation) is far superior to the first-order method in error reduction and extremely fine grids (with massive storage requirements) would be required by the latter if it were to compete with the former.

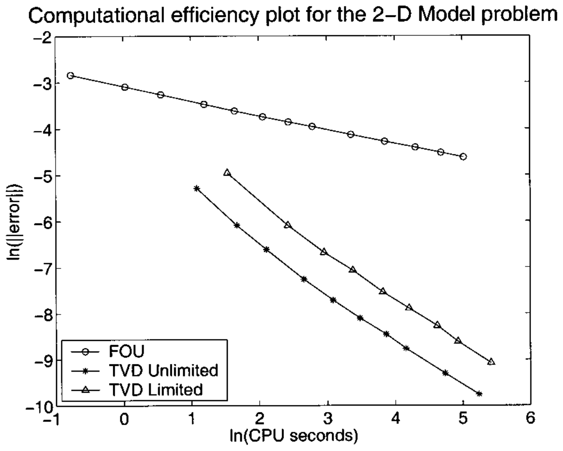

Figure 6. Two-dimensional hyperbolic model problem: computational efficiency plot showing the cost of error reduction for the first-order and third-order (unlimited and limited with quadratic extrapolation) methods.

Table IV. Global accuracies of the unlimited TVD scheme with $\phi = 1/3$ for the two-dimensional model problem with different extrapolations.

Extrapolation	Global accuracy
None (FOU)	$(\Delta x)^{0.93}$
Zeroth	$(\Delta x)^{0.99}$
First	$(\Delta x)^{2.17}$
Second	$(\Delta x)^{2.89}$
Third	$(\Delta x)^{3.07}$

Table V summarizes the results of limited and unlimited computations carried out with the two-dimensional model problem. When specifying the values of the variables in the fictitious grid points, for the scheme with $\phi = 1/3$, quadratic extrapolation was employed and for the others, linear extrapolation was used. The following conclusions can be made: in all cases the algorithms are convergent, the limited and unlimited schemes for each value of ϕ have similar orders of spatial accuracy, the scheme with $\phi = 1/3$ is nearly third-order accurate (with a slight reduction with respect to the one-dimensional observations), and for other values of ϕ the schemes are in the region of second-order accuracy.

Therefore, the conclusions reached for the one-dimensional test case also extend to the two-dimensional situation and the use of the operator splitting approach is effective in constructing two-dimensional TVD schemes for steady state computations involving source terms.

Table V. Summary of results for the two-dimensional scalar hyperbolic model problem (number of cells = number of nodes − 1).

Scheme		Iterations			L_1 error			Global accuracy
ϕ	State	60^2 cells	80^2 cells	100^2 cells	60^2 cells	80^2 cells	100^2 cells	
FOU	—	270	344	415	3.11×10^{-2}	2.36×10^{-2}	1.91×10^{-2}	$(\Delta x)^{0.95}$
1/3	Unlimited	1090	1378	1642	7.04×10^{-4}	3.03×10^{-4}	1.57×10^{-4}	$(\Delta x)^{2.94}$
1/3	TVD limited	1034	1310	1574	8.55×10^{-4}	3.74×10^{-4}	2.03×10^{-4}	$(\Delta x)^{2.82}$
−1	Unlimited	723	919	1112	2.91×10^{-3}	1.68×10^{-3}	1.09×10^{-3}	$(\Delta x)^{1.93}$
−1	TVD limited	751	954	1136	2.84×10^{-3}	1.67×10^{-3}	1.07×10^{-3}	$(\Delta \mathrm{x})^{1.91}$
0	Unlimited	805	1007	1208	1.21×10^{-3}	6.51×10^{-4}	4.04×10^{-4}	$(\Delta x)^{2.15}$
0	TVD limited	768	987	1196	1.37×10^{-3}	7.49×10^{-4}	4.45×10^{-4}	$(\Delta x)^{2.19}$
1/2	Unlimited	1718	1916	2230	1.25×10^{-3}	6.83×10^{-4}	4.30×10^{-4}	$(\Delta x)^{2.09}$
1/2	TVD limited	1559	1780	2091	1.29×10^{-3}	7.37×10^{-4}	4.45×10^{-4}	$(\Delta x)^{2.09}$
−1/3	Unlimited	689	878	1060	1.68×10^{-3}	9.30×10^{-4}	5.90×10^{-4}	$(\Delta x)^{2.05}$
−1/3	TVD limited	683	898	1084	1.80×10^{-3}	1.01×10^{-3}	6.18×10^{-4}	$(\Delta x)^{2.08}$

5. EXTENSION TO SYSTEMS OF EQUATIONS

It will now be demonstrated how the analysis for the scalar equations can also be extended to a system of two equations. The problem under consideration is

$$[U_t] + a_1 U_x + a_2 U_y - b_1 U_{xx} - b_2 U_{yy} = U^2 - 4U + V \equiv S_u \tag{22}$$

$$[V_t] + a_1 V_x + a_2 V_y - b_1 V_{xx} - b_2 V_{yy} = U + V - \frac{4}{9} V^2 \equiv S_v \tag{23}$$

with $a_1 \lessgtr 0$, $a_2 \lessgtr 0$, $b_1, b_2 > 0$. This particular problem was constructed, because as in the assumption for the k–ϵ equations (uniqueness and existence of solutions), it is required to have at least one pair of asymptotically stable roots (U^*, V^*) such that U^*, $V^* > 0$, while the other roots (including the trivial one) are unstable. For the above model equations, in the absence of the convection–diffusion terms and in steady state, $U = 1$, $V = 3$ is the stable root and $(U, V) = (0, 0)$, $(9/2, -5/2)$, $(5/2, 15/4)$ are the unstable ones. Figure 7 presents a plot containing the curves $S_u = 0$, $S_v = 0$ and also arrows indicating the signs and magnitudes of vector components (S_u, S_v) in the U–V plane and it clearly indicates the asymptotic stability of the four roots. In the neighborhood of the stable root ($U = 1$, $V = 3$), the arrows point towards it from all directions, but in the vicinity of the others there are arrows pointing away from them, thereby implying that they are unstable.

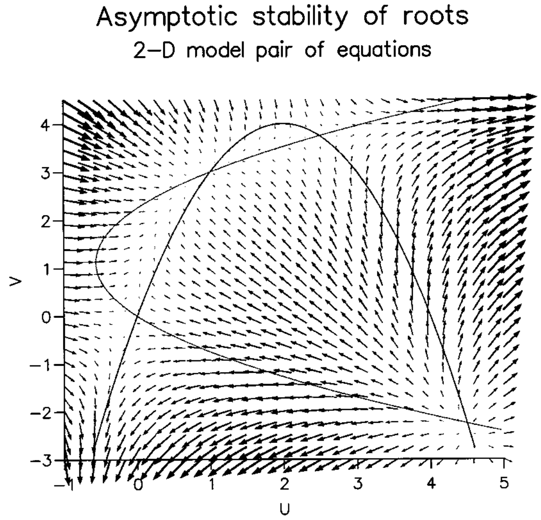

Figure 7. Source term vectors for the two-dimensional model pair of equations showing the asymptotic stability of the roots.

In the above pair of model equations, the convection and diffusion coefficients were taken to be the same because in the two-dimensional k–ϵ equations, the convection coefficients for the k and ϵ equations are identical and the diffusion coefficients have similar values.

Under the explicit discretization for the convection–diffusion terms, the resulting point equations at a node (i, j) and pseudo-time level $n+1$ on a mesh with uniform mesh spacings Δx and Δy (in the two co-ordinate directions) for the special case $a_1, a_2 > 0$ are

$$\Delta t_{i,j} w^2 - (1 + 4\Delta t_{i,j})w + \Delta t_{i,j} z + \{U^n_{i,j} - D^n_{i,j}\Delta t_{i,j}\} \equiv \mathscr{F}_1(w, z) = 0 \tag{24}$$

$$\frac{4}{9}\Delta t_{i,j} z^2 + (1 - \Delta t_{i,j})z - \Delta t_{i,j} w - \{V^n_{i,j} - E^n_{i,j}\Delta t_{i,j}\} \equiv \mathscr{F}_2(w, z) = 0 \tag{25}$$

where $w = U^{n+1}_{i,j}$, $z = V^{n+1}_{i,j}$ and $D^n_{i,j}$, $E^n_{i,j}$ are the discrete convection–diffusion terms for the U and V equations respectively.

The roots (w, z) are formed when $\mathscr{F}_1 = \mathscr{F}_2 = 0$, which represents the intersection of two bi-quadratics, one in w and the other in z. It is not trivial to determine the conditions on $\Delta t_{i,j}$ to achieve this (such that $w, z > 0$). Moreover, even if this were possible, as multiple positive (w, z) roots can occur, it is not straightforward to decide which particular root to converge to. Therefore, the conditions on $\Delta t_{i,j}$ to produce positive-valued w and z roots are determined by analyzing a simplified system of decoupled quadratic equations, where, in the quadratic equation for w, z is fixed at the previous time level and vice versa so that

$$\Delta t_{i,j} w^2 - (1 + 4\Delta t_{i,j})w + \{U^n_{i,j} + (V^n_{i,j} - D^n_{i,j})\Delta t_{i,j}\} = 0 \tag{26}$$

$$\frac{4}{9}\Delta t_{i,j} z^2 + (1 - \Delta t_{i,j})z - \{V^n_{i,j} - (E^n_{i,j} - U^n_{i,j})\Delta t_{i,j}\} = 0 \tag{27}$$

5.1. Time step restrictions

The aim is to find the restrictions on the time step that will ensure the existence of a single, real and positive-valued (w, z) root. If this is not possible, it will be attempted to create multiple positive-valued, real (w, z) roots. Each of the above decoupled quadratic equation (26) and (27) has the same form as the point one-dimensional scalar model equations, and hence time step restrictions can be determined by extending that analysis, as shown in Reference [14]. When these time step restrictions are used, however, for the simplified two-dimensional model pair of point equations (above) two real positive w roots and a single real positive z root will occur. For the true two-dimensional model pair of point equations (24) and (25), it is also expected that these time step restrictions will enable the existence of at least a single real positive attractive (w, z) root and that the Newton method along with a sufficiently accurate initial guess will be able to converge to this root.

The numerical stability and positivity preservation conditions can only be evaluated once the two-dimensional model pair of equations has been suitably linearized. Consider linearizing the model system about the asymptotically stable root, $U = 1.0$, $V = 3.0$ (with $u = U + 1.0$, $v = V + 3.0$) so that

$$[u_t] + a_1u_x + a_2u_y - b_1u_{xx} - b_2u_{yy} = -2u + v \tag{28}$$

$$[v_t] + a_1v_x + a_2v_y - b_1v_{xx} - b_2v_{yy} = u - \frac{5}{3}v \tag{29}$$

It is not obvious how to extend the one-dimensional scalar modified maximum principle analysis (Section 3.1) to a coupled linear system of equations, but a technique that is applicable under special circumstances is presented in Reference [14]. That method involves writing the discrete equivalents of Equations (28) and (29) together in matrix form and finding the conditions to get positive-valued coefficients for the variables U^n, U^{n+1}, V^n, V^{n+1} and then forcing $C^{n+1} \leq C^n$, where $C^n = \max_{\forall i,j}\{U^n_{i,j}, V^n_{i,j}\}$. Using the earlier notation it then turns out that a modified principle is established if $v_x + v_y + 2\mu_x + 2\mu_y \leq v_{\max}$.

The overall pointwise time step limits used in the algorithm (as in the proposition of Appendix A) are determined by taking into account all of the restrictions that apply to both (U and V) point equations (simultaneously).

With regards to the numerical solution of the non-linear point equations (24) and (25), the starting guesses for the Newton method are be taken to be $(U^n_{i,j}, V^n_{i,j})$, which are the previous time values of (U, V) and the exiting criterion is

$$\frac{1}{\Delta t_{i,j}} \max(|\mathscr{F}_1|, |\mathscr{F}_2|) < \text{Tol} \tag{30}$$

where 'Tol' is a sufficiently small prescribed tolerance.

Numerical examples to verify the reliability of this analysis are given in Reference [14] but a proper study of accuracy is not possible due to the lack of availability of analytic solutions (unlike the previously described model problems). A similar model problem but with stiff source terms (with a stiffness ratio of 999) has also been studied in Reference [14]. Furthermore, as the true test of the validity of this analysis will be the solution of the Chien $k-\epsilon$ system of equations, that will be the test considered (next) in this paper.

6. FINITE VOLUME FORMULATION OF THE POINT IMPLICIT SCHEMES

Like most numerical schemes, TVD schemes have been primarily constructed for use on uniform grids. The extension of these schemes to non-uniform grids is not trivial, particularly with respect to maintaining the same accuracy as that on uniform grids. The finite volume procedure given here is based on that of References [8,10,19] and it is constructed by first transforming the differential equations from a physical domain to a computational domain consisting of rectangular cells and then discretizing the resulting metrics (with central differences [27]) in computational space. Further details regarding the transformations can be found in References [27,28] and the derivation of the finite volume scheme and the resulting time step restrictions (which are direct extensions from the finite difference formulation) are given in Reference [14].

In order to test this finite volume algorithm, the two-dimensional hyperbolic model problem was solved on a series of stretched rectangular meshes. The meshes were generated using geometric progressions with one starting at the lower boundary and extending in the increasing y-direction, while the other was started at the left-hand boundary and extended in the increasing x-direction. The same geometric ratio, r, and the number of grid points were maintained in the two spatial directions; the computational domain was always $[0, 1] \times [0, 1]$. The results on a series of meshes each containing 61×61 nodes are included in Table VI. It is evident that in all cases the algorithm converges to the given tolerance (1.0×10^{-12}) and the errors on the stretched meshes do not differ significantly from that on the uniform mesh ($r = 1.0$). The limited and unlimited forms of the TVD scheme (with $\phi = 1/3$) are seen to produce errors that are appreciably less in magnitude than those for the FOU scheme.

Therefore, it can be concluded that the finite volume formulation of the point implicit scheme encompassing the Chakravarthy–Osher TVD scheme is capable of generating accurate numerical solutions (which are in keeping with higher-order spatial discretizations) on both uniform and stretched rectangular meshes with non-trivial levels of stretching.

7. TURBULENT FLAT PLATE COMPUTATIONS

The flat plate geometry is ideally suited to test turbulent flow computations because a well-known set of empirical solutions are available for it (for example, see Reference [14]). Furthermore, the k–ϵ model is a good system of equations to test positivity preserving numerical methods because the positiveness of k and ϵ can easily be checked and numerical algorithms usually show blow-up if these quantities become negative.

The mean-flow effects are modeled using the mass-averaged compressible Navier–Stokes equations [14,16,17,29–36] and the turbulent effects are included by the use of the Chien low-Reynolds number k–ϵ model [15–17] and by relying on the eddy–viscosity hypothesis [37]; the turbulence model is presented in Appendix B. The mean-flow equations are solved using an explicit cell vertex finite volume scheme as demonstrated in Reference [14]. In this method the variables are stored at cell vertices (nodes). First, the node-based variables are integrated along cell edges (as a line integral) to recover their gradients at cell centroids, which are then interpolated to nodes to recover node-based gradients: this process is spatially second-order accurate. Then, the inviscid and viscous fluxes are assembled at nodes and they

Table VI. L_1 errors for the finite volume TVD scheme on stretched meshes (r, geometric ratio).

r	FOU		TVD unlimited ($\phi = 1/3$)		TVD limited ($\phi = 1/3$)	
	L_1 error	Iterations	L_1 error	Iterations	L_1 error	Iterations
1.000	3.11×10^{-2}	274	7.04×10^{-4}	1090	8.55×10^{-4}	1034
1.025	2.96×10^{-2}	266	5.13×10^{-4}	1096	8.16×10^{-4}	1736
1.050	2.61×10^{-2}	244	1.02×10^{-3}	1068	1.89×10^{-3}	941
1.075	2.19×10^{-2}	219	1.60×10^{-3}	1058	2.08×10^{-3}	924
1.100	1.81×10^{-2}	197	1.96×10^{-3}	1166	3.09×10^{-3}	877

are integrated in the same manner to recover cell-based (cell-averaged) residuals. Next, these residuals are mapped (distributed) to the surrounding nodes via (Lax–Wendroff) distribution matrices after the inclusion of artificial dissipation contributions (to enhance numerical stability, control grid-based oscillations and enable shock capturing). Thereafter, the node-based residuals are adjusted for boundary conditions (derivative boundary conditions are implemented during the assembling of the fluxes) and employed to update the node-based variables in time. Once the norm (an area weighted L_2 norm) of the node-based residuals falls below a certain prescribed tolerance (usually 1.0×10^{-6}), the numerical solutions are assumed to have reached the steady state.

The turbulent flow equations that are decoupled from the mean-flow equations are solved (for a known Navier–Stokes flow field) by directly extending the numerical methods presented in this paper. The time step restrictions for the numerical stability and positivity preservation are determined by the use of a simplified Chien k–ϵ system, where the source terms have been linearized. The restrictions on the time steps in order to obtain positive k and ϵ roots are determined by using another simplified point Chien k–ϵ equation system, where in the point k equation, ϵ is fixed (at the previous time level) and in the point ϵ equation, k is fixed (at the previous time level); moreover, in the latter the damping function $E(k, \epsilon)$ is treated as known (by evaluating it with k and ϵ values at the previous time level). Further details on these issues can be found in Reference [14].

Multigrid is employed to provide convergence acceleration and it is only applied to the mean-flow equations. The Multigrid algorithm used in the present work is that of Reference [38]. In this algorithm, the smoothing relaxations are performed using the generalized Lax–Wendroff method and the restriction operator is volume weighted; the prolongation operator uses volume weighted interpolation. This scheme is based on the full approximation scheme (FAS) of Brandt [39]. Furthermore, the algorithm also uses W cycles as opposed to V cycles as it is seen to increase its robustness; the robustness has been further improved by employing a full multigrid (FMG) start-up, where the initial guess for successively finer grids is taken as the prolonged solution from the coarse grid. Typical grid sequences are four or five levels.

The geometry and the mesh containing 129×81 points are shown in Figures 8 and 9 respectively. In designing the grid it has been ensured that the dimensionless wall co-ordinate y^+ is less than unity, with also 65 grid points along the plate and 40 within the boundary layer. In the present work, due to the use of a low-Reynolds number k–ϵ model where the equations are integrated up to solid boundaries, a stretched grid within the boundary layer has to be used in order to get sufficiently close to them, thereby making grid refinement studies somewhat meaningless. Hence, grid refinement studies have not been undertaken in the present work, although it has been demonstrated clearly for the model problems that the higher-order schemes generate solutions that are more accurate than those from the first order accurate scheme.

The turbulence model is applied only from the leading edge onwards; the area before it is filled with a globally fixed level of free-stream k and ϵ values (predicted by a prescribed value of free-stream turbulence intensity, Tu [14]). When employing the Chakravarthy–Osher third-order TVD scheme for the solution of the k–ϵ equations, however, due to the extended stencil for convection, the k and ϵ values are linearly interpolated across the leading edge [14].

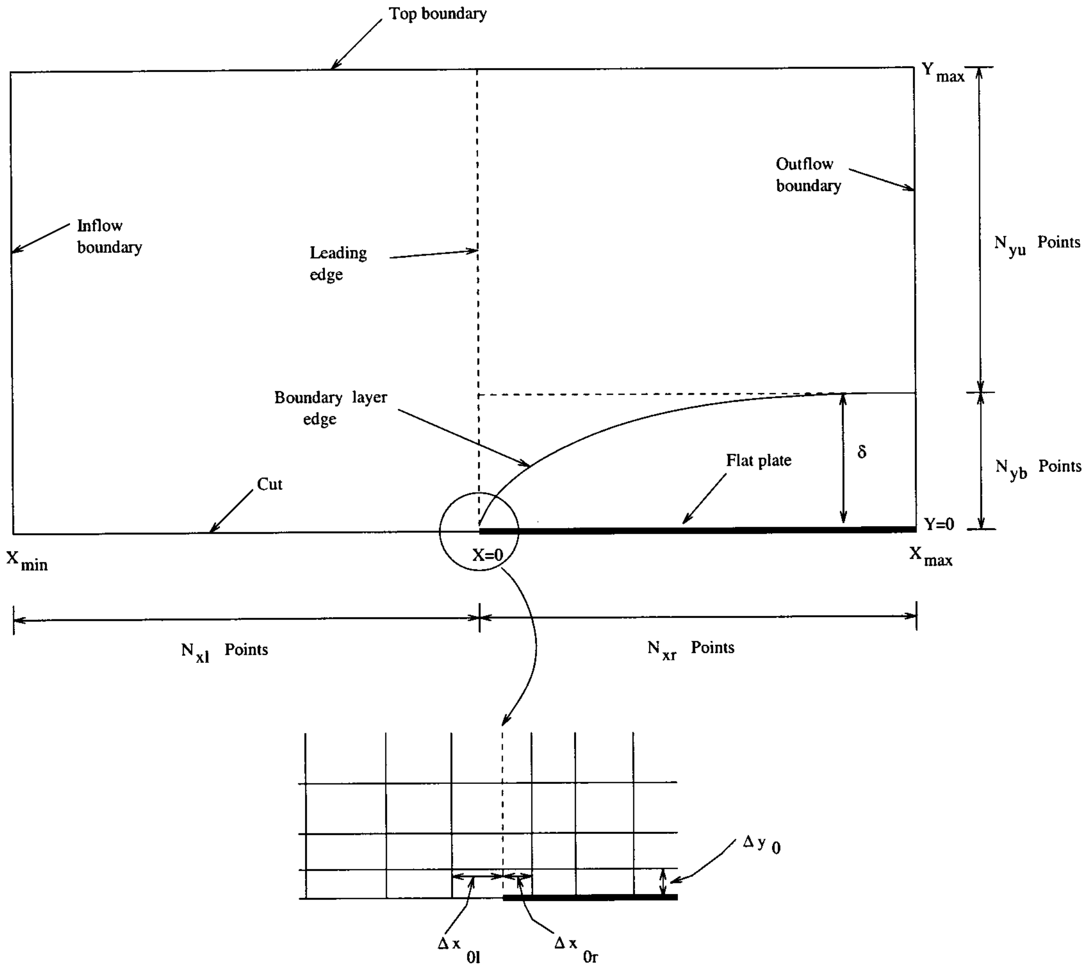

Figure 8. Schematic diagram of the flat plate geometry. The flat plate occupies the region from $X = 0$ to $X = X_{\max}$ and $X_{\max} = 1.0$.

For the mean-flow equations, the inflow, outflow, and top boundaries are treated with Riemann invariant based boundary conditions [14], and along the flat plate both components of the velocity and the normal derivative of the temperature are set to zero. Along the cut before the leading edge of the plate (Figure 8), the transverse velocity and transverse gradients of the temperature and streamwise velocity were set to zero. For the k–ϵ equations, the condition $k = \epsilon = 0$ is prescribed along the flat plate and the variables are either extrapolated (linearly for FOU and quadratically for TVD schemes) or set to zero (on the assumption that the flow at infinity is laminar) at the top and outflow boundaries (Figure 8) depending on

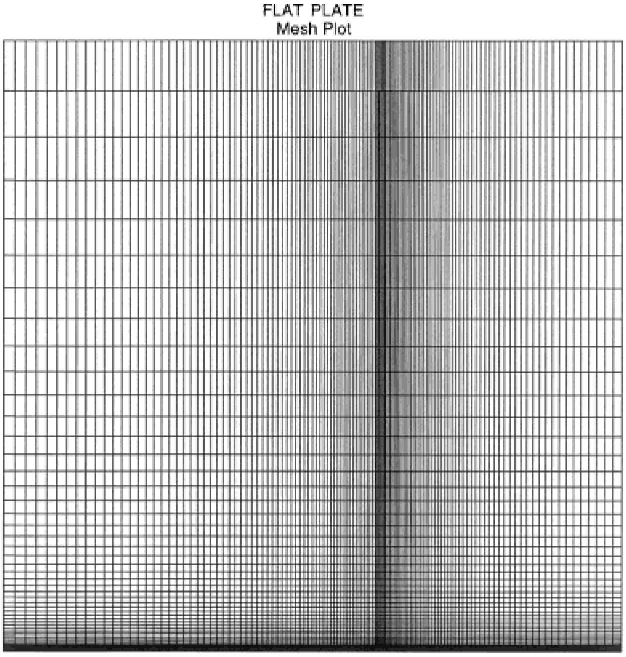

Figure 9. Plot of the mesh (129 × 81 nodes) over the flat plate.

whether the boundary grid points satisfy locally outflow or inflow conditions. The initial conditions for the mean flow field are provided by a turbulent flow solution obtained by using the algebraic Baldwin–Lomax turbulence model and those for the k–ϵ field are given by the free-stream k and ϵ values.

At the start of the computations, the k–ϵ equations are iterated by themselves keeping the mean flow field fixed (frozen). Then, once the k–ϵ flow field has developed sufficiently, the two systems of equations are iterated together by performing a mean-flow variable time update (iteration) followed by a k–ϵ variable time update (although there are many other ways of iterating the mean-flow and k–ϵ equation systems—see Reference [14]).

By adjusting the level of the free-stream k and ϵ values, it is possible to make the Chien k–ϵ model predict transition at different locations. However, it is seen to occur too early [14], which is a well-known property of this model. For a flat plate geometry, if the Reynolds number is known, the position (past the leading edge) where transition occurs can be calculated (for $Re_\infty = 5 \times 10^6$ and $Tu = 1.0$ per cent, transition occurs at 13.36 per cent of the plate length past the leading edge). A procedure also described in Reference [14], which involves the setting of the eddy viscosity and the production terms in the k and ϵ equations to zero before the transition grid point, rectifies the shortcoming of the model.

Numerical results with a free-stream Reynolds number (based on plate length) of 5×10^6 and Mach numbers of 0.2, 0.8, 1.5 for fully and partly turbulent flat plate flows are presented in Reference [14]. The greatest differences between the numerical and empirical results are seen to occur for the supersonic case. Hence, the results for this test case and that involving transition are presented in this paper.

The convergence histories for the iteration of the k–ϵ equations for a fixed (frozen) Navier–Stokes flow field and for the iteration of the two systems together are given in Figures 10 and 11 respectively; a tolerance of 1.0×10^{-12} was selected for the former as it is sufficiently close to machine (double) precision and for the latter the flow was considered to have reached the steady state when the global node-based residual norm $\|R\|$ associated with the Navier–Stokes flow quantities fell below 1.0×10^{-6}.

The convergence results presented in Table VII indicate that in all cases the numerical algorithms are stable and convergent and Multigrid (using the optimal parameters) is effective in accelerating convergence and the speed-up factors are generally greater for the higher-order (TVD) computations than the first-order (FOU) ones. Furthermore, in terms of the number of iterations (work units), the TVD computations are not significantly more expensive than their FOU counterparts.

The distribution of the skin-friction coefficients (c_f) given in Figures 12 and 13 indicate that for the supersonic test case even with 40 grid points in the boundary layer, solutions of FOU and TVD schemes differ (but not drastically) with the latter giving a better agreement with the empirical estimate (the Van Driest relation [40]). For the test case involving transition (Mach number of 0.8), beyond transition the two sets of numerical predictions agree with each other and they are also in excellent agreement with the empirical prediction; some evidence showing the presence of numerical diffusion in the FOU solution is found in Figure 13.

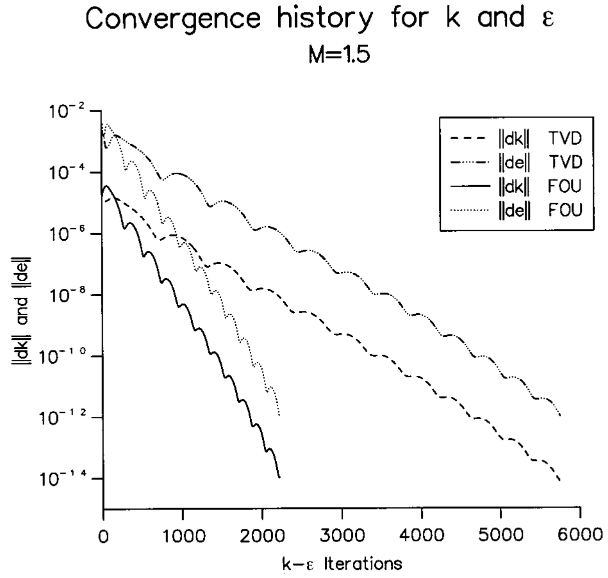

Figure 10. Convergence histories for k and ϵ when the k–ϵ equations are iterated by themselves (Mach number = 1.5).

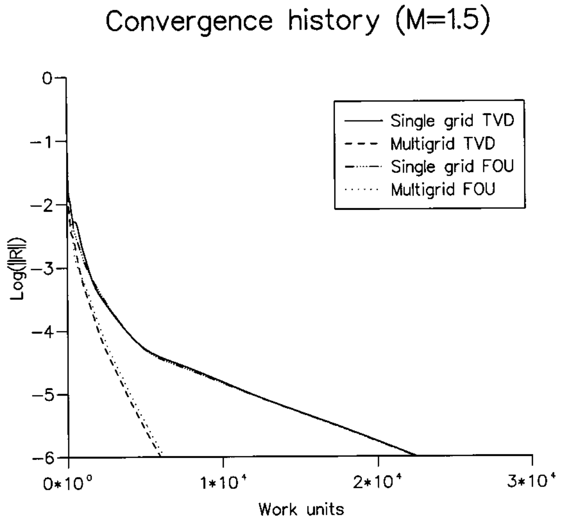

Figure 11. Convergence histories for the mean-flow equations (Mach number = 1.5) in terms of work units on the finest grid.

Table VII. Summary of convergence results (in terms of work units on the finest grid) for fully and partly turbulent flat plate flows.

Mach number	TVD			FOU		
	Single grid	Multigrid	Speed-up factor	Single grid	Multigrid	Speed-up factor
0.2	128 661	47 775	2.69	116 671	42 269	2.76
0.8	70 911	14 705	4.82	65 631	15 223	4.31
1.5	22 506	5973	3.77	22 411	6207	3.61
0.8 (with transition)	73 791	14 196	5.20	69 496	14 725	4.72

The surface temperature (T) plots given in Figures 14 and 15 are indicative of excessive numerical diffusion present in the solutions due to the FOU scheme but otherwise the two sets of numerical solutions are in agreement. The empirical estimate is the recovery temperature for an adiabatic, zero pressure gradient flat plate [14,41,42].

The plots of k, ϵ, and the eddy (turbulent) viscosity (μ_t) across the boundary layer presented in Figures 16–18 too depict the excessive numerical diffusion present in the FOU scheme; k and ϵ have been plotted against a logarithmic scale in order to highlight the behavior within the boundary layer more clearly. These profiles also resemble those in Reference [34].

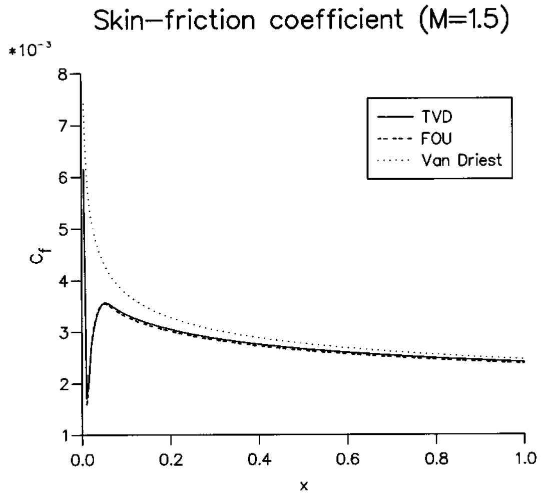

Figure 12. Skin-friction coefficients for fully turbulent flow with a Mach number of 1.5.

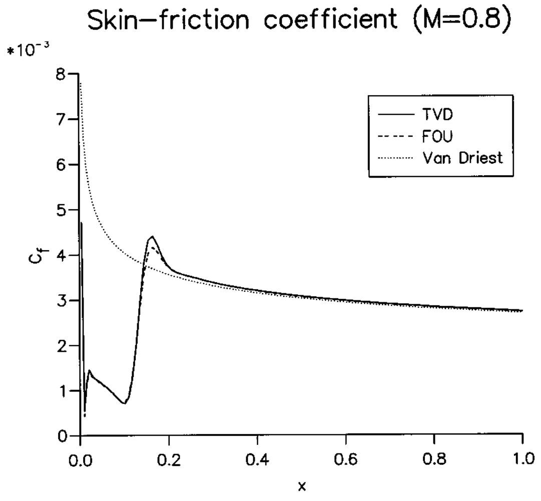

Figure 13. Skin-friction coefficients for turbulent flow with transition at a Mach number of 0.8.

The boundary layer profiles of the dimensionless streamwise velocity u^+ ($=u/u_\tau$ where u_τ is the friction velocity) in Figures 19 and 20 indicate that at both x stations, the agreement of the numerical solutions with each other and with the empirical predictions (given in Reference [14,41]) is excellent in the inner regions of the boundary layer. In the outer regions, the

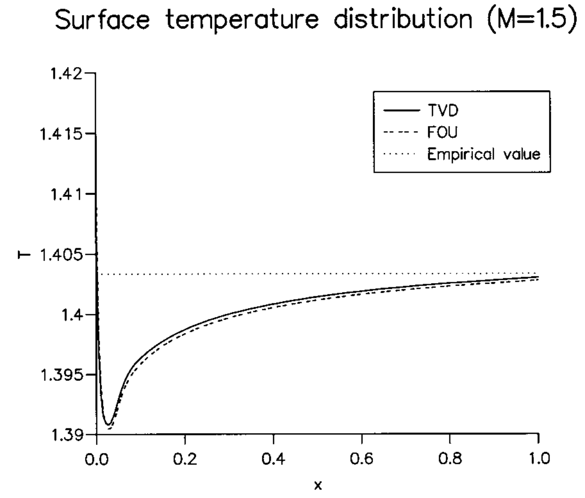

Figure 14. Surface temperature distribution for fully turbulent flow with a Mach number of 1.5.

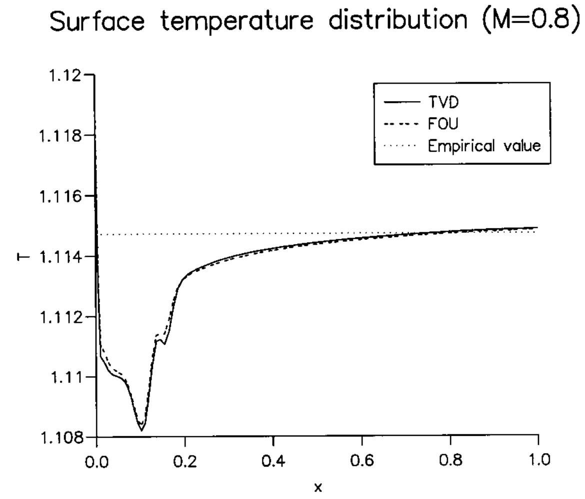

Figure 15. Surface temperature distribution for turbulent flow with transition at a Mach number of 0.8.

numerical predictions at $x = 0.208$ are poor and those at $x = 0.803$ are better. Moreover, in all cases, the solutions due to the TVD and FOU schemes are in close agreement.

In the computations described above, the pointwise time steps used in the k–ϵ algorithm were typically 80 per cent of those predicted by the analysis. In addition to ensuring the

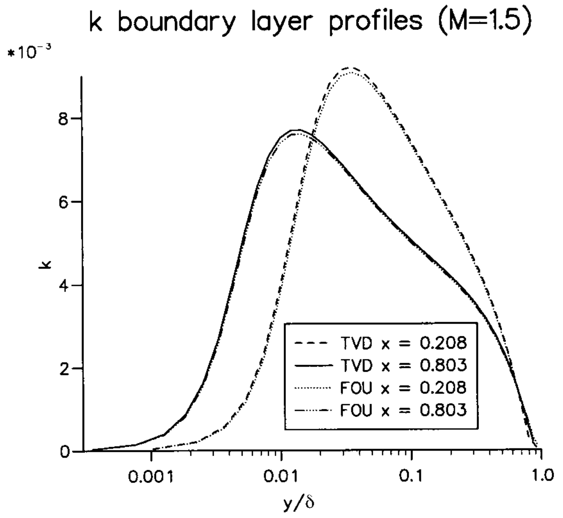

Figure 16. Boundary layer profiles of k for the test case with a Mach number of 1.5 (δ = boundary layer thickness at x station).

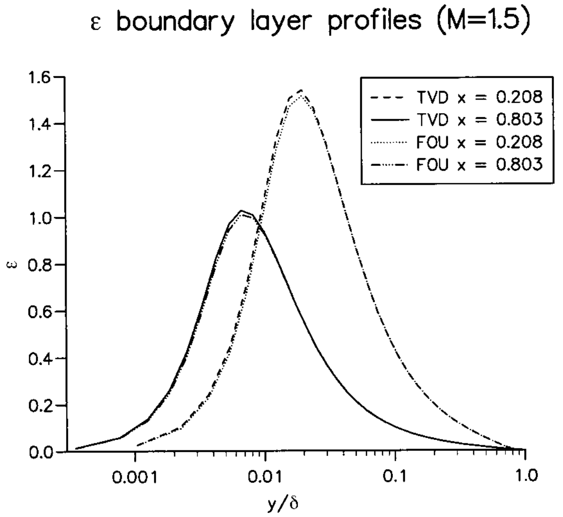

Figure 17. Boundary layer profiles of ϵ for the test case with a Mach number of 1.5 (δ = boundary layer thickness at x station).

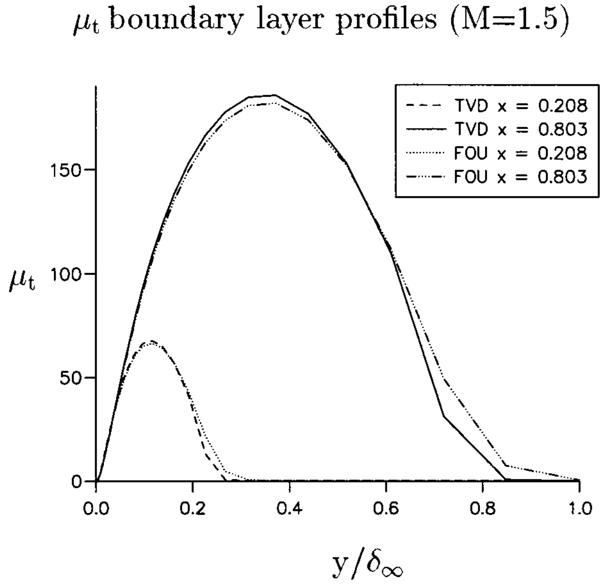

Figure 18. Boundary layer profiles of μ_t for the test case with a Mach number of 1.5. (δ_∞ = boundary layer thickness at $x = 1$).

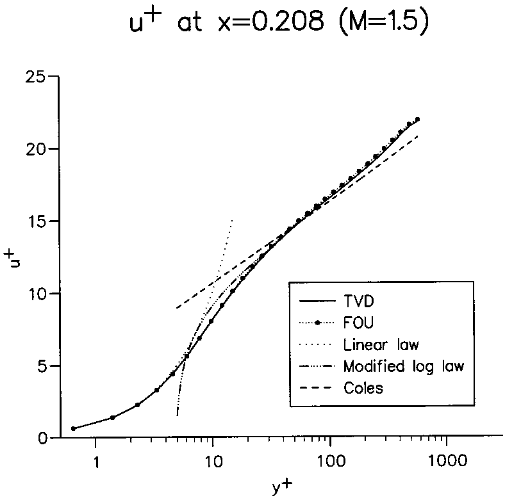

Figure 19. u^+ boundary layer profiles at $x = 0.208$ for the test case with a Mach number of 1.5 (y^+ = dimensionless wall co-ordinate).

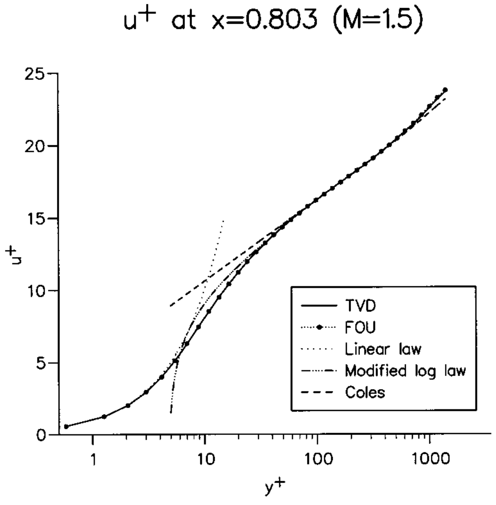

Figure 20. u^+ boundary layer profiles at $x = 0.803$ for the test case with a Mach number of 1.5 (y^+ = dimensionless wall co-ordinate).

stability and convergence of the algorithms, these time steps also maintained k and ϵ positive-valued and no *ad hoc* procedures (see, for example, Reference [32]) were required to prevent k and ϵ from becoming negative-valued.

8. CONCLUSIONS

A class of positivity preserving point implicit methods applicable to source term dominated systems of differential equations has been constructed in this paper. These methods are optimally suited for achieving steady state numerical solutions. The analysis of the discrete (pointwise) equations in a simplified manner led to the determination of very reliable time step restrictions that ensured the numerical stability and positivity preservation of the algorithms and the existence of and convergence to positive-valued roots. The point implicit treatment of the source terms did not adversely affect the positivity preservation, accuracy, or convergence properties of the numerical schemes when post-first-order accurate TVD schemes were applied to the discretization of convection. The operator splitting technique was adopted to handle multidimensional convection terms. Numerical experimentation with model problems confirmed the validity of the analysis and the computational efficiency associated with the third-order TVD scheme. The Newton method was successful in capturing the desired root(s).

The extension of the model problem ideas to the stiff Chien k–ϵ system too produced convergent and genuinely positivity preserving algorithms; no *ad hoc* procedures were required to maintain the positiveness of k and ϵ. The algorithms were able to cope with very general free-stream and initial k and ϵ values. Even though Multigrid was applied only to the mean-flow equations, an appreciable acceleration in convergence was observed. The numerical solutions indicated that although 40 points were placed within the boundary layer, the higher-order solutions still differed (but not significantly) from the first-order accurate ones; the lack of larger differences is due to the simple nature of this geometry, where the flow is also grid-aligned (some non-grid-aligned flows are given in Reference [43]). With regard to the convergence properties, particularly for higher Mach numbers, the higher-order accurate algorithm was as good as and if not better than the first-order accurate one. Therefore, it can be concluded that the approximation of convection with a higher-order, positivity preserving numerical scheme and the pointwise implicit treatment of the source terms in the solution of the Chien $k-\epsilon$ turbulence model even for geometries as simple as the flat plate are beneficial.

ACKNOWLEDGMENTS

The author would like to express his thanks and gratitude to Professor K.W. Morton for his guidance, support and encouragement throughout the research. The author would also like to thank Professor M.B. Giles for his guidance, advice and for teaching much about turbulence modeling. The help and support offered by Dr P.I. Crumpton is also gratefully acknowledged. Thanks and gratitude go to the Harold Hyam Wingate Foundation, the Overseas Research Students Awards Scheme, the Defense Research Agency (Farnborough), the Stuart Fund (Balliol College, Oxford), and the Oxford Overseas Bursary Scheme for their financial support.

APPENDIX A. PSEUDO-TIME STEP RESTRICTIONS FOR THE ONE-DIMENSIONAL SCALAR MODEL PROBLEM

When the non-linear model equation

$$[U_t] + aU_x - bU_{xx} = cU(1 - U); \quad a, b, c \ \text{(global constants)} > 0 \tag{31}$$

is discretized using the explicit, semi-implicit, and Gauss–Seidel convection–diffusion schemes on a uniform mesh with spacing Δx, the resulting iterations with $v = a\Delta t_j/\Delta x$, $\mu = b\Delta t_j/\Delta x^2$, and $z = U_j^{n+1}$ respectively are

$$f(z) := c\Delta t_j z^2 + (1 - c\Delta t_j)z + (C_j^n \Delta t_j - U_j^n) = 0 \tag{32}$$

$$\tilde{f}(z) := c\Delta t_j z^2 + (1 + v + 2\mu - c\Delta t_j)z + (\tilde{C}_j^n \Delta t_j - U_j^n) = 0 \tag{33}$$

$$\hat{f}(z) := c\Delta t_j z^2 + (1 + v + 2\mu - c\Delta t_j)z + (\hat{C}_j^{n+*} \Delta t_j - U_j^n) = 0 \tag{34}$$

where

$$C_j^n = \frac{a}{\Delta x}(U_j^n - U_{j-1}^n) - \frac{b}{\Delta x^2}(U_{j+1}^n - 2U_j^n + U_{j-1}^n) \tag{35}$$

$$\tilde{C}_j^n = -\frac{a}{\Delta x} U_{j-1}^n - \frac{b}{\Delta x^2}(U_{j+1}^n + U_{j-1}^n) \tag{36}$$

$$\hat{C}_j^{n+*} = -\frac{a}{\Delta x} U_{j-1}^{n+1} - \frac{b}{\Delta x^2}(U_{j+1}^n + U_{j-1}^{n+1}) \tag{37}$$

and it is observed that $\tilde{C}_j^n$, $\hat{C}_j^{n+*} < 0$ on the assumption that (i) $U_j^n > 0$ $\forall j$—that is, the previous time values of U are positive, and (ii) $U_{j-1}^{n+1} > 0$, which implies that the latest updates at grid locations before node j are also positive.

In each case, the discrete equations can be readily solved at each mesh point exactly but the Newton method will be employed to do so because the eventual aim is to solve the non-linear k–ϵ equations; the root to which the Newton method converges should be positive-valued (Section 2).

A.1. Time step restrictions

The aim when solving the point equations at each mesh point is to have only a single real positive root (time update).

Taking into account the quadratic nature of Equation (32) with the explicit discretization of the convection–diffusion terms, a single real positive root exists when $\Delta t_j < \Delta t_j^{+e}$ and

$$\begin{cases} \Delta t_j^{+e} = \dfrac{U_j^n}{C_j^n}, & C_j^n > 0 \\ \Delta t_j^{+e} = \infty, & C_j^n \leq 0 \end{cases} \tag{38}$$

For the semi-implicit and Gauss–Seidel discretizations of the convection–diffusion terms, however, there is no limit on Δt_j (that is, $\Delta t^+ = \infty$) in order to obtain a single real positive root because $\tilde{C}_j^n$, $\hat{C}_j^{n+*} < 0$.

The linearized version of Equation (5) in the neighborhood of the asymptotically stable root $U = 1$ (letting $U = 1 + u$) is

$$[u_t] + au_x - bu_{xx} = -cu \tag{39}$$

and it has the same form as Equation (4), which was studied before and hence for this equation, modified maximum principles apply to algorithms with

(a) the explicit discretization of the convection–diffusion terms when $v + 2\mu \leq 1$ so that $\Delta t_j < \Delta t^{Me}$, where

$$\Delta t^{Me} = \left(\frac{a}{\Delta x} + \frac{2b}{\Delta x^2}\right)^{-1} \tag{40}$$

(b) the semi-implicit and Gauss–Seidel discretizations of the convection–diffusion terms when $1+v+2\mu+c\Delta t_j>0$, showing that there are no restrictions placed on Δt_j (as $c>0$). Therefore, as expected, (b) is less restrictive than (a).

Now the time step restrictions for the solution of the point equations with the Newton method will be described. When the Newton method is employed to solve Equations (32)–(34), if $g(z)$ represents $f(z)$, $\tilde{f}(z)$ or $\hat{f}(z)$, then the $m+1$ Newton iteration is

$$z^{m+1}=z^m-\frac{g(z^m)}{g'(z^m)} \tag{41}$$

which at node j takes the following forms:

(a) Explicit convection–diffusion

$$z^{m+1}=\frac{c\Delta t_j(z^m)^2+(U_j^n-C_j^n\Delta t_j)}{1+c\Delta t_j(2z^m-1)} \tag{42}$$

(b) Semi-implicit convection–diffusion

$$z^{m+1}=\frac{c\Delta t_j(z^m)^2+(U_j^n-\tilde{C}_j^n\Delta t_j)}{1+v+2\mu+c\Delta t_j(2z^m-1)} \tag{43}$$

(c) Gauss–Seidel convection–diffusion

$$z^{m+1}=\frac{c\Delta t_j(z^m)^2+(U_j^n-\hat{C}_j^{n+*}\Delta t_j)}{1+v+2\mu+c\Delta t_j(2z^m-1)} \tag{44}$$

Therefore, in order to maintain the Newton iterates, z^{m+1} positive (assuming $z^m>0$):

1. if $z^m>1/2$, the explicit discretization may need a time step restriction depending on whether $U_j^n+(c(z^m)^2-C_j^n)\Delta t_j \lesseqgtr 0$; however, the time step limit (38) that is employed for the outer iteration ensures that the Newton iterates will be positive. For $z^m>1/2$, the semi-implicit and Gauss–Seidel discretizations do not place any limits on Δt_j because $\tilde{C}_j^n$, $\hat{C}_j^{n+*}<0$; and
2. if $0<z^m<1/2$, the explicit discretization requires the additional condition (in addition to having $U_j^n+(c(z^m)^2-C_j^n)\Delta t_j>0$)

$$\Delta t_j<\frac{1}{c(1-2z^m)}=\Delta t_j^{Ne} \tag{45}$$

The restrictions on Δt_j due to the semi-implicit discretization are

$$1+v+2\mu+c(2z^m-1)\Delta t_j=1+\alpha\Delta t_j>0 \tag{46}$$

with $\alpha=a/\Delta x+2b/\Delta x^2+c(2z^m-1)$. Then, if $\alpha>0$, there is no limit on Δt_j, but when $\alpha<0$

$$\Delta t_j < \frac{1}{|\alpha|} = \Delta t_j^{Ns} \tag{47}$$

If, however, it is assumed that the mesh is sufficiently refined, then the term $a/\Delta x + 2b/\Delta x^2$ will dominate and cause α to be positive-valued. The corresponding maximum permitted time step, Δt^{Ng}, for the Gauss–Seidel discretization has the same form as that for the semi-implicit one. Therefore, there are again advantages of using the semi-implicit and Gauss–Seidel discretizations for the convection–diffusion terms.

The analysis for this one-dimensional scalar model problem can be summarized as follows:

Proposition
For the one-dimensional scalar model problem, at each node j

1. it is guaranteed that there will exist a single real positive root,
2. the Newton method with a positive starting guess will converge to this positive-valued root by generating only positive-valued iterates (the inner iteration), and
3. the outer iteration for the corresponding linearized equation will obey a modified maximum principle that ensures the numerical stability and positivity preservation of the scheme if the following conditions are satisfied:
   (I) when the Newton iterates are less than one half in magnitude, for the explicit discretization if $\Delta t_j < \min(\Delta t_j^{+e}, \Delta t^{Me}, \Delta t_j^{Ne})$, for the semi-implicit discretization if $\Delta t_j < \Delta t_j^{Na}$, and for the Gauss–Seidel discretizations if $\Delta t_j < \Delta t_j^{Ng}$; and
   (II) when the Newton iterates are greater than one half in magnitude, for the explicit discretization if $\Delta t_j < \min(\Delta t_j^{+e}, \Delta t^{Me})$ and for the semi-implicit and Gauss–Seidel discretizations there are no limits on Δt_j.

APPENDIX B. THE CHIEN k–ϵ MODEL OF TURBULENCE

The dimensionless Chien version of the k–ϵ model is [14]

$$\frac{\partial(\bar{\rho}k)}{\partial t} + \frac{\partial}{\partial x_\alpha}\left\{\bar{\rho}\tilde{V}_\alpha k - \frac{\mu_k}{Re_\infty}\frac{\partial k}{\partial x_\alpha}\right\} = S_k \tag{48}$$

$$\frac{\partial(\bar{\rho}\epsilon)}{\partial t} + \frac{\partial}{\partial x_\alpha}\left\{\bar{\rho}\tilde{V}_\alpha \epsilon - \frac{\mu_\epsilon}{Re_\infty}\frac{\partial \epsilon}{\partial x_\alpha}\right\} = S_\epsilon \tag{49}$$

where

$$\mu_k = \frac{\mu + \mu_t}{Pr_k}$$

$$\mu_\epsilon = \frac{\mu + \mu_t}{Pr_\epsilon}$$

$$\mu_t = C_\mu f_\mu \bar{\rho} \frac{k^2}{\epsilon} Re_\infty$$

and the source terms S_k, S_ϵ are

$$S_k = C_\mu f_\mu \Sigma \bar{\rho} \frac{k^2}{\epsilon} - \frac{2}{3} \bar{\rho} \left(\frac{\partial \tilde{V}_\gamma}{\partial x_\gamma} \right) k - \bar{\rho}\epsilon - \frac{2\mu k}{Re_\infty y_n^2} \tag{50}$$

$$S_\epsilon = C_1 C_\mu f_\mu \Sigma \bar{\rho} k - \frac{2}{3} C_1 \bar{\rho} \left(\frac{\partial \tilde{V}_\gamma}{\partial x_\gamma} \right) \epsilon - C_2 E \bar{\rho} \frac{\epsilon^2}{k} - \frac{2\mu F \epsilon}{Re_\infty y_n^2} \tag{51}$$

with $\Sigma = \tilde{S}_{\alpha\beta} \partial \tilde{V}_\alpha / \partial x_\beta$ and the following damping functions and set of model constants:

$$f_\mu = 1 - \exp(-0.0115 y^+), \quad E = 1 - \frac{2}{9} \exp\left(-\frac{R_t^2}{36} \right), \quad F = \exp(-0.5 y^+)$$

$$C_\mu = 0.09, \quad C_1 = 1.35, \quad C_2 = 1.8, \quad Pr_k = 1.0, \quad Pr_\epsilon = 1.3$$

In the above model; $\bar{\rho}$ is the Reynolds averaged density, $\tilde{V}_\alpha$ is the mass-averaged velocity, $\tilde{S}_{\alpha\beta} = (\tilde{V}_{\alpha,\beta} + \tilde{V}_{\beta,\alpha}) - \frac{2}{3}\delta_{\alpha\beta} \tilde{V}_{\gamma,\gamma}$ is the compressible strain tensor, y_n is the normal distance from a solid boundary, μ is the molecular viscosity, k is the turbulent kinetic energy, ϵ is the rate of dissipation of k, Pr_k, Pr_ϵ are the turbulent Prandtl numbers, $R_t = \bar{\rho} k^2 / \mu\epsilon$ is the turbulence Reynolds number, and Re_∞ is the free-stream Reynolds number.

REFERENCES

1. Godunov SK. A difference scheme for numerical computation of discontinuous solution of hydrodynamic equations. *Mathematics of the USSR-Sbornik* 1959; **47**: 357–393.
2. Harten A. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics* 1983; **49**: 357–393.
3. Hirsch C. *Numerical Computation of Internal and External Flows*, vol. 1. Wiley: New York, 1988; 488–500.
4. Yang HQ, Przekwas AJ. A comparative study of advanced shock-capturing schemes applied to Burger's equation. *Journal of Computational Physics* 1992; **102**: 139–159.
5. Thuburn J. Multidimensional flux-limited advection schemes. *Journal of Computational Physics* 1996; **123**: 74–83.
6. Thuburn J. Dissipation and cascades to small scales in numerical models using a shape preserving advection scheme. *Monthly Weather Review* 1995; **123**: 1888.
7. Lock AP, Leonard BP, MacVean MK. Positivity-preserving numerical schemes for multidimensional advection. Technical Report TM 106055/ICOMP-9305, NASA/ICOMP, 1993.
8. Chakravarthy SR, Szema KY, Goldberg UC, Gorski JJ. Application of a new class of high accuracy TVD schemes to the Navier–Stokes equations. AIAA Paper No. 85-0165, 1985.
9. Chuang CC, Chieng CC. Transonic turbulent seperated flow predictions using a two-layer turbulence model. *AIAA Journal* 1993; **31**: 816–817.
10. Gorski JJ, Chakravarthy SR, Goldberg UC. High accuracy TVD schemes for the k–ϵ equations of turbulence. AIAA Paper No. 85-1665, 1985.

11. Yoon BK, Chung MK. Computation of compressible ramp flow with a cross-diffusion modified k–ϵ model. *AIAA Journal* 1995; **33**: 1518–1520.
12. Zijlema M. On the construction of a third order accurate monotone convection scheme with the application to turbulent flows in general domains. *International Journal for Numerical Methods in Fluids* 1996; **22**: 619–641.
13. Zijlema M, Segal A, Wesseling P. Invariant discretisation of the k–ϵ model in general co-ordinates for prediction of turbulent flow in complicated geometries. *Computers and Fluids* 1995; **24**: 209–225.
14. Lanerolle LWJ. Numerical modelling of turbulent compressible flow. PhD thesis, Numerical Analysis Group, University of Oxford, 1997.
15. Chien KY. Prediction of channel and boundary layer flows with a low Reynolds number turbulence model. *AIAA Journal* 1982; **20**: 33–38.
16. Coakley TJ. Turbulence modelling methods for the compressible Navier–Stokes equations. AIAA Paper No. 83-1693, 1983.
17. Vandromme D, Ha Minh H. Turbulence modelling for compressible flows (Series 1987-06). In *Von Karman Institute for Fluid Dynamics Lecture Series*, Benocci C (ed.). Von Karman Institute: Rhode St. Genèse, Belgium, 1987.
18. Chakravarthy SR, Osher S. Computing with high resolution upwind schemes for hyperbolic equations. *Lectures in Applied Mathematics* 1985; **22**: 57–86.
19. Chakravarthy SR. High resolution upwind formulations for the Navier–Stokes equations. Technical Report, Rockwell Interational Science Centre, 1987.
20. Varga RS. *Matrix Iterative Analysis*. Prentice Hall: Englewood Cliffs, NJ, 1962.
21. Kreyszig E. *Advanced Engineering Mathematics* (6th edn). Wiley: New York, 1988; 154–179.
22. Sánchez DA. *Ordinary Differential Equations and Stability Theory: An Introduction*. Dover: New York, 1968; 68–90.
23. Ross SL. *Differential Equations* (3rd edn). Wiley: New York, 1984; 632–714.
24. Chakravarthy SR, Osher S. Numerical experimentation with the Osher upwind scheme for the Euler equations. *AIAA Journal* 1983; **21**: 1241–1248.
25. Gustafsson B. The convergence rate for difference approximations to mixed initial boundary value problems. *Mathematics of Computation* 1975; **59**: 396–406.
26. Humphrey JAC, Hayase T, Greif R. A consistently formulated QUICK scheme for fast and stable convergence using finite-volume iterative calculation procedures. *Journal of Computational Physics* 1992; **98**: 108.
27. Fletcher CAJ. *Computational Techniques for Fluid Dynamics*, vol. II. Springer: Berlin, 1988; 46–77.
28. Fletcher CAJ. *Computational Techniques in Aerodynamics*. Springer: Berlin, 1996; 172–190.
29. Kunz RF, Lakshminarayana B. Explicit Navier–Stokes computation of cascade flows using the k–ϵ turbulence model. *AIAA Journal* 1992; **30**: 13–22.
30. Viegas JR, Horstman CC. Comparison of multi-equation turbulence models for several shock separated boundary-layer interation flows. AIAA Paper No. 78-1165, 1978.
31. Viegas JR, Coakley TJ. Numerical investigation of turbulence models for shock separated boundary-layer flows. AIAA Paper No. 77-44, 1977.
32. Gerolymos GA. Implicit multiple-grid solution of the compressible Navier–Stokes equations using k–ϵ turbulence closure. *AIAA Journal* 1990; **28**: 1707–1717.
33. Vandromme D. About the coupling of turbulence closure models with averaged Navier–Stokes equations. *Journal of Computational Physics* 1986; **65**: 386–409.
34. Siikonen T. An application of Roe's flux-difference splitting for k–ϵ turbulence model. *International Journal for Numerical Methods in Fluids* 1995; **21**: 1017–1039.
35. Vandromme D, Ha Minh H. Turbulence modelling for compressible flows (Short Course 1985-01). In *Von Karman Institute for Fluid Dynamics Lecture Series*, Benocci C (ed.). Von Karman Institute: Rhode St. Genèse, Belgium, 1985.
36. Mohammadi B, Pironneau O. *Analysis of the K-Epsilon Turbulence Model*. Wiley: New York, 1994; 25, 29–32, 103–111, 133–134, 144.
37. Wilcox DC. *Turbulence Modeling for CFD*. DCW Industries Incorporated: La Canada, CA, 1993.
38. Crumpton PI, Mackenzie JA, Morton KW. Cell vertex algorithms for the compressible Navier–Stokes equations. *Journal of Computational Physics* 1993; **109**: 1–15.
39. Brandt J. Multi-level adaptive solutions to boundary value problems. *Mathematics of Computation* 1977; **31**: 333–390.

40. Radespiel R, Rossow C. A cell vertex finite volume scheme for the two-dimensional Navier–Stokes equations. Technical Report IB 129-87/40, DFVLR, 1987.
41. Young AD. *Boundary Layers*. BSP Professional Books, 1989; 6, 10–29, 107–109, 141–142, 162–164, 215, 237.
42. Schlichting H. *Boundary-Layer Theory*. McGraw-Hill: New York, 1979; 127–149, 557, 596–667, 702–728.
43. LWJ Lanerolle. High resolution multiple grid computation of turbulent shock wave/boundary layer interactions with the k–ϵ turbulence model. *International Journal for Computational Fluid Dynamics* (in revision).