Preserving Positivity for Hyperbolic PDEs Using Variable-Order Finite Elements

M. Berzins

SCI Institute, School of Computing, University of Utah, Salt Lake City, Utah

Dedicated to Joe Flaherty on the occasion of his 60th birthday.

Abstract

The positivity preserving approach of Berzins is generalized by using a derivation based on bounded polynomial approximations and order selection. The approach is extended from the B-spline based methods used previously to the use of more conventional continuous Galerkin elements. The conditions relating to positivity preservation are considered and a numerical example used to demonstrate the performance of the method on a model advection equation problem.

Key words: Hyperbolic equations, finite elements, positivity

1 Introduction

This paper follows earlier work of Berzins [3,4] which is concerned with the development of positivity preserving finite element methods for the solution of hyperbolic equations in one space variable. The focus in this paper again will again be the simple advection equation with non-negative initial data: $\frac{\partial U}{\partial t} + \frac{\partial U}{\partial x} = 0$ with appropriate initial and boundary conditions on a spatial interval $[x_l, x_r]$. Applying the standard Galerkin method with basis functions $\phi_i(x)$ on a mesh $x_i, i = 1, ..., N$ gives

$$\int_{x_{i-1}}^{x_{i+1}} \frac{\partial U}{\partial t} \phi_i(x)\, dx = \int_{x_{i-1}}^{x_{i+1}} -\frac{\partial U}{\partial x} \phi_i(x)\, dx, i = 1, ..., N, \tag{1}$$

Email address: mb@sci.utah.edu (M. Berzins).

where the approximate finite element solution to this p.d.e. is defined by $U(x,t) = \sum_{i=1}^{N} \phi_i(x) U_i(t)$.

Evaluation of the integrals, use of the initial and Dirichlet boundary conditions and defining the time-dependent vector $\underline{U}$ by $\underline{U} = [U_1, ..., U_N]^T$ where $\dot{U}_i = \frac{dU_i}{dt}$. gives rise to the numerical scheme defined by the system of equations:

$$A\underline{\dot{U}}(t) = \underline{F}(\underline{U}(t)) \tag{2}$$

where the matrix A is referred to as the mass matrix.

It is well-known that the standard Galerkin method is unsatisfactory for hyperbolic equations in a very similar way to that of linear central difference schemes, [19].

An important aspect of this poor performance is that the method does not preserve positivity with regard to the standard definition used here for a positivity preserving scheme for the advection equation. This definition requires (see Laney [17]) that the numerical solution at time t_{n+1} may be written in terms of the numerical solution at time t_n in the form

$$U_i(t_{n+1}) = \sum_j a_j U_j(t_n) \text{ where } \sum_j a_j = 1, \text{ and } a_j \geq 0 \,. \tag{3}$$

The key observation with regard to preserving positivity is due to Godunov [9] who proved that any scheme of better than first order which preserves positivity for the advection equation must be nonlinear. For example, the coefficients a_j in (3) above must depend on the numerical solution to the p.d.e.

There are recent papers addressing positivity preservation are referenced by Berzins [4] who also notes that the approach suggested in that paper differs from most of the others, but is perhaps closer to the method of Cockburn and Shu [7]. Other methods based on finite elements but taking a different approach to the one described here are those of Sheu et al. [22,23]. The simplest approach for deriving positivity preserving schemes goes back for steady state problems at least as far as Harten and Zvas [13] and is discussed in Chapter 22 of [17]. The idea is simply to use a scheme only when it preserves positivity and otherwise to switch to a more suitable scheme.

This paper extends the approach of Berzins by using a bounded polynomial approximation framework to derive the methods and extends the earlier work in which a family of Galerkin B-spline methods were modified to preserve positivity to the case of more conventional Galerkin methods.

2 H-P Finite Element Spaces

The approximation space used here differs from two standard choices of h-p finite element spaces in that for basis functions of degree p the pth derivative is allowed to be discontinuous. Let T_h be a subdivision of $[a,b]$ into individual elements. The approximation space is given by S_h where

$$S_h = S^{p,-1}([a,b],T_h) = \{v; v|_K \in P_p(K) \forall K \in T_h\} \tag{4}$$

Within this space are contained not only the usual discontinuous Galerkin functions but also the standard continuous Galerkin methods.

2.1 Interpolation Results.

Consider the case in which we have an approximating polynomial $U^I(x)$ based on a set of nodal values $U(x_i)$ and defined using divided differences as defined by the usual notation

$$U[x_i, x_{i+1}, ..., x_{i+k}] = \frac{U[x_{i+1}, x_{i+2} ..., x_{i+k}] - U[x_i, x_{i+1}, ..., x_{i+k-1}]}{x_{i+k} - x_i} \tag{5}$$

where $U[x_i] = U(x_i)$ and

$$U[x_i, x_{i+1}] = \frac{U[x_{i+1}] - U[x_i]}{x_{i+1} - x_i} \tag{6}$$

Suppose that a set of mesh points are given by $x_0, x_1, x_2, x_3, x_4 ...$ with associated solution values $U[x_0], ..., U[x_4] ...$ then the standard Newton divided difference form of the interpolating polynomial is given by

$$\begin{aligned} U^I(x) = U[x_0] + (x-x_0)U[x_0,x_1] + (x-x_0)(x-x_1)U[x_0,x_1,x_2] \\ +(x-x_0)(x-x_1)(x-x_2)U[x_0,x_1,x_2,x_3] + ... + \\ (x-x_0)...(x-x_{n-1})\; U[x_0, ...x_n] \end{aligned} \tag{7}$$

It is then possible to order the point so that the points $x_0, x_1, x_2, x_3, x_4 ...$ map onto the mesh points close to x_i as $x_i, x_{i-1}, x_{i+1}, x_{i-2}, x_{i+2} ...$ to give the following form of the approximating polynomial , see Hildebrand [15] to rewrite the approximating polynomial in divided difference form as either left or right biased interpolants denoted by $U_L^I(x)$ and $U_R^I(x)$ respectively where

$$\begin{aligned} U_L^I(x) = U[x_i] + b_{1,i}(x)\, U[x_{i-1},x_i] + b_{2,i}(x)\, U[x_{i-2},x_{i-1},x_i] \\ +b_{3,i}(x)\, U[x_{i-3},x_{i-2},x_{i-1},x_i] + ...+ \end{aligned} \tag{8}$$

for $x \le x_i$ where

$$\begin{aligned} b_{1,i}(x) = (x-x_i), \;\; b_{2,i}(x) = (x-x_i)(x-x_{i-1}), \\ b_{3,i}(x) = (x-x_i)(x-x_{i-1})(x-x_{i-2}) \end{aligned} \tag{9}$$

$$\begin{aligned} U_R^I(x) = U[x_i] + c_{1,i}(x)\, U[x_i,x_{i+1}] + c_{2,i}(x)\, U[x_i,x_{i+1},x_{i+2}] \\ +c_{3,i}(x)\, U[x_i,x_{i+1},x_{i+2},x_{i+3}] + ...+ \end{aligned} \tag{10}$$

for $x \ge x_i$ where

$$\begin{aligned} c_{1,i}(x) = (x-x_i), \;\; c_{2,i}(x) = (x-x_i)(x-x_{i+1}), \\ c_{3,i}(x) = (x-x_i)(x-x_{i+1})(x-x_{i+2}) \end{aligned} \tag{11}$$

Harten makes the observation [14] (see also Arandiga et al. [1], p.9) that the lack of smoothness may effect the quality of the approximation., For example in the case when $\frac{d^pU}{dx^p}$ has a jump discontinuity in the interval $[x_i,...,x_{i+m}]$ then one can prove that if the the kth divided differences has a jump discontinuity then

$$\begin{aligned} U[[x_i,...,x_{i+m}] &= O([U^{(p)}])/h^{m-p} \;\text{ if } m \;>\; p \\ &= O([U^{(m)}]) \quad \text{ otherwise} \end{aligned} \tag{12}$$

Which immediately suggests that there may be no advantage in using a polynomial of order higher than p. A similar situation occurs when a steep gradient is found close to areas of zero gradient and may then appear like a discontinuity. Consider the second divided difference

$$U[x_{i-1},x_i,x_{i+1}] = \frac{U[x_i,x_{i+1}] - U[x_{i-1},x_i]}{(x_{i+1}-x_{i-1})} \tag{13}$$

and suppose that $U[x_{i-1},x_i] = O(\varepsilon)U[x_i,x_{i+1}]$. It then follows that

$$U[x_{i-1},x_i,x_{i+1}] = (1-O(\varepsilon))\frac{U[x_i,x_{i+1}]}{(x_{i+1}-x_{i-1})} \tag{14}$$

At the top and bottom of discontinuities this has the implication that the highest order polynomial to be used in a steep gradient should have a stencil that is contained within the front. In the case when neighboring divided differences have different signs (e.g. let $U[x_{i-1},x_i] = -\lambda U[x_i,x_{i+1}]$ for $\lambda > 0$), it follows that

$$U[x_{i-1},x_i,x_{i+1}] = (1+\lambda))\frac{U[x_i,x_{i+1}]}{(x_{i+1}-x_{i-1})} \tag{15}$$

and the situation is similar to that of a discontinuity in the highest derivative. As an aside it is perhaps worth mentioning that the DASSL DAE code of Petzold [5] decreases order when it detects increasing higher-order differences.

2.2 Example of Polynomial Order Selection Procedure

The properties of this space are illustrated by the following example taken from Berzins [4] in which an advection problem has a solution which is both smooth and which has a steep profile is given by an 11th order polynomial which is defined in terms of the variable z, where

$$z = (0.3+t+ds*0.5-x)/ds; \tag{16}$$

In the case when $z > 1$ then $u(x,t) = 1.1$ while if $z < 0$ then $u(x,t) = 0.1$. For $0 \le z \le 1$ the value of $u(x,t) = p(z)$ where

$$p(z) = z^6\left[-252z^5+1386z^4-3080z^3+3465z^2-1980z+462\right] \tag{17}$$

and where z is defined by equation (16). The solution has a front of width ds centered about $0.3+t$. The numerical experiments conducted with this problem consisted of using spline approximations of degree 0 to 3 and also of using a h-p approximations in which the order was selected by the following procedure.

(i) construct linear quadratic and cubic spline interpolants based on a set of nodal values, $U(x_i)$.

(ii) construct first second and third divided difference approximations, $U[x_{i-1},x_{i+1}]$ $U[x_{i-1},x_i,x_{i+1}]$ and $U[x_{i-2},x_{i-1},x_i,x_{i+1}]$.

(iii) select the order of approximation on the sub-interval$[x_i,x_{i+1}]$ to be that for which there is no sign change in the highest divided difference used in the polynomial.

The errors in the different cases are as follows in the L1 norm.

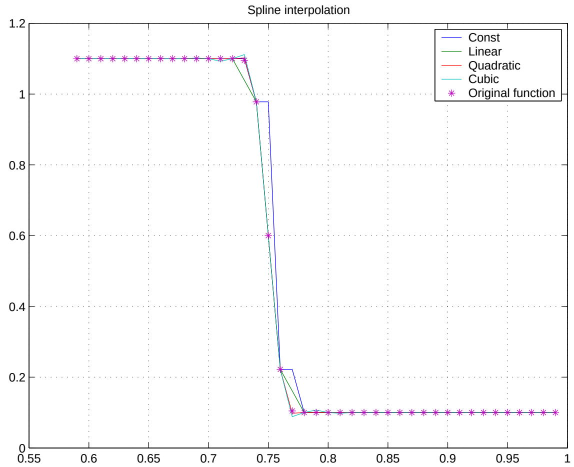

Fig. 1. Comparison of individual p version approximations

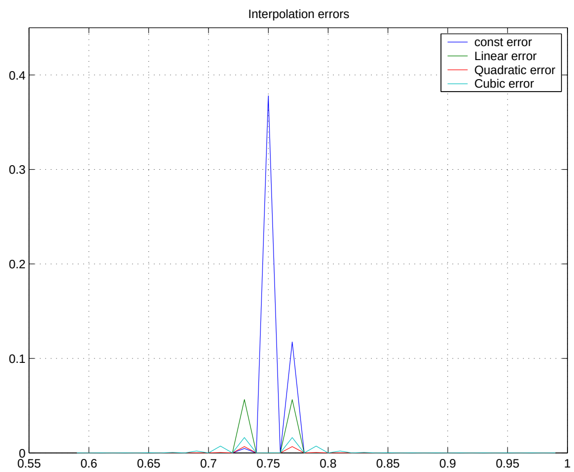

Fig. 2. Errors in individual p version approximations

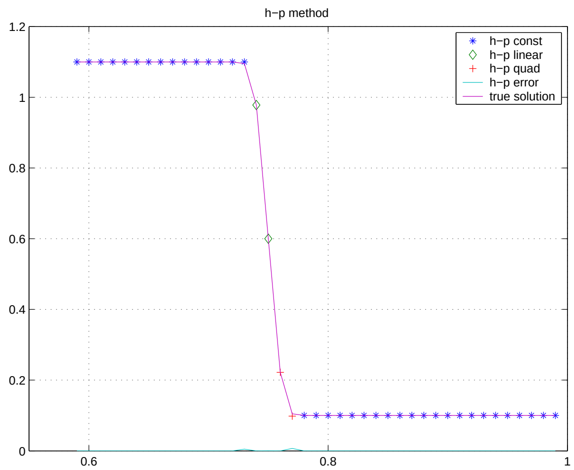

Fig. 3. New h-p solution and Errors

Method	dx	0	1	2	3	h-p
L1 Error	0.08	1.8e-2	6.0e-2	9.5e-2	1.1e-1	6.0e-2
L1 Error	0.04	2.0e-2	1.7e-2	1.8e-2	2.1e-2	1.4e-2
L1 Error	0.02	5.0e-3	1.1e-3	1.4e-4	5.2e-4	1.3e-4

Table 1
Comparison of Spline Error Norms on Travelling Front Example

The key feature of these results is that high order polynomials are used inside the steep gradient. This is demonstrated by the results shown in Figures 1 to 3 which show the individual *p*th order approximations, their errors and the h-p approximation. Table 1 shows that the h-p approximation outperforms the fixed p approximations in a way that is consistent with many other h-p results, e.g. Schwab [21].

3 Data Bounded Polynomial Interpolants

The example above is illuminating and while it suggests a way of changing order it does not help directly to reduce oscillations. One possible step in developing positive finite element schemes is to define bounded polynomial interpolants which may be based on ratios of divided differences such as, for example,

$$r_{[i-1,i]}^{[i,i+1]} = \frac{U[x_i, x_{i+1}]}{U[x_{i-1}, x_i]} \tag{18}$$

with obvious extensions to higher differences and other indices.

In order to construct bounded interpolants the function $\Phi(r)$ is defined by $\Phi(r) = max(0, min(r, R))$ where R is a known maximum value yet to be defined, see [4]. This function is applied to divided difference terms to define bounded divided differences denoted by $[....]^B$.

$$U[x_i, x_{i+1}]^B = \Phi(r_{[i-1,i]}^{[i,i+1]})U[x_{i-1}, x_i] \tag{19}$$

which may be written as

$$U[x_i, x_{i+1}]^B = \hat{R}_{[i-1,i]}^{[i,i+1]})U[x_{i-1}, x_i] \ . \tag{20}$$

where $\hat{R}_{[i-1,i]}^{[i,i+1]}) = \Phi(r_{[i-1,i]}^{[i,i+1]})$. It should be noted that $U[x_i, x_{i+1}]^B = r_{[i-1,i]}^{[i,i+1]} U[x_{i-1}, x_i]$ if $0 \le r_{[i-1,i]}^{[i,i+1]}) \le R$. Furthermore let

$$U[x_{i-1}, x_i]^B = \hat{S}_{[i-2,i-1]}^{[i-1,i]} U[x_{i-1}, x_i] \tag{21}$$

where $\hat{S}^{[i-1,i]}_{[i-2,i-1]} = \frac{\Phi(r^{[i-1,i]}_{[i-2,i-1]})}{r^{[i-1,i]}_{[i-2,i-1]}}$ is a filter that has value one if $0 \le r^{[i-1,i]}_{[i-2,i-1]} \le 1$ and has value zero otherwise.

A number of useful illustrations of this are:

$$U[x_{i-1},x_i,x_{i+1}]^B = \frac{\Phi(r^{[i,i+1]}_{[i-1,i]})U[x_{i-1},x_i] - \Phi(r^{[i-1,i]}_{[i-2,i-1]})U[x_{i-2},x_{i-1}]}{x_{i+1}-x_{i-1}} \tag{22}$$

and

$$U[x_{i-1},x_i,x_{i+1}]^B = \frac{\left(\Phi(r^{[i,i+1]}_{[i-1,i]}) - \frac{\Phi(r^{[i-1,i]}_{[i-2,i-1]})}{r^{[i-1,i]}_{[i-2,i-1]}}\right)}{(x_{i+1}-x_{i-1})}\, U[x_{i-1},x_i] \tag{23}$$

which may be written as

$$U[x_{i-1},x_i,x_{i+1}]^B = \frac{S^{[i-i,i]}_{[i-1,i,i+1]}}{(x_{i+1}-x_{i-1})}U[x_{i-1},x_i] \tag{24}$$

where

$$S^{[i-1,i]}_{[i-1,i,i+1]} = \left(\Phi(r^{[i,i+1]}_{[i-1,i]}) - \frac{\Phi(r^{[i-1,i]}_{[i-2,i-1]})}{r^{[i-1,i]}_{[i-2,i-1]}}\right). \tag{25}$$

Furthermore the $\hat{S}^{[\dots]}_{[\dots]}$ operator will be used in connection with mass matrices and positivity

$$U[x_{i-1},x_i,x_{i+1}]^M = \hat{S}^{[i,i+1]}_{[i-1,i]}\, U[x_{i-1},x_i,x_{i+1}] \tag{26}$$

where $\hat{S}^{[i,i+1]}_{[i-1,i]}$ is defined as in equation (21) and which, when substituted, gives

$$U[x_{i-1},x_i,x_{i+1}]^M = \frac{\Phi_M(r^{[i,i+1]}_{[i-1,i]})}{r^{[i,i+1]}_{[i-1,i]}} \frac{\left(1-\frac{1}{r^{[i,i+1]}_{[i-1,i]}}\right)}{(x_{i+1}-x_{i-1})}\, U[x_i,x_{i+1}] \tag{27}$$

where $\Phi_M(r) = r$ if $0 \leq r \leq 1$ and is zero otherwise. The extensions to higher divided differences follow in a similar way:

$$U[x_{i-1},x_i,x_{i+1},x_{i+2}]^B = \frac{(\Phi(r^{[i-1,i,i+1]}_{[i,i+1,i+2]} - \frac{\Phi(r^{[i-1,i,i+1]}_{[i-2,i-1,i]})}{r^{[i-1,i,i+1]}_{[i-2,i-1,i]}})}{(x_{i+2}-x_{i-1})} U[x_{i-1},x_i,x_{i+1}]^B \quad (28)$$

which may be written as

$$U[x_{i-1},x_i,x_{i+1},x_{i+2}]^B = \frac{S^{[i-1,i,i+1]}_{[i-1,i,i+1,i+2]}}{(x_{i+2}-x_{i-1})} U[x_{i-1},x_i,x_{i+1}]^B. \quad (29)$$

The bounded form of the function $\Phi(r)$ makes it straightforward to bound the functions $S^{[\dots]}_{[\dots]}$ by

$$-1 \leq S^{[\dots]}_{[\dots]} \leq R. \quad (30)$$

Consider the case of polynomials of degree 2 given by equations (8) and (9)

$$U^I_L(x) = U[i] + \left[b_{1,i}(x) + \frac{b_{2,i}(x)}{(x_i - x_{i-2})} S^{[i-1,i]}_{[i-2,i-1,i]} \right] U[x_{i-1},x_i]. \quad (31)$$

The condition for this polynomial to be data bounded in that it satisfies
$U[i-1] \leq U^I(x) \leq U[i]$, is

$$0 \leq \frac{\left[b_{1,i}(x) + \frac{b_{2,i}(x)}{(x_i - x_{i-2})} S^{[i-1,i]}_{[i-2,i-1,i]} \right]}{x_i - x_{i-1}} \leq 1\ . \quad (32)$$

Let $x = x_{i-1} + \lambda\ h_i$, $0 \leq \lambda \leq 1$, $h = x_i - x_{i-1}$ and $\mu h = x_{i-1} - x_{i-2}$. Substituting in the previous equation then gives

$$0 \leq (1-\lambda) \left(1 + \frac{\lambda}{1+\mu} S^{[i-1,i]}_{[i-2,i-1,i]} \right) \leq 1\ . \quad (33)$$

Routine consideration of values of λ, μ gives

$$-1 \leq S^{[i-1,i]}_{[i-2,i-1,i]} \leq 1\ . \quad (34)$$

which corresponds to equation (30) with $R = 1$. The extension to higher order polynomials would seem to require further restrictions on the higher-order divided differences to maintain data-boundedness. In the context of time-varying solutions the addition of (t_n) will signify the time at which solution values are used when evaluating the divided difference expressions.

3.1 *Relationship to Polynomial Filtering*

The application of the successive multipliers such as $S^{[i,i+1]}_{[i-1,i,i+1]}$ to the divided difference terms in a recursive way may be immediately interpreted as a polynomial filtering method. The idea behind the polynomial filtering methods of Gottlieb and Shu [10] is to modify the polynomial coefficients so as to improve accuracy in the presence of discontinuities. The application of this idea to hyperbolic equations when spectral methods are used is described by Gottlieb and Hesthaven [11].

In teh context of this work, for example, the polynomial defined by equation (9) may be modified by replacing the coefficients $b_{j,i}$ to get modified coefficients $\bar{b}_{j,i}$. The form of the modified coefficient may be seen by combining equations (8),(22) and (28) to define the modified polynomial by

$$U^{I,B}(x) = U[x_i] + b_{1,i}(x)\, U[x_{i-1},x_i] + b_{2,i}(x)\, U[x_{i-2},x_{i-1},x_i]^B + b_{3,i}(x)\, U[x_{i-3},x_{i-2},x_{i-1},x_i]^B \tag{35}$$

for $x > x_i$. From this it is straightforward to write this polynomial as

$$U^{I,B}(x) = U[x_i] + \bar{b}_{1,i}(x)\, U[x_{i-1},x_i] + \bar{b}_{2,i}(x)\, U[x_{i-2},x_{i-1},x_i] + \bar{b}_{3,i}(x)\, U[x_{i-3},x_{i-2},x_{i-1},x_i] \tag{36}$$

where $\bar{b}_{1,i}(x) = b_{1,i}(x)U[x_{i-1},x_i]^B/U[x_{i-1},x_i]$, $\bar{b}_{2,i}(x) = b_{2,i}(x)\, U[x_{i-2},x_{i-1},x_i]^B/U[x_{i-2},x_{i-1},x_i]$ and $\bar{b}_{3,i}(x) = b_{3,i}(x)\, U[x_{i-3},x_{i-2},x_{i-1},x_i]^B/U[x_{i-3},x_{i-2},x_{i-1},x_i]$. It is is also worth remarking that the filter defined here is nonlinear as the modifying coefficients depend on the solution and on approximations to its derivatives.

4 Overview of Positive Finite Element Method

An overview of the new positive methods discussed below is given as follows. Consider a pth order basis consisting of functions $\Phi_{i,p}(x)$ and suppose, for ease of exposition, that the number of basis functions and their spatial position does not change as the order changes. Let the pth order polynomial approximation based on a set of nodal solution values be given by

$$U_p(x,t) = \sum_{i=1}^{N} \phi_{i,p}(x) U_i(t) \tag{37}$$

where the nodal values are given by $U_i(t)$ and where the inflow values are specified by a Dirchlet condition. The standard Galerkin approach then gives:

$$\int_{x_L}^{x_R} \frac{\partial U_p}{\partial t} \phi_{i,p}(x)\, dx = \int_{x_L}^{x_R} -\frac{\partial U_p}{\partial x} \phi_{i,p}(x)\, dx, \quad i = 1, \ldots, N, \tag{38}$$

Suppose that we discretize in time using the forward Euler method or a positive Runge - Kutta method such as those of Shu and Osher [20] which involve stages similar in form to forward Euler. Equation (38) may then be written as:

$$\int_{x_L}^{x_R} (U_p(x,t_{n+1}) - U_p(x,t_n)) \phi_{i,p}(x)\, dx = -\delta\, t \int_{x_L}^{x_R} \frac{\partial U_p(x)}{\partial x}(x,t_n) \phi_{i,p}(x)\, dx \\ , i = 1, \ldots, N \tag{39}$$

where $t_{n+1} = t_n + \delta t$. Define the residual $R_p(U_p, x, t)$ by

$$R_p(U_p, x, t_n) = (U_p(x,t_{n+1}) - U_p(x,t_n)) + \delta t \frac{\partial U_p}{\partial x} \tag{40}$$

and rewrite equation (39) as

$$(R_p(U_p, x, t_n), \phi_{i,p}) = 0, \quad i = 1, \ldots, N \tag{41}$$

In the simplest case the piecewise constant (DG) method for the advection equation with positive velocity equal to one is given by

$$(R_0(U_0, x, t_n), \phi_{i,0}) = (U_0(x_i, t_n + \delta t) - U_0(x_i, t_n)) \\ + \frac{\delta t}{\delta x}(U_0(x_i, t_n) - U_0(x_{i-1}, t_n)) \tag{42}$$

where δx is the uniform mesh spacing. The higher order positive schemes considered here (and by Berzins [4]) will mostly have the form:

$$a_{0,p}U_i^{n+1} + a_{1,p}(U_i^{n+1} - U_{i+1}^{n+1}) = a_{0,p}U_i^n - a_{3,p}(U_i^n - U_{i-1}^n) - a_{2,p}(U_i^n - U_{i-1}^n) \tag{43}$$

It will however also be necessary to consider the alternative form given by

$$a_{0,p}U_i^{n+1} + \bar{a}_{1,p}(U_i^{n+1} - U_{i-1}^{n+1}) = a_{0,p}U_i^n - \bar{a}_{3,p}(U_i^n - U_{i+1}^n) - a_{2,p}(U_i^n - U_{i-1}^n) \tag{44}$$

in the case of the equations at the edges of quadratic elements. In order to satisfy the positivity condition given by equation (3) the coefficients satisfy the equations, [4];

$$0 \le a_{1,p}/a_{0,p} \le 1 \tag{45}$$

$$0 \le (a_{2,p} + a_{3,p})/a_{0,p} \le 1 \tag{46}$$

for $i = 1,2,3,\ p = 0,1,2$. (N.B. In the form considered by Berzins [4] all the equations were scaled beforehand by dividing through by $a_{0,p}$). In the case of equation (44) $a_{1,p}$ and $a_{3,p}$ are replaced with $\bar{a}_{1,p}$ and $\bar{a}_{3,p}$ respectively. In the case of the piecewise constant method defined by equation (42) with forward Euler time integration employed, the coefficients are $a_{0,0} = 1$, $a_{1,0} = a_{2,0} = 0$ and $a_{3,0} = \frac{\delta t}{\delta x}$. For higher order basis functions the coefficients $a_{j,p}$ are defined by:

$$a_{0,p} = \int_{x_L}^{x_R} \phi_{i,p}(x)\ dx \tag{47}$$

$$a_{1,p} = \frac{1}{U_i^{n+1} - U_{i+1}^{n+1}} \int_{x_L}^{x_R} (U_p^M(x,t_{n+1}) - U_p^M(x_i,t_{n+1}))\phi_{i,p}(x)\ dx \tag{48}$$

$$a_{3,p} = \frac{1}{U_i^n - U_{i-1}^n} \int_{x_L}^{x_R} (U_p^m(x,t_n) - U_p^m(x_i,t_n))\phi_{i,p}(x)\ dx \tag{49}$$

$$a_{2,p} = \frac{\delta t}{U_i^n - U_{i-1}^n} \int_{x_L}^{x_R} \frac{\partial U_p^B(x,t_n)}{\partial x} \phi_{i,p}(x)\ dx \tag{50}$$

$$i = 1, \ldots, N.$$

In the case of the method written in the form defined by equation (44)

$$\bar{a}_{1,p} = \frac{1}{U_i^{n+1} - U_{i-1}^{n+1}} \int_{x_L}^{x_R} (U_p^M(x,t_{n+1}) - U_p^M(x_i,t_{n+1}))\phi_{i,p}(x)\ dx \tag{51}$$

$$\bar{a}_{3,p} = \frac{1}{U_i^n - U_{i+1}^n} \int_{x_L}^{x_R} (U_p^m(x,t_n) - U_p^m(x_i,t_n))\phi_{i,p}(x)\, dx \qquad (52)$$

$$i = 1, ..., N.$$

The general approach that will be used is that polynomials defined in Sections 2 and 3 will be used to define the polynomials $U^M(x,t), U^m(x,t)$ and $U^B(x,t)$ in the integrands. The method defined by equation (43) is implicit so Jacobi iteration will be used to solve for the new solution values. In the case of iteration m the equations can then be written, after dividing through by the quantity $(a_{0,p} + a_{1,p})$, as:

$$U_i^{n+1,m+1} = \frac{a_{1,p}U_{i+1}^{n+1,m} + U_i^n(a_{0,p} - (a_{2,p} + a_{3,p})) + (a_{2,p} + a_{3,p})U_{i-1}^n}{(a_{0,p} + a_{1,p})} \qquad (53)$$

to get an iteration which is also positive if the positivity conditions (45) and (46) are satisfied.The predicted values of $U_i^{n+1,m+1}$ are given by the explicit methods defined by equation (43) with $a_{1,p} = a_{3,p} = 0$.

In the case of the method written as in equation (44) a similar iteration is given by:

$$U_i^{n+1,m+1} = \frac{\bar{a}_{1,p}U_{i-1}^{n+1,m} + U_i^n(a_{0,p} - (a_{2,p} + \bar{a}_{3,p})) + a_{2,p}U_{i-1}^n + \bar{a}_{3,p}U_{i+1}^n}{(a_{0,p} + \bar{a}_{1,p})} \qquad (54)$$

Upon using equations (45) and (46) in (53) it can be seen that these conditions ensure that the value of $U_i^{n+1,m+1}$ is positive. Equation (54) yields a similar result. It should also be noted that the directionality in the formula defined by equation (43) reflects the directionality of the underlying advection equation. The extension to negative velocity is relatively straightforward, [4].

4.1 Quadratic Basis Function Method

For ease of exposition consider the case of quadratic approximations. Berzins [4] considered the case of quadratic splines whereas here we consider the case of standard C^0 continuous Galerkin Methods, [12]. There are two kinds of situations to consider in the Galerkin orthogonality relationships in equation (41). In the case when the test function is at at the element edge, as shown by ϕ_i^e in Figure 4, the the underlying polynomial approximation is only C^0 continuous at the edge of an element. The second case is that of a test function interior to and non-zero only in the element as shown by ϕ_{i+1}^m in Figure 4.

The first step is to define the form of the solution in the cases when the test function is at the edge of an element and when it is the interior to an element. In the first case

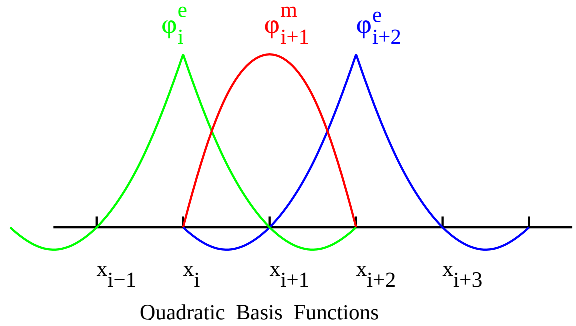

Fig. 4. Comparison of individual p version approximations

it is necessary to consider the divided difference form of the solution spanning both sides of the element edge and then to replace the divided difference polynomials by bounded approximations. The approximating polynomial is only continuous at the node x_i and so has the form:

$$U_p(x,t_n) = U[x_i]^n + b_{1,i}(x)\; U[x_{i-1},x_i]^n + b_{2,i}(x)\; U[x_{i-2},x_{i-1},x_i]^n \tag{55}$$

for $x \leq x_i$ where $b_{1,i}(x)$ and $b_{2,i}(x)$ are defined by equation (9). The subscript ^{n} indicates that the divided difference expression is evaluated at t_n.

$$U_p(x,t_n) = U[x_i]^n + c_{1,i}(x)\; U[x_i,x_{i+1}]^n + c_{2,i}(x)\; U[x_i,x_{i+1},x_{i+2}]^n \tag{56}$$

for $x \geq x_i$ where $c_{1,i}(x)$ and $c_{2,i}(x)$ are defined by equation (11). The form of the solution in the domain in which the midpoint test function polynomial is non-zero and has value one at node x_i is given by

$$U_p(x,t_n) = U[x_i]^n + d_{1,i}(x)\; U[x_{i-1},x_i]^n + d_{2,i}(x)\; U[x_{i-1},x_i,x_{i+1}]^n. \tag{57}$$

In the case when linear basis functions are used $b_{2,1} = c_{2,1} = 0$ and as every node is the edge of an element equation (57) is not used.

4.1.1 *Positive form of Derivative Terms*

In defining the coefficient $a_{2,p}$ the new forms of the two polynomials defined by equations (55) and (56) are derived by modifying the polynomials so that their derivatives contain the factor $(U_i^n - U_{i-1}^n)$ in a similar way to equation (31) and are given by

$$U_p^B(x,t_n) = U[x_i]^n + \left(b_{1,i}(x)\hat{S}^{[i-1,i]}_{[i-2,i-1]}(t_n) + b_{2,i}(x)\frac{S^{[i-1,i]}_{[i-1,i,i+1]}(t_n) \times \hat{R}^{[i-1,i,i+1]}_{[i-2,i-1,i]}(t_n)}{(x_{i+1}-x_{i-1})} \right) U[x_{i-1},x_i]^n \tag{58}$$

for $x \leq x_i$ and

$$U_p^B(x,t_n) = U[x_i]^n + \left(c_{1,i}(x)\hat{R}^{[i,i+1]}_{[i-1,i]}(t_n) + c_{2,i}(x)\frac{S^{[i,i+1]}_{[i-1,i,i+1]}(t_n) \times \hat{R}^{[i-1,i,i+1]}_{[i,i+1,i+2]}(t_n)}{(x_{i+1} - x_{i-1})} \right) U[x_{i-1}, x_i]^n \tag{59}$$

for $x \geq x_i$ and where $\hat{R}, \hat{S}, S$ and r are defined in Section 3.

The modified form of the solution in the domain in which the solution polynomial is given by equation (57) is given by

$$U_p^B(x,t_n) = U[x_i] + \left(d_{1,i}(x)\hat{S}^{[i-1,i]}_{[i-2,i-1]} + d_{2,i}(x)\frac{S^{[i-1,i]}_{[i-1,i,i+1]}(t_n)}{(x_{i+1} - x_{i-1})} \right) U[x_{i-1}, x_i]^n \tag{60}$$

In each case differentiating these modified polynomials gives an expression of the form

$$\frac{\partial U_p(x,t_n)}{\partial x} = w(x)\ U[x_{i-1}, x_i]^n \tag{61}$$

where the generic polynomial $w(x)$ is defined by differentiating the right side of equations (58), (59) or (60) as required. Hence the generic form of the coefficient $a_{2,p}$ is

$$a_{2,p} = \delta t \int_{x_L}^{x_R} w(x)\phi_{i,p}(x)\ dx. \tag{62}$$

4.1.2 Positive form of Mass Matrix Terms

In order to obtain an expression for $a_{1,p}$ the same series expansions as in the previous sub-section are used. The forms of the modified polynomials specific to the mass matrix are different however and are required by equation (43) to contain the factor $(U_i^{n+1} - U_{i-1}^{n+1})$ in a similar way to that of the previous sub-section and are given by

$$U_p^M(x,t_{n+1}) = U[x_i]^{n+1} + \hat{S}^{[i,i+1]}_{[i-i,i]}(t_n) \times \left(b_{1,i}(x) + b_{2,i}(x)\frac{\hat{R}^{[i-2,i-1,i]}_{[i-1,i,i+1]}(t_{n+1}) \times (r^{[i,i+1]}_{[i-1,i]}(t_{n+1}) - 1)}{(x_{i+1} - x_{i-1})} \right) U[x_{i-1}, x_i]^{n+1} \tag{63}$$

for $x \le x_i$ and

$$U_p^M(x,t_{n+1}) = U[x_i]^{n+1} + \hat{S}_{[i-i,i]}^{[i,i+1]}(t_{n+1})U[x_{i-1},x_i]^{n+1} \times \left(c_{1,i}(x) r_{[i-1,i]}^{[i,i+1]})(t_{n+1}) + c_{2,i}(x) \frac{\hat{R}_{[i-1,i,i+1]}^{[i,i+1,i+2]}(t_{n+1}) \times (r_{[i-1,i]}^{[i,i+1]}(t_{n+1}) - 1)}{(x_{i+1} - x_{i-1})} \right) \tag{64}$$

for $x \ge x_i$. The midpoint polynomial modified form is given by

$$U_p^M(x,t_{n+1}) = U[x_i]^{n+1} + \hat{S}_{[i-i,i]}^{[i,i+1]}(t_{n+1})U[x_i,x_{i+1}]^{n+1} \times \left(d_{1,i}(x) r_{[i,i+1]}^{[i-1,i]}(t_{n+1}) \; + d_{2,i}(x) \; \frac{(1 - r_{[i,i+1]}^{[i-1,i]}(t_{n+1}))}{(x_{i+1} - x_{i-1})} \right) \tag{65}$$

Thus in the case of equations (63) and (64) the generic form of the polynomial used here is thus given by:

$$U_p^M(x,t_{n+1}) = U(x_i,t_{n+1}) + w(x) \; (U_i^{n+1} - U_{i-1}^{n+1}) \tag{66}$$

where $w(x)$ is a polynomial whose precise form is given by equations (63) and (64).

$$\bar{a}_{1,p} = \int_{x_L}^{x_R} w(x) \phi_{i,p}(x) \; dx. \tag{67}$$

In the case of equation (65) the generic form of the polynomial used here is thus given by:

$$U_p^M(x,t_{n+1}) = U(x_i,t_{n+1}) + w(x) \; (U_{i+1}^{n+1} - U_i^{n+1}) \tag{68}$$

where $w(x)$ is a polynomial whose precise form is given by equations (57) and (65). Hence

$$a_{1,p} = \int_{x_L}^{x_R} w(x) \phi_{i,p}(x) \; dx. \tag{69}$$

and In the final case the situation is complicated because of the need to constrain polynomials in a way that is consistent across the time step. This is done by using the function $\hat{S}_{[i-i,i]}^{[i,i+1]}(t_{n+1})$ in the mass matrix polynomials at both t_n and t_{n+1}. The modified forms of the mass matrix polynomials used at the edges of elements are given by

$$U_p^m(x,t_n) = U[x_i]^n + \hat{S}_{[i-i,i]}^{[i,i+1]}(t_{n+1}) \times \left(b_{1,i}(x) r_{[i,i+1]}^{[i-1,i]}(t_n) + b_{2,i}(x) \frac{\hat{R}_{[i-1,i,i+1]}^{[i-2,i-1,i]}(t_n) \times (1 - r_{[i,i+1]}^{[i-1,i]}(t_n))}{(x_{i+1} - x_{i-1})} \right) U[x_i, x_{i+1}]^n \tag{70}$$

for $x \leq x_i$

$$U_p^m(x,t_n) = U[x_i]^n + \hat{S}_{[i-i,i]}^{[i,i+1]}(t_{n+1}) \times \left(c_{1,i}(x) + c_{2,i}(x) \frac{\hat{R}_{[i-1,i,i+1]}^{[i,i+1,i+2]}(t_n) \times (1 - r_{[i,i+1]}^{[i-1,i]}(t_n))}{(x_{i+1} - x_{i-1})} \right) U[x_i, x_{i+1}]^n \tag{71}$$

for $x \geq x_i$.

The midpoint polynomial for an element in modified form is given by

$$U_p^m(x,t_n) = U[x_i]^n + \hat{S}_{[i-i,i]}^{[i,i+1]}(t_{n+1}) \times \left(d_{1,i}(x) \; + d_{2,i}(x) \; \frac{(r_{[i-1,i]}^{[i,i+1]}(t_n) - 1)}{(x_{i+1} - x_{i-1})} \right) U[x_{i-1}, x_i]^n. \tag{72}$$

At the edges of elements the form of the polynomial that results may be written as:

$$U_p^m(x,t_n) = U(x_i,t_n) + w(x) \; (U_{i+1}^n - U_i^n) \tag{73}$$

where $w(x)$ is a polynomial whose precise form is given by equations (70) and (71) respectively. Hence

$$\bar{a}_{3,p} = \int_{x_L}^{x_R} w(x) \phi_{i,p}(x) \; dx. \tag{74}$$

and in the case of equation (72)

$$U_p^m(x,t_n) = U(x_i,t_n) + w(x) \; (U_i^n - U_{i-1}^n) \tag{75}$$

where $w(x)$ is a polynomial whose precise form is given by equation (72). Hence

$$a_{3,p} = \int_{x_L}^{x_R} w(x)\phi_{i,p}(x)\, dx. \tag{76}$$

4.2 Galerkin Orthogonality

The definitions of the three polynomials $U_p^B(x,t), U_p^M(x,t)$ and $U_p^m(x,t)$ given above now make it possible to define the residual function $R_p^*(U_p,x,t)$ associated with the new method as

$$R_p^*(U_p,x,t_n) = (U_p^M(x,t_{n+1}) - U_p^m(x,t_n)) + \delta t \frac{\partial U_p^B}{\partial x} \tag{77}$$

and to note that the method is defined by ensuring that the Galerkin orthogonality condition

$$(R_p^*(U_p,x,t_n), \phi_{i,p}) = 0, \quad i = 1,...,N \tag{78}$$

is satisfied. This makes it possible to describe the method as a nonlinear and variable-order Petrov-Galerkin method.

5 Positive Linear/Constant Finite Element Method

The standard linear finite element discretization of the advection equation on a uniform mesh (the case of a non-uniform mesh is considered by Gresho and Sani [12]) is given by:

$$\frac{1}{6}\left[\dot{U}_{i-1} + 4\dot{U}_i + \dot{U}_{i+1}\right] = \frac{-1}{2\delta x}(U_{i+1} - U_{i-1}) \tag{79}$$

where δx is the uniform mesh spacing in this case and where $\dot{U}_i = \frac{dU_i}{dt}$. The residual, as defined by equation (41) with $p = 1$, may be written as

$$U_i^{n+1} + \frac{1}{6}\delta^2 U_i^{n+1} = U_i^n + \frac{1}{6}\delta^2 U_i^n - \frac{\delta t}{\delta x}(U_i^n - U_{i-1}^n)x - \frac{\delta t}{2\delta x}\delta^2 U_i^n \tag{80}$$

where U_k^n denotes the value at mesh point k at time t_n and where δ^2 is defined by $\delta^2 U_i^n = \left[U_{i-1}^n - 2U_i^n + U_{i+1}^n\right]$ The processes of Sections 4.1.1. and 4.1.2 then give rise to three different approximations to the δ^2 terms in this equation and to the expressions:

$$a_{0,1} = \delta x \tag{81}$$

$$a_{1,1} = \frac{(r^{[i-1,i]}_{[i,i+1]}(t_{n+1}) - 1)}{6} \hat{S}^{i,i+1}_{i-1,i}(t_{n+1}) \delta x \tag{82}$$

$$a_{3,1} = \frac{(r^{[i,i+1]}_{[i-1,i]}(t_n) - 1)}{6} \hat{S}^{i,i+1}_{i-1,i}(t_{n+1}) \delta x \tag{83}$$

$$a_{2,1} = \frac{\delta t}{\delta x} \left[1 + \frac{S^{[i-1,i]}_{[i-1,i,i+1]}(t_n)}{2} \right] \delta x. \tag{84}$$

The positivity conditions as given by Berzins [3] are then

$$0 \leq r^{[i,i+1]}_{[i-1,i]}(t_{n+1}), r^{[i,i+1]}_{[i-1,i]}(t_n) \leq 1 \text{ and } 0 \leq \frac{\delta t}{\delta x}(1 + \frac{S^{[i-1,i]}_{[i-1,i,i+1]}(t_n)}{2})) \leq \frac{5}{6}. \tag{85}$$

If these conditions do not hold the piecewise constant method defined by equation (42) is used instead.

6 Positive Quadratic finite Element Method

Using the standard quadratic finite element method, again on a regular mesh, with its differing treatment of edge and interior nodes, e.g. see Gresho and Sani [12], gives rise to the o.d.e.s at element edges defined by:

$$\frac{-\dot{U}_{i-2} + 2\dot{U}_{i-1} + 8\dot{U}_i + 2\dot{U}_{i+1} - \dot{U}_{i+2}}{10} = \frac{-4(U_{i+1} - U_{i-1}) + (U_{i+2} - U_{i-2})}{4\delta x}. \tag{86}$$

at interior nodes

$$\frac{\dot{U}_{i-1} + 8\dot{U}_i + \dot{U}_{i+1}}{10} = \frac{-(U_{i+1} - U_{i-1})}{2\delta x}. \tag{87}$$

In order to understand the way that these methods are modified by using the polynomials of Section 4 apply the Forward Euler method as in equation (40) and a some manipulation to rewrite the method in the case of edge nodes as

$$U_i^{n+1} - (\frac{(s_i^{n+1} + \frac{1}{s_{i-1}^{n+1}})}{10})\delta^2 U_i^{n+1} = U_i^n - (\frac{(s_i^n + \frac{1}{s_{i-1}^n})}{10})\delta^2 U_i^n$$
$$- \frac{\delta t}{\delta x}(U_i^n - U_{i-1}^n) - \frac{\delta t}{2\delta x}\,(1 + \frac{1}{2}(s_i^n - \frac{1}{s_{i-1}^n}))\,\delta^2 U_i^n \tag{88}$$

and in the case of nodes in the interior of elements as

$$U_i^{n+1} + (\frac{1}{6} - \frac{1}{15})\delta^2 U_i^{n+1} = U_i^n + (\frac{1}{6} - \frac{1}{15})\delta^2 U_i^n$$
$$- \frac{\delta t}{\delta x}(U_i^n - U_{i-1}^n) - \frac{\delta t}{2\delta x}\,\delta^2 U_i^n \tag{89}$$

where the second derivative ratios at time levels such as t^{n+1} are given by

$$s_i^{n+1} = r_{[i-1,i,i+1]}^{[i,i+1,i+2]}(t_{n+1}) \text{ and } s_{i-1}^{n+1} = r_{[i-2,i-1,i]}^{[i-1,i,i+1]}(t_{n+1})$$

and the ratios at time level n denoted by s_i^n and s_{i-1}^n are similarly defined. The positive quadratic method is defined by applying the function $\Phi()$ to terms as s_*^* and by using the same functions as were applied in the linear case to the δ^2 terms. Define the functions

$$S_+^{i,n} = \left[\hat{R}_{[i-1,i,i+1]}^{[i,i+1,i+2]}(t_n) + \hat{R}_{[i-1,i,i+1]}^{[i-2,i-1,i]}(t_n)\right], \tag{90}$$
$$S_-^{i,n} = \left[\hat{R}_{[i-1,i,i+1]}^{[i,i+1,i+2]}(t_n) - \hat{R}_{[i-1,i,i+1]}^{[i-2,i-1,i]}(t_n)\right], \tag{91}$$

where the function $\hat{R}(.)$ is defined as in equation (20). The quantities $S_\pm^{i,n+1}$ are similarly defined at time t_{n+1}. Using these quantities and the form of the polynomials defined in Section 4 equations (88) may be written as

$$U_i^{n+1} - \frac{S_+^{i,n+1}}{10}\hat{S}_{[i-1,i]}^{[i,i+1]}(t_{n+1})(r_{i-1,i}^{i,i+1}(t_{n+1}) - 1)(U_i^{n+1} - U_{i-1}^{n+1})$$
$$= U_i^n - \frac{S_+^{i,n}}{10}\hat{S}_{[i-1,i]}^{[i,i+1]}(t_{n+1})(1 - r_{i,i+1}^{i-1,i}(t_n))(U_{i+1}^n - U_i^n)$$
$$- \frac{\delta t}{\delta x}(U_i^n - U_{i-1}^n) - \frac{\delta t}{2\delta x}\,(1 + \frac{1}{2}(S_-^{i,n}))S_{[i-1,i,i+1]}^{[i-1,i]}(t_n)(U_i^n - U_{i-1}^n) \tag{92}$$

In the case of nodes at the edges of elements the coefficients defined in Section 2 are given by

$$a_{0,2} = a_{0,1}\frac{2}{3} \tag{93}$$

$$\bar{a}_{1,2} = a_{3,1}\frac{6}{10}S_{+}^{i,n+1}a_{0,2} \tag{94}$$

$$\bar{a}_{3,2} = \frac{a_{1,1}}{a_{0,1}}\frac{6}{10}S_{+}^{i,n}a_{0,2} \tag{95}$$

$$a_{2,2} = (\frac{a_{2,1}}{a_{0,1}} + (\frac{a_{2,1}}{a_{0,1}} - 1)\frac{1}{2}(S_{-}^{i,n})))a_{0,2} \tag{96}$$

In the case of the interior nodes the only difference between the linear and quadratic cases lies in the scaling of the δ^2 terms and so the coefficients are then very similar to those defined by equations (81) to (89) and are given by:

$$a_{0,2} = \delta x \frac{2}{3} \tag{97}$$

$$a_{1,2} = \frac{(r_{[i,i+1]}^{[i-1,i]}(t_{n+1}) - 1)}{10}\hat{S}_{i-1,i}^{i,i+1}(t_{n+1})a_{0,2} \tag{98}$$

$$a_{3,2} = \frac{(r_{[i-1,i]}^{[i,i+1]}(t_n) - 1)}{6}\hat{S}_{i-1,i}^{i,i+1}(t_{n+1})a_{0,2} \tag{99}$$

$$a_{2,2} = \frac{\delta t}{\delta x}\left[1 + \frac{S_{[i-1,i,i+1]}^{[i-1,i]}(t_n)}{2}\right]a_{0,2} \tag{100}$$

The definition of these coefficients then makes it possible to determine the conditions for positivity by substituting from equations (94) and (95) into equation (46) to get:

$$0 \le \frac{S_{+}^{i,n}}{10}\hat{S}_{[i-1,i]}^{[i,i+1]}(t_{n+1})(r_{[i,i+1]}^{[i-1,i]}(t_n) - 1) + \frac{\delta t}{2\delta x}\left[2 + (1 + \frac{S_{-}^{i,n}}{2})S_{[i-1,i,i+1]}^{[i-1,i]}(t_n)\right] \le 1 \tag{101}$$

where $0 \le r_{[i-1,i]}^{[i,i+1]}(t_n) \le 1$. The stability condition thus involves ratios of solution values at both t_n and t_{n+1}. The condition for the stability of the predictor (i.e. no mass matrix) is somewhat simpler in that a worst case analysis gives:

$$\frac{\delta t}{\delta x} \le \frac{4}{5} \tag{102}$$

which is similar to that of Berzins [4], equation (34).

6.1 Adaptive order Algorithm.

The general strategy employed in changing the order of the polynomial used in the method of Berzins [4] is to use the highest order possible unless its use is precluded by the positivity conditions operating in such a way as to reduce the order. This strategy has been influenced by variable order strategies such as the h-p methods used by Biswas, Devine and Flaherty [6] in which successive polynomial derivatives are limited. For the purposes of changing the order only the space derivative terms are considered, although the mass matrix may still be modified independently. Moreover in the quadratic case the switching criteria are defined by the equations corresponding to the edges of elements and not to the interior. In changing from piecewise discontinuous to piecewise linear if $\beta_i = 1$ then the limiters are both zero so we stay with the discontinuous method of equation (42). In changing from linear to quadratic if $S_{-}^{j,n} = 0$ then then quadratic terms in the derivative are switched off and so a linear basis is used. In the case when $S_{-}^{i,n} = 0$ (and the method defaults to the linear method in approximating the the space derivative) it follows that values of s_i^n and $\frac{1}{s_{i-1}^n}$ are either less than 0 or greater than 1. Alternatively $s_i^n = \frac{1}{s_{i-1}^n}$ and so $\delta^2 u_{i+1}^n = \delta^2 u_{i-1}^n$ thus implying that second derivative approximations are constant and thus that a linear approximation is more appropriate. This approach is applied on a point by point basis.

6.2 Numerical Example

The numerical problem used to demonstrate the performance of the code is the same advection problem used by Berzins [4] whose solution is given by equations (16) and (17) above. A mesh of 101 evenly spaced points is used with a cfl number of 0.2. Figure 5 show the profiles at $t = 1.5$ and Table 1 shows the the errors in the L1 norm at the same time. The method of this section is described as C^0 quadratic and gives comparable results than the quadratic B-spline method in Berzins [4]. The value of R used in equation(19) is varied as in [4] and has the values 1 or 4. The van Leer limiter used by Berzins [4] gives results comparable with $R = 4$. In the case of the C^0 method described here the results for $R = 1$ are much worse than for $R = 4$. The penultimate column of the table shows the results with a lumped mass matrix and thus that this is mostly due to the treatment of the spatial derivative rather than the mass matrix.

Figure 5 shows that the method does preserve positivity as it should and that quadratic polymnomial approximations are used in the vicinity of the steep gradient.

L1 Error Norms						
	Finite Element					
Finite Volume using FE limiter	Linear Basis	Quadratic B spline	Cubic B spline	C^0 quadratic	C^0 lumped quadratic	R value used
2.1e-2	2.4e-2	2.1e-2	1.8e-2	2.0e-2	2.0e-2	4
4.3e-2	1.5e-2	1.5e-2	1.3e-2	5.8e-2	4.4e-5	1

Table 2
Comparison of Error Norms on Travelling Front Example

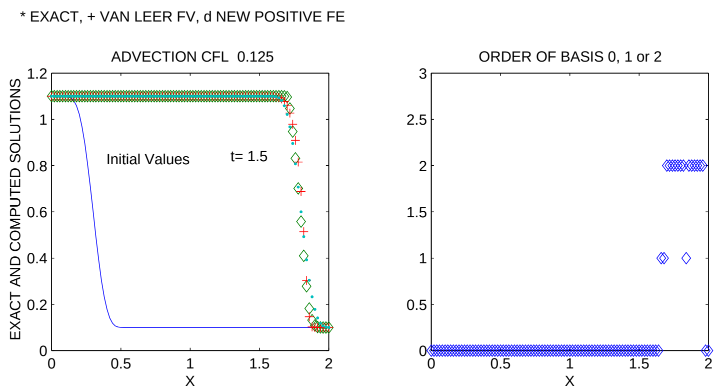

Fig. 5. Comparison of individual p version approximations

7 Further Developments.

There are a number of further develolpments that merit attention.

7.1 *Higher Order Methods.*

In considering the extension of the above ideas to higher order polynomials one constraint in the present method is that interpolation is avoided by the used of common points between the methods of different orders. The use of fourth order polynomials would make it possible to decompose a fourth order element into two quadratic polynomials which could then be decomposed into four linear elements. The quartic C^0 elements based on equally- spaced points suffer from a disadvantage in that the basis functions have sub-optimal approximation properties. This manifests itself in the plots of the basis functions N1 to N5 as shown in Figure 6 where the functions N2 and N4 both exceed one. The obvious alternative to this is to adopt a spectral element or DG approach with non-uniform mesh spacing, see [11,8].

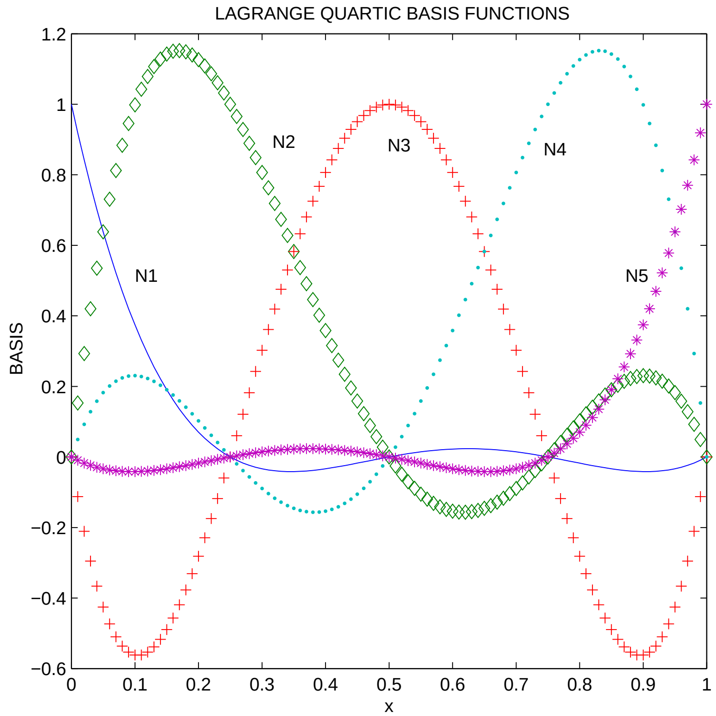

Fig. 6. Quartic Basis Functions with nodes at p/4 , p = 0,...4

7.2 Conservative Form.

Local conservation properties in hyperbolic schemes are important for ensuring that the numerical solution travels with the correct speed, [16]. This remains an issue with Galerkin schemes although Larson at al. [18] have shown how to post-process steady Galerkin method solutions to ensure conservation. In the case of the scheme defined above the piecewise constant method is well-known to be conservative and noting the results of Hou and LeFloch [16] who show how a conservative method may be obtained by switching from a non-conservative method to a conservative one may be applied to higher order schemes, see [4].

7.3 Nonlinear Conservation Laws.

The extension of the method described above to nonlinear conservation laws is considered by Berzins [4] and may be used equally well with the methods described here. An important issue is how to extend the ideas presented here to the case when approximate Riemann solvers need to be used.

7.4 *Two Dimensional Problems.*

Extensions to advection in two dimensions have been undertaken by Berzins and Hubbard [2]. Although initial results are promising further work is needed.

8 Summary

In this paper a novel approach to preserving positivity for variable-order finite element methods has been extended in a general way using the idea of filtered polynomial approximations. The approach relies on using a nonlinear form of the mass matrix in conjunction with positivity preserving conditions on the method coefficients.

References

[1] Arandiga F. Chiavassa G. and Donat R. Harten's Framework for Multiresolaution with Applications from Conservation Laws to Image Compression. GrAN Report 01-01 may 15th 2001. Departament de Matematica Aplicada , Universitat de Valencia, 4619900 Burfassot , Valencia SPAIN.

[2] Berzins M, Hubbard ME. Positive Finite Element Schemes on Irregular Triangular Meshes, *Paper in preparation* 2004.

[3] Berzins M. Modified mass matrices and positivity preservation for hyperbolic and parabolic pde's. *Communications in Numerical Methods in Engineering* **17**:659:666, 2001.

[4] Berzins M. Variable-Order Finite Elements and Positivity Preservation for Hyperbolic PDEs *Applied Numerical Mathematics* (To appear 2004)

[5] Brenan K. E. Campbell S.L. and Petzold L.R. Numerical Solution of Initial-Value problems in Differential Algebraic Equations *North Holland*, 1989.

[6] Biswas R., Devine K. and Flaherty J.E. Parallel adaptive finite element methods for conservation laws. *Applied Numerical Mathematics* 1995; **14**:2557–283.

[7] Cockburn B, and Shu C-W. Nonlinearly Stable Compact Schemes for Shock Calculations. *SIAM J. Numer. Anal.* 1994; **31**:607–627.

[8] Cockburn B, Karniadakis GE, Shu C-W. (eds). *Discontinuous Galerkin Methods, Theory Computations and Applications. Lecture Notes in Computational Science and Engineering 11* Springer Berlin Heidelberg, 2000; 3–53.

[9] Godunov SK. Finite difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Math Sbornik* 1959; **47**:271–306.

[10] Gottlieb D, and Shu C-W. Resolution properties of the Fourier method for discontinuous waves, in Analysis Algorithms and Applications of Spectral and High-Order Methods for Partial Differential equations edited by C,Bernardi and Y.Maday, Selected Papers from the international Conference on Spectral and High Order Methods (ICOSAHOM'92) Le Corum, Montpellier France June 22-26 1992 (North Holland Amsterdam 1994), p27.

[11] Gottlieb D, and Hesthaven J.S. Spectral Methods for Hyperbolic Problems. Journal of Computational and Applied Mathematics 128 (2001) 83-131.

[12] Gresho P.M. Saini R. L. *Incompressible Flow and the Finite Element Method, Volume 1, Advection Diffusion* Wiley , 1998.

[13] Harten A and Zwas G. Self Adjusting Hybrid Schemes for Shock Computations *J. of Computational Physics* 1972; **9**:568–583.

[14] Harten A Multiresolution Algorithms for the Numerical Solution of Hyperbolic Conservation Laws. *Comm. Pure and Applied Math.* 1995; **XIVIII**: 1305–1342.

[15] Hildebrand F.B. *Introduction to Numerical Analysisa* McGraw-Hill Book Company Inc 1956;

[16] Hou T.Y. and Le Floch P.G. Why nonconservative schemes converge to wrong solutions: error analysis. *Mathematics of Computation* 1994; **62**:497–530.

[17] Laney C. *Computational Gasdynamics*. Cambridge University Press.

[18] Larson M.G., Hughes T.J.R., Engel G. and Mazzei L. The continuous Galerkin method is conservative *J. of Computational Physics* 2000; **163**, 2:467-488.

[19] Segal A. Finite element methods for advection-diffusion equations. In *Numerical Methods for Advection Diffusion Problems. Notes on Numerical Fluid Mechanics, Volume 45* Vreugdenhil CB, Koren B (eds). Vieweg: Braunscweig/Wiesbaden, 1993; 195–214.

[20] Shu C-W and Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes. *J.Computational Physics* 1988; **77**:439–471.

[21] C. Schwab, p- and hp- Finite Element Methods - Theory and Applications in Solid and Fluid Mechanics Clarendon Press. 1998. Series: Numerical Mathematics and Scientific Computation

[22] Sheu T. W.h., Tsai S.F. and Wang M.M.T. A monotone finite element method with test space of Legendre polynomials. *Comput. Methods Appl. Mech. Engrg.* 1997; **143**:349–372.

[23] Sheu T. W.h., Lee P.H. and R.K. Lin Development of a high resolution hyperbolic model on quadratic elements *Comput. Methods Appl. Mech. Engrg.* 2003; **192**:5037–5036.