COMPUTATIONAL METHOD FOR SOLVING
THE BURGERS' EQUATION

Ahmed H. A. Ali

Mathematics Department, Faculty of Science,
Menoufia University, Shebein El-Koom, Egypt.

ABSTRACT

The nonlinear Burgers' equation is solved numerically by a method based on collocation of quintic splines over finite elements. A linear stability analysis is set up which shows that the scheme is unconditionally stable. Three standard problems are used to test the algorithm. All have analytic solutions so that the L_2- and L_∞-error norms can be used to estimate the accuracy of the numerical method. For the problems studied there are also published numerical solutions, so that further comparisons can be made. The method is found to be both accurate and efficient.

INTRODUCTION

Burgers' equation arises in the approximate theory of flow through a shock wave propagating in a viscous fluid (Cole) and in the modeling of turbulence (Burgers). Burgers' equation and Navier-Stokes equation are similar in the form of their nonlinear terms and in the occurrence of higher order derivatives with small coefficients in both (Varoglu et al.).

Burgers' equation can only be solved analytically for a restricted set of initial conditions. Difficulties also arise in the numerical solution of Burgers' equation for small values of the viscosity coefficient, that is large Reynolds number, which corresponds to steep wave fronts. Recently, Galerkin and Petrov-Galerkin finite element methods, involving a time dependent grid, have been used successfully to obtain accurate numerical solutions (Cadwell et al. and Herbst et al.) even for small viscosity coefficients.

In this paper we set up a finite element approach using a collocation method with quintic spline interpolation functions and a constant grid of elements and test its usefulness when the viscosity coefficient is both large or small. The collocation method has two great advantages, in that the set up procedure does not involve integrations, and that the resulting matrix equation is banded with a small band width. Quintic splines have the additional advantage that the resulting matrix system is pentadiagonal and so can be solved using the pentadiagonal algorithm. In the method described here a single (N+1)x(N+1) pentadiagonal matrix equation is obtained. A similar approach has been reported recently (Cadwell) which replaces the partial differential equation by a set of 3 independent tridiagonal NxN matrix equations for nodal values of function and derivatives. Cubic splines have also been used by Rubin and Graves.

THE GOVERNING EQUATION AND FINITE ELEMENT SOLUTION

An appropriate form for Burgers' equation is

$$U_t + U U_x - \nu U_{xx} = 0 \qquad a \le x \le b \tag{1}$$

where the subscripts t and x denote the differentiation and ν is positive parameter. Boundary conditions are chosen from:

$$U(a,t) = \alpha \qquad \text{and} \qquad U(b,t) = \beta,$$

The region is partitioned into N finite elements of equal length h by the knots x_i such that $a = x_0 < x_1 < \quad < x_N = b$. The quintic splines ϕ_i with knots at x_i form a complete basis for the functions defined over [a,b]. A global approximation $U_N(x,t)$ to the solution $U(x,t)$ is given by

$$U_N(x,t) = \sum_{i=-2}^{N+2} \delta_i(t) \, \phi_i(x) \tag{2}$$

where the δ_i are time dependent quantities to be determined from the boundary and collocation conditions. Each quintic spline spans 5 finite

elements, so that 5 splines cover each element. The spline $\phi_j(x)$ and its 2 principal derivatives vanish outside the region $[x_{i-3}, x_{i+3}]$. In Table I the values of ϕ_i and its principal derivatives at the relevant knots are listed. At the knots x_i the numerical solution $U_N(x,t)$ is given by

$$
\left.
\begin{aligned}
U_i &= \delta_{i-2} + 26\delta_{i-1} + 66\delta_i + 26\delta_{i+1} + \delta_{i+2} \\
hU_i' &= 5\,\delta_{i+2} + 50\,\delta_{i+1} - 50\,\delta_{i-1} - 5\,\delta_{i-2} \\
h^2U_i'' &= 20(\delta_{i-2} + 2\delta_{i-1} - 6\delta_i + 2\delta_{i+1} + \delta_{i+2})
\end{aligned}
\right\}
\tag{3}
$$

The function U and its first 2 derivatives are continuous across element boundaries. We substitute (2) into (1), and identify the collocation points with the knots and then use equation (3) to evaluate U_i and its space derivative (Prenter).

Table I : The quintic spline ϕ

x	x_{i-3}	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}	x_{i+3}
ϕ_i	0	1	26	66	26	1	0
$h\phi_i'$	0	5	50	0	-50	-5	0
$h^2\phi_i''$	0	20	40	-120	40	20	0

Thus implementing the method of lines leads to a set of ordinary differential equations with the form

$$
\dot{\delta}_{i-2} + 26\dot{\delta}_{i-1} + 66\dot{\delta}_i + 26\dot{\delta}_{i+1} + \dot{\delta}_{i+2} +
$$

$$
\frac{5}{h}Z_i(\delta_{i+2}+10\delta_{i+1}-10\delta_{i-1}-\delta_{i-2})-\frac{20}{h^2}v(\delta_{i-2}+2\delta_{i-1}-6\delta_i+2\delta_{i+1}+\delta_{i+2})=0,
$$

$$
i = 0, 1, ..., N,
\tag{4}
$$

where

$$
Z_i = \delta_{i-2} + 26\delta_{i-1} + 66\delta_i + 26\delta_{i+1} + \delta_{i+2}
$$

The system of ordinary differential equations may now be solved using an appropriate software package, for example, by using the routine D02CAF of the Numerical Algorithms Group program library.

In an alternative approach, which is used in this paper, a recurrence relationship based on a Crank-Nicolson approximation in time is derived. Suppose that $\mathbf{d} = (\delta_{-2}, \delta_{-1}, \delta_0, ..., \delta_{N+2})^T$, the vector of nodal parameters, is linearly interpolated between two time levels n and n+1 then $\mathbf{d}$ and its time derivative are given by

$$\mathbf{d} = \frac{1}{2}(\mathbf{d}^{n+1} + \mathbf{d}^{n}), \quad \overset{\bullet\bullet}{\mathbf{d}} = \frac{1}{\Delta t}(\mathbf{d}^{n+1} - \mathbf{d}^{n}) \tag{5}$$

where $\mathbf{d}^n$ are the parameters at the time $n\Delta t$. Hence using Eq.(5) in Eq.(4), we have for each knot an equation relating parameters at adjacent time levels, δ_i^{n+1} to δ_i^{n} (Ali and Gardner et al.),

$$\alpha_{i1}\delta_{i-2}^{n+1} + \alpha_{i2}\delta_{i-1}^{n+1} + \alpha_{i3}\delta_{i}^{n+1} + \alpha_{i4}\delta_{i+1}^{n+1} + \alpha_{i5}\delta_{i+2}^{n+1} =$$

$$\beta_{i1}\delta_{i-2}^{n} + \beta_{i2}\delta_{i-1}^{n} + \beta_{i3}\delta_{i}^{n} + \beta_{i4}\delta_{i+1}^{n} + \beta_{i5}\delta_{i+2}^{n},$$

$$i = 0, 1, 2, ..., N \tag{6}$$

where

$$\alpha_{i1} = 1-R_1 Z_i-R_2, \qquad \alpha_{i2} = 26-10R_1 Z_i-2R_2,$$
$$\alpha_{i3} = 66+6R_2, \qquad \alpha_{i4} = 26+10R_1 Z_i-2R_2, \ \alpha_{i5} = 1+R_1 Z_i-R_2.$$
$$\beta_{i1} = 1+R_1 Z_i+R_2, \qquad \beta_{i2} = 26+10R_1 Z_i+2R_2,$$
$$\beta_{i3} = 66-6R_2, \qquad \beta_{i4} = 26-10R_1 Z_i+2R_2, \ \beta_{i5} = 1-R_1 Z_i+R_2.$$
$$Z_i = \delta_{i-2} + 26\delta_{i-1} + 66\delta_i + 26\delta_{i+1} + \delta_{i+2}, \ R_1 = \frac{5\Delta t}{2h} \ \text{and} \ R_2 = \frac{10v\Delta t}{h^2}$$

The system (6) consists of N+1 nonlinear equations in N+5 unknowns $(\delta_{-2}, \delta_{-1}, \delta_0, ..., \delta_{N+2})^T$. To obtain a solution to this system we need 4 additional constraints. These are obtained from the boundary conditions, and can be used to

eliminate δ_{-2}, δ_{-1}, δ_{N+1}, δ_{N+2} from the set (6) which then becomes a matrix equation for the N+1 unknowns $d^{n+1} = (\delta_0, \delta_1, \delta_2, ..., \delta_N)^T$.

$$A(d^n) \, d^{n+1} = B(d^n) \, d^n + r \tag{7}$$

where $A(d^n)$ and $B(d^n)$ are pentadiagonal matrices, and r is a N+1 vector which depends on the boundary conditions.

The time evolution of the approximate solution $U_N(x,t)$ is determined by the time evolution of the vector d^n. This is found by repeatedly solving the recurrence relationship once the initial vector d^0 has been computed from the initial conditions. The recurrence relationship (7) is pentadiagonal and a direct algorithm for the rapid solution of the equations is available. However, an inner iteration is also needed, at each time step, to cope with the nonlinear terms. The following solution procedure is followed.

1. At time $t = 0$, for the initial step of the inner iteration we approximate A and B by A^* and B^* calculated from d^0 only and obtain a first approximation to d^1 from (7). We then iterate, using (7) with matrices A and B calculated from $d = 0.5(d^0 + d^1)$ to refine the approximation to d^1.

2. At all other time steps we use for matrices A and B, at the first step of the inner iteration, A^* and B^* derived from $d^* = 0.5(d^n + d^{n-1})$ to obtain a first approximation to d^{n+1} by solving (7). We then iterate, using (7) with matrices A and B calculated from $d = 0.5(d^n + d^{n+1})$, two or three times to refine the approximation to d^{n+1}.

STABILITY ANALYSIS

An investigation into the stability of the numerical scheme (6) is based on the von Neumann theory in which the growth factor of a typical of Fourier mode defined as

$$\delta_j^n = \hat{\delta}^n e^{ijkh} \tag{8}$$

where, k is the mode number and h is the element size, is determined for a linearisation of the numerical scheme.

The nonlinear term UU_x of Burgers' equation (Ali and Gardner et al.) is linearised by making the quantity U locally constant which is equivalent to assuming that the corresponding values of δ_j^n are equal to a local constant d. Substituting the Fourier mode (8) in equation (6) we obtain

$$\hat{\delta}^{n+1} = g\, \hat{\delta}^n$$

where the growth factor g has the form

$$g = \frac{a - ib}{a_1 + ib} \tag{9}$$

where

$$a = (1 - R_2)\cos(2kh) + (26 + 2R_2)\cos(kh) + 66 + 6R_2,$$

$$a_1 = (1 + R_2)\cos(2hk) + (26 - 2R_2)\cos(kh) + 66 - 6R_2,$$

$$b = R_1^*(\sin(2kh) + 10\sin(kh)),\quad R_1^* = (\frac{300d\Delta t}{h}),\qquad R_2 = \frac{10\Delta t v}{h^2}$$

Taking the modulus of Eq.(9) gives $|g| < 1$; therefore the linearised scheme is unconditionally stable.

THE INITIAL STATE

From the initial condition U(x,0) on the function U(x,t) we must determine the initial vector $\mathbf{d}^0$ so that the time evolution of $\mathbf{d}$, using (7), can be started.

Firstly rewrite Eq.(2) for the initial condition as

$$U_N(x,0) = \sum_{j=-2}^{N+2} \delta_j^0 \phi_j(x) \tag{10}$$

30

where δ_j^0 are unknown parameters to be determined. To do this we require $U_N(x,0)$ to satisfy the following constraints:

(a) It must agree with the initial condition $U(x,0)$ at the knots x_j, j=0,1,...,N.

(b) The first and the second derivatives of the approximate initial condition shall agree with those of the exact initial condition at both ends of the range; Eq.(3) produces two further equations.

The initial vector $\mathbf{d}^0$ is then determined as the solution of a matrix equation derived from Eq.(3)

$$\mathbf{M} \ \mathbf{d}^0 = \mathbf{b} \tag{11}$$

THE TEST PROBLEMS

We now obtain the numerical solutions of Burgers' equation for two standard problems. To measure the accuracy of the numerical methods we compute the difference between the analytic and numerical solutions at each mesh point after specified time steps, and use these to compute the discrete L_2- and L_∞- error norms.

(a) Consider the well known analytic solution of Burger's equation:

$$u(x,t) = \frac{x/t}{1 + (t/t_0)^{1/2} \ \exp(x^2/4vt)} \quad , \quad t \geq 1 \tag{12}$$

where $t_0 = \exp(1/8v)$. We take initial condition to be the Eq.(12) evaluated at time $t = 1$, and use the boundary conditions $u(a,t) = u(b,t) = 0$.

To test the accuracy of the numerical scheme we have calculated the L_2- and L_∞- error norms for problem (a). These are given in Table 2 and 3. In Table 2, we compare between two methods for solutions of Burgers' equation. From Table 2 with $v = 0.005$ we see that the calculated errors of the previous scheme

31

(Gardner et al.) is smaller than the present scheme.

Table (2)

$\nu = 0.005 \quad \Delta t = 0.1 \quad h = 0.02$				
Time		$t = 1.7$	$t = 2.4$	$t = 3.1$
$L_2 X10^3$	Present	1.293	0.807	1.137
	Gardner	0.857	0.423	0.235
$L_\infty X10^3$	Present	3.836	2.332	4.790
	Gardner	2.576	1.242	0.688

Table (3)

	$\nu = .0005 \quad \Delta t = 0.01 \quad h = 0.005$			
Time		1.75	2.5	3.25
$L_2 X \; 10^3$	Present	0.536	0.180	0.108
	Gardner	0.567	0.308	0.239
$L_\infty X \; 10^3$	Present	5.868	1.582	0.728
	Gardner	5.880	2.705	2.291

Table 3 shows us that for $\nu = 0.0005$ the evaluated errors for this problem using the present scheme is smaller than the previous scheme (Gardner et al.), i.e in the case with small value of ν (a big Reynold's number) the present method is a best fit for solving the Burger's equation.

it is noticed that as the viscosity value ν decreases the errors increase but remain acceptable small.

(b) A second analytic solution is of Burgers' equation (Rubin et al.)

$$u(x,t) = \frac{1}{2} [1 - \tanh\{\frac{1}{4v}(x - 15 - \frac{1}{2} t)\}]$$ (13)

We take as initial condition (13) at t = 0 over the range $0 \le x \le 30$ with boundary conditions u(0,t) = u(1,t) are used.

Table (4)

ν = 0.5		Δt =0.025	h =1/300			
Time		t =2	t =4	t =6	t =8	t =10
$L_2 X\ 10^3$	Present	0.002	0.003	0.005	0.013	0.035
	Gardner(1991)	0.006	0.010	0.014	0.022	0.043
$L_\infty X\ 10^3$	Present	0.001	0.002	0.006	0.016	0.045
	Gardner(1991)	0.004	0.006	0.007	0.017	0.045

Table (5)

ν = 1/8		Δt = 0.025	h = 1/300			
Time		t =2	t =4	t =6	t =8	t =10
$L_2 X\ 10^3$	Present	0.028	0.035	0.034	0.035	0.035
	Gardner(1991)	0.228	0.227	0.227	0.227	0.226
$L_\infty X\ 10^3$	Present	0.034	0.039	0.040	0.041	0.041
	Gardner(1991)	0.346	0.339	0.340	0.347	0.226

For ν = 1/2 a very weak shock wave develops, when ν = 1/8 we obtain a moderate shock wave and when ν = 1/24 a strong shock wave is produced. As the

value of ν is decreased the propagation front becomes steeper. The L_2- and L_∞-error norms for the these simulations are given in Tables 4, 5 and 6. For a large values of ν the errors are small and as the value of ν is decreased the errors tend to increase, but for all the values of ν used here, the errors are still acceptable.

Table (6)

$\nu = 1/24$		$\Delta t = 0.025$	$h = 1/300$			
Time		t =2	t =4	t =6	t =8	t =10
$L_2 X 10^3$	Present	0.277	0.279	0.279	0.279	0.279
	Gardner(1991)	3.431	3.426	3.426	3.426	3.429
$L_\infty X 10^3$	Present	0.624	0.626	0.626	0.626	0.626
	Gardner(1991)	4.430	4.336	4.363	4.402	4.432

CONCLUSION

We have shown that the collocation method with quintic spline interpolation functions over constant sized finite elements is capable of producing numerical solutions for the Burgers' equation of high accuracy even for small values of the viscosity. We therefore conclude that the algorithm outlined in this paper is a very serious candidate when accurate and efficient solution of Burgers' equation are required.

REFERENCES

Ali, A.H.A. (1989): Finite Element Studies of the Korteweg-de Vries Equation, PhD. Thesis, University College of North Wales, Bangor, Gwynedd (U.K.).

Ali, A.H.A.; Gardner, L.R.T.; Gardner, G.A. (1992): A collocation solution for the Burgers' equation using cubic B-spline finite elements, Computer Methods in Applied Mechanics and Engineering, 100, p. 325-337.

Burgers, J.(1948): A Mathematical Model Illustrating the Theory of Turbulence, Advances in Applied Mechanics, Academic Press, p. 171-199.

Caldwell, J., Wanless, P. and Cook, A.E. (1981): A Finite Element Approach to Burgers' Equation, Appl. Math. Modeling., 5, p. 189-193.

Caldwell, J. (1987): Application of Cubic Splines to the Nonlinear Burgers' equation, "Numerical Methods for Nonlinear Problems", (Eds. E Hinton et al: Pineridge Press) Vol.3 , p. 253-261.

Christie, I., Griffiths, D. F., Mitchell, A. R., and Sanz-Serna,J.M. (1981): Product Approximation for Nonlinear Problems in the Finite Element Method, IMA. J. Num. Anal., 1, p. 253-266.

Cole, J. D. (1951): On a Quasi-linear Parabolic Equation Occurring in Aerodynamics, Quarterly of Applied Mathematics, 9, p. 225-236.

Gardner, L. R. T., Gardner, G. A. and Ali, A. H. A. (1990): Solitons of the Modified KdV Equation, in Numerical Methods in Engineering, ed. G N Pande & J. Middleton, Elsevier Applied Science, London, 1, p. 590-597.

Gardner, L. R. T., Gardner, G. A. and Ali, A.H.A. (1991): A Method of Lines Solution for Burgers' Equation, Computational Mechanics, Cheung, Lee & Leung (eds), Balkema, Rotterdam, p. 1555-1561.

Herbst, B.M., Schoombie, S.W. and Mitchell, A.R. (1982): A Moving Petrov-Galerkin Method for Transport Equations", Int. J. for Num. Meth. in Engng., 18, p. 1321-1336.

Nguyen, H. and Reynen, J. (1987): A Space-time Finite Element Approach to Burgers' Equation, "Numerical Methods for Nonlinear Problems", (Eds E Hinton et al: Pineridge Press), Vol.3, p. 718-728.

Rubin, S. G. and Graves, R. A. (1975): Viscous Flow Solutions With a Cubic Spline Approximation, Computers and Fluids, Pergamon Press, 3, 1-36.

Prenter, P.M. (1975): Splines and Variational Methods, J. Wiley, New York.

Varoglu, E. and Finn, W. D. L. (1983): Space-time Finite Elements Incorporating Characteristics for the Burgers' Equation, Int. J. for Num. Meth. in Engng., 16, p. 171-184.

طريقة حسابية لحل معادلة بيرجر

أحمد.حسن أحمد على

هذا البحث يعتمد أساساً على طريقة العنصر المحدود لحل معادلة بيرجر وهى معادلة تفاضلية جزئية غير خطية وذلك نظراً لإستخدامتها في العديد من التطبيقات الفيزيائية. وقد إسخدمنا إحدى طرق العنصر المحدود وهى طريقة نقاط الملاءمة التى تعتمد على الوصلات الخماسية الشكل. وقد تم عمل مقارنة بين النتائج التى توصلنا إليها والنتائج السابقة ووجد أن الطريقة التى إستخدمناها أكثر دقة وأكثر كفاءة. ولهذا نوصى بتطبيق هذه الطريقة على المعادلة المشابه لها.