A Triangular Spectral Element Method for the 2-D Viscous Burgers Equation

Mengy Zhang and Hua Wu
Department of Mathematics, Shanghai University, Shanghai, 200444, China
Email: hwu@staff.shu.edu.cn

Abstract. A triangular spectral element method is established for the two-dimensional viscous Burgers equation. In the spatial direction, a new type of mapping is applied. We splice the local basis functions on each triangle into a global basis function. The second-order Crank-Nicolson/leap-frog (CNLF) method is used for discretization in the time direction. Due to the use of a quasi-interpolation operator, the nonlinear term can be handled conveniently. We give the fully discrete scheme of the method and the implementation process of the algorithm. Numerical examples verify the effectiveness of this method.

Keywords. Burgers equation, triangular spectral element method, quasi-interpolation.

1. Introduction

The spectral method is one of the important numerical methods for solving partial differential equations. Its main advantage is high accuracy, especially for infinitely smooth problems, the convergence rate can reach infinite order. Because of its advantages, the spectral method is widely used in scientific and engineering computations. In recent years, the triangular element spectral method has been developed [1,2,3], that is, the spectral method is used in the triangular element while maintaining the high accuracy. Through the realization of the triangular element spectral method, due to the flexibility of the triangle itself, the application of triangulation to divide the region boundary into triangles can make the spectral method better applied to complex regions.

Recently, a new type of mapping is proposed in [4]. This mapping is a one-to-one mapping, which maps a vertex of a quadrilateral to the midpoint of the hypotenuse of the triangle, making the distribution of nodes more uniform than frequently used Duffy mapping [5]. This new type of mapping is applied in elliptic equation and a class of irrational basis functions is constructed on a triangular element in [6]. In [7], comparing Duffy mapping with this new type of mapping. By introducing auxiliary variables, a mixed triangular spectral element method is constructed in complex regions.

The Burgers equation is a classic nonlinear second-order partial differential equation which describes dissipation on the energy balance. It originates from the turbulence model [8]. It is often used in engineering computations [9] and as a model equation for testing numerical methods.

Let \( \Omega = [-1,1]^2 \). Consider the following two-dimensional viscous Burgers equation.
where \( T \) and \( v \) are the positive constants. \( F(u) \) is a function of a certain smoothness class. Under the above conditions, the solution of (1) is unique. If the Dirichlet boundary condition is nonhomogeneous, we can convert the boundary condition to homogeneous, and then consider the above situation.

In this paper, we present a triangular spectral element method for (1) and give its semi-discrete scheme and fully discrete scheme. We use polynomials of different degrees and triangulations of different numbers to approximate the unknown functions. Numerical results show the effectiveness of this method.

2. Triangular Spectral Element Method

In this section, we discrete spatial direction through triangular spectral element method and discrete time through the second-order Crank-Nicolson/leap-frog (CNLF) scheme to obtain the corresponding fully discrete scheme.

2.1. Preliminaries

Let \( \langle \cdot , \cdot \rangle_{\Omega} \) be the inner product of \( L^2(\Omega) \). For an integer \( m > 0 \), let \( H^m(\Omega) \) be the standard Sobolev space. Define

\[
H^1_0(\Omega) = \left\{ v \big| H^1(\Omega) : v|_{\partial \Omega} = 0 \right\}.
\]

For an integer \( N > 0 \), let \( \mathbb{L} = \{(\xi, \eta) : -1 \leq \xi, \eta \leq 1\} := \Lambda^2 \) and \( P_N(\Lambda) \) be the polynomial of degree less than \( N \) in domain \( \Lambda \). Considering the new type of mapping in [4], let \( \Gamma = \{\square_k\}_{k=1}^{K} \) be triangulations of domain \( \Omega \), and we usually assume the triangle \( \triangle_k \) to be shaped regular. For the convenience of calculation, we denote \( M \) as the number of quadrilateral divisions in \( x \) and \( y \) direction and get triangles through connecting with the left diagonal of these quadrilaterals.

The new type of mapping \( T_k : \square \rightarrow \triangle_k \) is as follow

\[
\begin{align*}
x &= x_{a_0} \frac{1+\xi}{8} + x_{a_0} \frac{1-\xi}{4} + x_{c_0} \frac{3-\xi}{8}, \\
y &= y_{a_0} \frac{1+\xi}{8} + y_{a_0} \frac{1-\xi}{4} + y_{c_0} \frac{3-\xi}{8}, \\
&\quad \forall (\xi, \eta) \in \mathbb{L}.
\end{align*}
\]

We respectively chose \( A(x_{a_0} , y_{a_0}), B(x_{a_0} , y_{a_0}), C(x_{c_0} , y_{c_0}) \) as the three vertices of \( \triangle_k \). However, this kind of mapping will bring singularity [4] in calculation. We divide the domain in this way and arrange the singular point at the midpoint of the shared side of the two triangles, making it easier to calculate.

Let \( J_k \) and \( J_k^{-1} \) be the Jacobian matrix and Jacobian inverse matrix of the mapping \( T_k \). Denote \( J_k \) is the corresponding Jacobian determinant. In this article, by mapping \( T_k \) the function \( u(x, y) \) in \( \square \) and the function \( \hat{u}^k(\xi, \eta) \) in \( \triangle_k \) can be matched, which is

\[
\begin{align*}
\frac{\partial u}{\partial t} + \nabla \cdot F(u) - \nu \nabla^2 u &= f(x, y, t), \quad (x, y) \in \Omega, t \in (0, T], \\
u(x, y, t) &= 0, \quad (x, y) \in \partial \Omega, t \in (0, T], \\
u(x, y, 0) &= u_0(x, y), \quad (x, y) \in \Omega \cup \partial \Omega.
\end{align*}
\]
\[ u(x, y) \big|_{\gamma_k} \circ T_k = \hat{u}_k(\xi, \eta), \quad \forall (x, y) \in \Omega, \forall (\xi, \eta) \in \square. \]

2.2. Approximation Space and Quasi-Interpolation Operator

As defined in [10], we chose the spectral approximation space with the consistency condition as follow

\[ V_{N, \Gamma} := \left\{ \omega \in H^1(\Omega), \hat{\omega}^k \in \left[ P_N(\Lambda) \right]^2, \left( \partial_\xi \hat{\omega}^k + \partial_\eta \hat{\omega}^k \right) \big|_{(1,1)} = 0 \right\}, \]  

where \( k \neq \Gamma \). Next, a set of global basis functions of \( V_{N, \Gamma} \) are constructed by splicing local basis functions.

Let \( \{ h_j, j = 0,1,\ldots, N \} \) be the Lagrange interpolation basis functions based on Legendre-Gauss-Lobatto (LGL) points corresponding to the interval \( \Lambda \). And we denote \( \phi_j(\xi, \eta) = h_j(\xi)h_j(\eta) \) and \( \mu_j = -\frac{h_j'(1)}{2h_N'(1)}, j = 0,1,\ldots, N-1 \). Let \( \tilde{h}_j(z) = h_j(z) + \mu_j h_N(z), 0 \leq j \leq N-1 \), we chose local basis functions of the triangle \( \square_k \) as

\[ \hat{\psi}_o(\xi, \eta) := \begin{cases} h_j(\xi)h_j(\eta), & 0 \leq i, j \leq N-1, \\
\tilde{h}_j(\xi)h_N(\eta), & 0 \leq i \leq N-1, \\
h_N(\xi)\tilde{h}_j(\eta), & 0 \leq j \leq N-1. \end{cases} \]  

We define \( \psi^k_o(x, y) := \hat{\psi}_o(\xi, \eta) \circ T_k^{-1} \). The \( C^0 \) global basis functions are chosen as following, let \( \{(\xi_i, \eta_j)\}_{i,j=0}^N \) and \( \{\omega_{ij}\}_{i,j=0}^N \) are the LGL points and the corresponding weights in \( \Lambda^2 \). Define

\[ \Omega_N := \left\{ (x, y) : (x, y) = T_k \left( \xi_i, \eta_j \right), 0 \leq i, j \leq N, 1 \leq k \leq K \right\}. \]  

We can know that \( \Omega_N \) is a set of all the LGL points in \( \Omega \). Let \( N_\Omega \) is the number of elements in \( \Omega_N \), and \( \Omega_N \) can also be expressed in order as

\[ \Omega_N = \left\{ (x, y) : (x, y) = (x_\kappa, y_\kappa), \kappa = 1,2,\ldots, N_\Omega \right\}, \]  

where \( \kappa = \kappa(i, j, k), 0 \leq i, j \leq N, 1 \leq k \leq K \). And we define \( \Omega_\kappa \) is the set of all singular points in \( \Omega \), \( K_\Omega \) is the number of points contained in \( \Omega_\kappa \).

We chose basis functions corresponding to the internal points of domain \( \square_k \) as

\[ \varphi_\kappa(x, y) = \begin{cases} \psi^k_o(x, y), & (x, y) \in \square_k, \\
0, & \text{other}. \end{cases} \]  

For the LGL points on the edge of the triangles which are jointed, such as the edge \( \gamma_{kk'} := \partial \square_k \cap \partial \square_{k'}, k', k'' \in \{1,\ldots, K\} \). The corresponding basis function is
\[
\varphi(x, y) = \begin{cases} 
\psi_{ij}^k(x, y), & (x, y) \in I_i, \\
\psi_{ij}^{*k}(x, y), & (x, y) \in I_{*i}, \\
0, & \text{other.}
\end{cases}
\]

where \( \psi_{ij}^k(x, y) \) and \( \psi_{ij}^{*k}(x, y) \) are the local basis functions corresponding to the same LGL point of jointed triangles \( I_i \) and \( I_{*i} \), respectively.

Similarly, the local basis functions on each triangle can be spliced into a global basis function. In this way, non-nodal basis functions can be represented by nodal basis functions. So they still have the property of tensor product, making them easy to calculate. We next introduce a quasi-interpolation operator from space \( C(\Omega) \) to the spectral approximation space \( V_{N,I} \).

For any \( u(x, y) \in C(\Omega) \), let \( u_k = \hat{u}(\xi, \eta), 0 \leq i, j \leq N, 1 \leq k \leq K \), where \( \kappa = \kappa(i, j, k) \). Define the quasi-interpolation operator by using the global basis functions of \( V_{N,I} \), which leads to

\[
I_{N,I} : C(\Omega) \rightarrow V_{N,I}
\]

\[
I_{N,I}u(x, y) := \sum_{k=1}^{N_i-K_i} u_k \phi_k(x, y), \quad \forall (x, y) \in \Omega.
\]

2.3. Scheme
The weak form to (1) is: Find \( u \in H_0^1(\Omega) \), for any \( v \in H_0^1(\Omega) \) such that

\[
\begin{align*}
\int (\partial_t u, v) + (F, \nabla \cdot v) + \nu (\nabla u, \nabla v) = (f, v), & \quad \forall t \in (0, T]. \\
(u(x, y, 0), v) = (u_0(x, y), v), & \quad \forall t \in (0, T].
\end{align*}
\]

(6)

Define \( V^0_{N,I} := V_{N,I} \cap H_0^1(\Omega) \). The semi-discrete scheme to (1) is: Find \( u_N \in V^0_{N,I} \), for any \( v_N \in V^0_{N,I} \) such that

\[
\begin{align*}
\int (\partial_t u_N, v_N) - (F, \nabla \cdot v_N) + \nu (\nabla u_N, \nabla v_N) = (f, v_N), & \quad \forall t \in (0, T]. \\
(u_N(x, y, 0), v_N) = (u_0(x, y), v_N), & \quad \forall t \in (0, T].
\end{align*}
\]

(7)

The second-order Crank-Nicolson/leap-frog (CNLF) scheme is applied to discretize in the time direction. Let \( S_t = \{ k \tau : k = 1, 2, \ldots, n_t, t = n_t \tau \} \), we can give the fully discrete scheme to (1), which is:

Find \( u_N \in V^0_{N,I} \), for any \( v_N \in V^0_{N,I} \) such that

\[
\begin{align*}
\int (u_{N,t}, v_N) - (F, \nabla \cdot v_N) + \nu (\nabla u_{N,t}, \nabla v_N) = (f, v_N), & \quad \forall t \in S_{t-\tau}. \\
(u_N(x, y, 0), v_N) = (u_0(x, y), v_N), & \quad \forall t \in S_{t-\tau}.
\end{align*}
\]

(8)

where \( \tau \) is the step length in the time direction. The definitions of \( u_i(t) \) and \( \bar{u}(t) \) are as follow

\[
u_i(t) = \frac{u(t + \tau) - u(t - \tau)}{2\tau}, \quad \bar{u}(t) = \frac{u(t + \tau) + u(t - \tau)}{2}.
\]
The specific algorithm of (8) is given below, let 
\[ u_N(x, y, t) = \sum_{k=1}^{N_\Omega-K_\Omega} u_k(t) \phi_{k}(x, y), \]
\[ U(t) = (u_1(t), \ldots, u_{N_\Omega-K_\Omega}(t))^T. \]
Substitute \( v_N \) for the global basis functions of \( V_{N, T} \) in (8), which leads to
\[
\begin{aligned}
M \frac{dU}{dt} + \nu K U &= L(t) + \alpha(t), \\
U(0) &= I_{N, T} u_0.
\end{aligned}
\]
(9)
where \( M = (\phi_{\kappa}, \phi_{\kappa'})_{\Omega} \) and \( K = (\nabla \phi_{\kappa}, \nabla \phi_{\kappa'})_{\Omega}, \kappa, \kappa' \in \{1, \ldots, N_\Omega - K_\Omega\} \). They are the mass matrix and the stiffness matrix, respectively. \( L(t) = (F, \nabla \cdot \phi_{\alpha})_{\Omega}, \alpha(t) = (f, \phi_{\alpha})_{\Omega} \). Then the second-order CNLF scheme is used to discretize (9) in the time direction. Let \( U^t \) be the value of \( U \) at time \( t \), we can get
\[
\begin{aligned}
M \frac{U^{t+\tau} - U^{t-\tau}}{2\tau} + \nu K \frac{U^{t+\tau} + U^{t-\tau}}{2} &= L(t) + \alpha(t), \\
U^0 &= I_{N, T} u_0.
\end{aligned}
\]
(10)
By simplifying the first equation of (10), we get
\[
(M + \tau \nu K) U^{t+\tau} = \eta^{t+\tau},
\]
(11)
where \( t = \tau, 2\tau, \ldots, T - \tau \). On the right-hand side of the equation (11), \( \eta^{t+\tau} \) is a known value based on calculation from the value at time \( t \) and time \( t - \tau \). We calculate initial value by
\[
U^t = I_{N, T} \left( u_0 + \tau \frac{\partial u}{\partial t}_{|t=0} \right).
\]
In this way, \( U^{2\tau}, U^{3\tau}, \ldots, U^T \) can be computed successively by (11).

Remark 2.1 All the integrations in this section are calculated using the high-precision Gaussian quadrature formula as
\[
(u, v)_{\Omega} \approx \sum_{k=1}^{K} \sum_{i, j=0}^{N} ((uv)|_{\xi} \circ T_{k} \cdot J_{k})_{(\xi, \eta)} \omega_{\eta}.
\]

3. Numerical Examples
In this section, we give the results of some numerical examples. And we report \( L^2 \)-norm which is defined by
\[
E^2 = \left( \sum_{i, j=0}^{N} \sum_{k=1}^{K} \left( \hat{u}^k - (\hat{u}_N)^k \right)^2 J_{k} \right)_{(\xi, \eta)} \omega_{\eta}^{\frac{1}{2}},
\]
Example 3.1 Consider (1) with \( f(x, y, t) = 0, F(u) = \frac{u^2}{2} \), which leads to
The initial condition $u_0(x, y)$ is taken from the exact solution:

$$u(x, y, t) = \frac{1}{2} - \tanh \left( \frac{x + y - t}{2\nu} \right).$$

The example is calculated by (8). From the table 1 and the table 2, we can see that for different viscosity coefficients $\nu$, the method maintains the high precision. When taking a smaller viscosity coefficient $\nu$, we can increase $N$ and $M$ to keep the algorithm highly accurate.

**Table 1.** $E_{L^2}$ at $n = 1, t = 10^{-4}, T = 1$ of the method (8).

| N | M=1 | M=2 | M=4 | M=6 |
|---|-----|-----|-----|-----|
| 2 | 2.32e-4 | 3.25e-5 | 3.93e-6 | 1.11e-6 |
| 4 | 8.87e-7 | 2.68e-8 | 8.30e-10 | 1.10e-10 |
| 6 | 3.02e-9 | 3.83e-11 | 2.20e-11 | 2.21e-11 |
| 8 | 2.88e-11 | 2.67e-11 | 2.24e-11 | 2.24e-11 |

**Table 2.** $E_{L^2}$ at $n = 0.5, t = 10^{-4}, T = 1$ of the method (8).

| N | M=1 | M=2 | M=4 | M=6 |
|---|-----|-----|-----|-----|
| 4 | 1.64e-5 | 8.12e-7 | 2.63e-8 | 3.19e-9 |
| 6 | 1.67e-7 | 3.44e-9 | 5.04e-10 | 5.13e-10 |
| 8 | 5.68e-9 | 5.44e-10 | 5.10e-10 | 5.16e-10 |
| 10 | 5.77e-10 | 5.43e-10 | 5.13e-10 | 5.18e-10 |

**Example 3.2** Consider (12), the initial condition $u_0(x, y)$ is taken from the exact solution [11, 12]:

$$u(x, y, t) = \left( 1 + \exp \left( \frac{x + y - t}{2\nu} \right) \right)^{-1}.$$  

As in [11], we denote norm infinity and norm relative of errors as follow

$$E_{L^\infty} = \max_{1 \leq i \leq N-\xi_0-K_0} |u^{(i)} - u_N^{(i)}|, \quad E_{rel} = \left( \sum_{i=1}^{N-\xi_0-K_0} \left( u^{(i)} - u_N^{(i)} \right)^2 \right)^{\frac{1}{2}},$$
where \( u^{(i)} \) and \( u_N^{(i)} \) are the exact value and the computed value of the solution \( u \) at LGL point \( i \).

From the table 3 and the table 4, we list norm infinity and norm relative of errors about solving (12) by the ChSC method [11] and the method (8), respectively. In the time direction, Runge–Kutta method of order four is used in [11]. We can find our method has better spatial accuracy at the same situation and can reach high precision when \( N \) is large enough and \( t \) is small enough.

**Table 3.** Norm infinity and norm relative of errors in [11].

| \( N \) | \( t \) | \( n \) | \( E_{it} \) \( T=0.05 \) | \( E_{it} \) \( T=0.25 \) | \( E_{rel} \) \( T=0.05 \) | \( E_{rel} \) \( T=0.25 \) |
|---|---|---|---|---|---|---|
| 5  | 0.0050 | 1.00 | 8.94e-8 | 1.19e-7 | 1.50e-7 | 1.70e-7 |
| 10 | 0.0005 | 1.00 | 7.45e-7 | 8.05e-7 | 8.50e-7 | 9.82e-7 |
| 10 | 0.0010 | 0.10 | 1.37e-6 | 2.09e-6 | 2.15e-6 | 2.63e-6 |
| 10 | 0.0050 | 0.10 | 1.28e-6 | 5.84e-6 | 2.49e-6 | 4.02e-6 |
| 30 | 0.0005 | 0.01 | 4.14e-5 | 4.32e-3 | 4.65e-5 | 2.26e-3 |

**Table 4.** Norm infinity and norm relative of errors of the method (8) \( (M = 1) \).

| \( N \) | \( t \) | \( n \) | \( E_{it} \) \( T=0.05 \) | \( E_{it} \) \( T=0.25 \) | \( E_{rel} \) \( T=0.05 \) | \( E_{rel} \) \( T=0.25 \) |
|---|---|---|---|---|---|---|
| 5  | 0.0050 | 1.00 | 8.70e-9 | 6.42e-9 | 7.19e-9 | 7.69e-9 |
| 10 | 0.0005 | 1.00 | 8.29e-11 | 6.67e-11 | 7.35e-11 | 7.86e-11 |
| 10 | 0.0010 | 0.10 | 4.98e-7 | 1.41e-6 | 5.01e-7 | 1.34e-6 |
| 16 | 0.0005 | 0.10 | 6.38e-8 | 1.62e-7 | 8.45e-8 | 1.60e-7 |
| 64 | 0.0001 | 0.01 | 9.28e-7 | 3.15e-5 | 5.50e-7 | 1.39e-5 |

### 4. Conclusion

In this paper, a triangular spectral element method for the two-dimensional viscous Burgers equation is developed. Some numerical examples show the effectiveness of this method. We compare it with other method to show the superiority. We will continue the above research, verify the stability and convergence of this method. And we are about to apply the method to discontinuous problem.

**References**

[1] Shen J, Wang L L and Li H Y. 2009 A triangular spectral element method using fully tensorial rational basis functions *SIAM J. Numer. Anal* **47** (3) 1619-1650.

[2] Chen L Z, Shen J and Xu C J. 2011 A triangular spectral method for the Stokes equations *Numer. Math.-theory Me* **4** (2) 158-179.

[3] Pasquetti R and Rapetti F 2010 Spectral element methods on unstructured meshes: which interpolation points? *Numer Algorithms* **55** (2-3) 349-366.

[4] Li Y Y, Wang L L, Li H Y and Ma H P. 2011 A new spectral method on triangles *J. Comput. Methods Sci. Eng* **76** 237-246.

[5] Duffy M G 1982 Quadrature over a pyramid or cube of integrands with a singularity at a vertex *SIAM J. Numer. Anal* **19** (6) 1260-1262.

[6] Samson M, Li H Y and Wang L L 2013 A new triangular spectral element method I: implementation and analysis on a triangle *Numer. Algorithms* **64** (3) 519-547.

[7] Zhou B Z, Wang B, Wang L L and Xie Z Q 2019 A new triangular spectral element method II: mixed formulation and hp-error estimates *Numer. Math.-theory Me* **12** (1) 72-97.
[8] Burger J M 1949 A mathematical model illustrating the theory of turbulence *Adv. Appl. Mech* 1 (2) 171–175.

[9] Lees M, Forsythe G E and Wasow W R 1961 Finite-difference methods for partial differential equations *AM Math. Mo* 68 (8) 826.

[10] Li J L, Ma H P and Li H Y 2018 A triangular spectral element method for elliptic and Stokes problems *Appl. Math. Comput* 321 195-208.

[11] Khater A H, Temsah R S and Hassan M M 2008 A Chebyshev spectral collocation method for solving Burgers’-type equations *J. Comput. Appl. Math* 222 (2) 333-350.

[12] Yang X, Ge Y and Lan B 2021 A class of compact finite difference schemes for solving the 2D and 3D Burgers’ equations *Math. Comput. Simulat* 185 (2) 510-524.