Theory of Approximation and Applications

Vol. 13, No.1, (2019), 27-42

Numerical Solution of the Burgers' Equation Based on Sinc Method

Gh. Kazemi Gelian [a,*], M. A. Fariborzi Araghi [b]

[a] *Department of Mathematics, Shirvan Branch, Islamic Azad University, Shirvan, Iran.*

[b] *Department of Mathematics, Central Tehran Branch, Islamic Azad University, P.O. Box 13185.768, Tehran, Iran.*

Received 12 March 2011; accepted 19 April 2012

Abstract

In this paper, numerical solution of Burgers'equation is considered by applying Sinc method. For this purpose, we apply Sinc method in cooperative with a classic finite difference formula to Burgers'equation. Numerical examples are provided to verify the validity of proposed method.

Key words: Burgers' equation, Sinc method, single exponential transformation.

2010 AMS Mathematics Subject Classification : 34A12, 34A30, 34Bxx, 34Lxx.

* Corresponding author's E-mail: kazemigelian@yahoo.com.

1 Introduction

For a given velocity u and viscosity coefficient v the general form of Burgers' equation (also known as viscous Burgers' equation) is [4]:

$$u_t + uu_x = vu_{xx}, \quad a < x < b, \ t > 0 \tag{1.1}$$

subject to the initial boundary conditions:

$$\begin{aligned} u(x,0) &= f(x), \quad a \leq x \leq b, \\ u(a,t) &= g_a(t), \quad t \geq 0, \\ u(b,t) &= g_b(t), \quad t \geq 0, \end{aligned} \tag{1.2}$$

When $v = 0$ the Burgers' equation becomes the inviscid Burgers' equation. Eq.(1.1) arises in various areas of applied mathematics, such as modeling of dynamics, heat conduction, and acoustic waves. Also, this equation has a large variety of applications in the modeling of water in unsaturated soil, dynamics of soil water, models of traffic, turbulence and fluid flow, mixing and turbulent diffusion [4,11].

Many researchers tried to find analytic and numerical solutions of Eq.(1.1) by using appropriate transformation to other known problems such as the Bäcklund transformation [9], Darboux transformation [10], sine - cosine methods [16], modified extended tanh-function method [12], the Hopf-Cole transformation [5].

Various numerical techniques specially based on variational iteration method, Adomian decomposition method, a finite-difference approach, mixed finite-difference and boundary element methods have been applied to solve Eq.(1.1). For further information see [1,3,10,15].

In [7], Kutluay et al. have obtained an approximate solution of the equation by the least-squares quadratic B-spline finite element method.

Sinc method is a powerful numerical tool for finding fast and accurate solution in various areas of problems. References [8,14], contain full discussion about sinc approximation.

The purpose of this paper is to extend the application of the sinc method for solving Burgers'equation by considering stability analysis of the method.

In this paper, we propose Sinc method for solving the Burgers' equation. The rest of this paper is organized as follows. In section 2, we present some required preliminaries for Sinc method. The proposed method is drawn in section 3. In section 4, stability analysis of the method is considered. Numerical examples are given in section 5.

2 Sinc definitions and preliminaries

Let f be a function defined on $\mathbb{R}$ and $h > 0$ is step size then the Whittaker cardinal defined by the series

$$C(f,h)(x) = \sum_{j=-\infty}^{\infty} f(jh)S(j,h)(x), \tag{2.1}$$

whenever this series convergence, and

$$S(j,h)(x) = \frac{\sin[\pi(x-jh)/h]}{\pi(x-jh)/h}, \quad j = 0, \pm 1, \pm 2, .. \tag{2.2}$$

where $S(j,h)(x)$ is known as $j-th$ Sinc function evaluated at x.

Throughout of this paper, let $d > 0$, and D_d denote the region $\{z = x + iy \quad |y| < d\}$ in the complex plan $\mathcal{C}$ and ϕ the conformal map of a simply connected domain D in the complex domain onto D_d, such that $\phi(a) = -\infty, \phi(b) = \infty$, where a, b are boundary points of D with $a, b \in \partial D$. Let ψ denote the inverse map of ϕ, and let the arc Γ, with end points $a, b \ (a, b \in \Gamma)$, given by $\Gamma = \psi(-\infty, \infty)$. For $h > 0$, let the points x_k on Γ given by $x_k = \psi(kh), \quad k \in Z$.

To construct Sinc approximation on the interval (a, b), which apply in this paper, the eye-shaped :

$$D_E = \{z = x + iy : |arg(\frac{z-a}{b-z})| < d \leq \frac{\pi}{2}\},$$

is mapped onto infinite strip D_d by $\phi(z) = \ln(\frac{z-a}{b-z})$. So, the basis Sinc function on (a, b) is given by :

$$S(j,h)o\phi(x) = Sinc(\frac{\phi(x) - jh}{h}).$$

where

$$Sinc(x) = \begin{cases} 1 \text{ if } x = 0 \\ \frac{\sin(\pi x)}{\pi x} \text{ if } x \neq 0, \end{cases}$$

and,

$$x_k = \phi^{-1}(kh) = \frac{a + b\exp(kh)}{1 + \exp(kh)}, \quad k = -N..N.$$

Definition 1. [8] Let $B(D_E)$ be the class of functions F which are analytical in D_E, satisfy

$$\int_{\psi(s+L)} |F(z)dz| \to 0, \quad s \to \pm\infty,$$

where $L = iv : |v| < d \leq \frac{\pi}{2}$, and

$$N(F) = \int_{\partial D_E} |F(z)dz| < \infty.$$

.

Theorem 1. [8] If $|F(x)| \leq Ce^{-\alpha|\phi(x)|}$, $x \in \Gamma$, for positive constants C, α be selecting $h = \sqrt{\frac{\pi d}{\alpha N}}$, we have the following interpolation relation:

$$|F(x) - \sum_{j=-N}^{N} F(x_j)S(j,h)o\phi(x)| \leq C\sqrt{N}e^{-\sqrt{\pi d\alpha N}}, \quad x \in \Gamma.$$

3 The Method

By discretizing time derivative of Eq.(1.1) with a classic finite difference formula and space derivative by the θ-weighted ($0 \leq \theta \leq 1$) scheme, between successive two time levels n and $n+1$ we have:

$$\begin{aligned} \frac{u^{n+1}-u^n}{\delta t} &+ \theta((uu_x)^{n+1} - v(u_{xx})^{n+1}) \\ &+ (1-\theta)((uu_x)^{n+1} - v(u_{xx})^{n+1}) = 0, \end{aligned} \tag{3.1}$$

where $u^{n+1} = u(x, t^{n+1}), t^{n+1} = t^n + \delta t$ and δt is a time step size with $n\delta t < T$ and $t \in (0, T)$. Also we have :

$$u^{n+1} = u(x, t^{n+1}) = \sum_{j=-N}^{N} u_j^{n+1} S(j, h) o \phi(x). \tag{3.2}$$

To have a better formula, we simplify terms $(uu_x)^{n+1}$ in (3.1), so

$$\begin{aligned} (uu_x)^{n+1} &= (uu_x)^n + \delta t\left(\frac{u^{n+1}-u^n}{\delta t} u_x^n + u^n \frac{(u_x)^{n+1} - (u_x)^n}{\delta t}\right) + O(\delta t^2) \\ &= u^{n+1}u_x^n + u^n(u_x)^{n+1} - u^n u_x^n + O(\delta t^2). \end{aligned} \tag{3.3}$$

By substituting (3.3) in (3.1) we obtain

$$\begin{aligned} u^{n+1} - u^n + &\theta\delta t\{u^{n+1}u_x^n + u^n(u_x)^{n+1} - u^n(u_x)^n - v(u_{xx})^{n+1}\} \\ &+ (1-\theta)\delta t\{(uu_x)^n - v(u_{xx})^n\} = 0. \end{aligned} \tag{3.4}$$

or

$$\begin{aligned} u^{n+1} &+ \theta\delta t\{u^{n+1}u_x^n + u^n(u_x)^{n+1} - v(u_{xx})^{n+1}\} \\ =&u^n + (uu_x)^n(2\theta\delta t - \delta t) + (1-\theta)\delta t v u_{xx})^n. \end{aligned} \tag{3.5}$$

To apply collocation method in order to get unknowns u_j^n in (3.5) we use

$2N+1$ Sinc grid points as

$$
\begin{aligned}
&x_{-N} = a,\\
&x_k = \phi^{-1}(kh) = \frac{a+b\exp(kh)}{1+\exp(kh)}, \quad k=-N+1..N-1,\\
&x_N = b.
\end{aligned} \tag{3.6}
$$

Furthermore, we have [8,14]

$$
u^{n+1}(x_k) = \sum_{j=-N}^{N} u_j^{n+1} S(j,h)o\phi(x)|_{x=x_k} = u_k^{n+1}. \tag{3.7}
$$

and

$$
u_x^{n+1}(x_k) = \sum_{j=-N}^{N} u_j^{n+1} \frac{d}{dx} S(j,h)o\phi(x)|_{x=x_k} = \frac{1}{h}\delta_{jk}^{(1)}\phi'(x_k). \tag{3.8}
$$

also,

$$
\begin{aligned}
u_{xx}^{n+1}(x_k) &= \textstyle\sum_{j=-N}^{N} u_j^{n+1} \frac{d^2}{dx^2} S(j,h)o\phi(x)|_{x=x_k}\\
&= \textstyle\frac{1}{h^2}\delta_{jk}^{(2)}(\phi'(x_k))^2 + \frac{1}{h}\delta_{jk}^{(1)}\phi''(x_k).
\end{aligned} \tag{3.9}
$$

so by replacing (3.9) in (3.5) we get:

$$
\begin{aligned}
&(\sum_{j=-N}^{N} u_j^{n+1} S(j,h)o\phi(x)) + \theta\delta t\{(\sum_{j=-N}^{N} u_j^{n+1} S(j,h)o\phi(x)) \\
&(\sum_{j=-N}^{N} u_j^{n+1} \frac{d}{dx} S(j,h)o\phi(x)) \\
&+(\sum_{j=-N}^{N} u_j^{n} S(j,h)o\phi(x))(\sum_{j=-N}^{N} u_j^{n+1} \frac{d}{dx} S(j,h)o\phi(x)) \\
&-v(\sum_{j=-N}^{N} u_j^{n+1} \frac{d^2}{dx^2} S(j,h)o\phi(x))\} \\
&=(\sum_{j=-N}^{N} u_j^{n} S(j,h)o\phi(x)) + (\sum_{j=-N}^{N} u_j^{n} S(j,h)o\phi(x)) \\
&(\sum_{j=-N}^{N} u_j^{n} \frac{d}{dx} S(j,h)o\phi(x))(2\theta\delta t - \delta t) \\
&+(1-\theta)\delta t v \sum_{j=-N}^{N} u_j^{n} \frac{d^2}{dx^2} S(j,h)o\phi(x), \quad k=-N+1..N-1.
\end{aligned} \tag{3.10}
$$

so we have the following system of equations at sinc pints

$$
\begin{aligned}
&(\sum_{j=-N}^{N} u_j^{n+1} \delta_{jk}^{(0)}) + \theta\delta t\{(\sum_{j=-N}^{N} u_j^{n+1} \delta_{jk}^{(0)})(\sum_{j=-N}^{N} u_j^{n+1} \frac{1}{h}\delta_{jk}^{(1)} \phi'(x_k)) \\
&+(\sum_{j=-N}^{N} u_j^{n} \delta_{jk}^{(0)})(\sum_{j=-N}^{N} u_j^{n+1} \frac{1}{h}\delta_{jk}^{(1)} \phi'(x_k)) \\
&-v(\sum_{j=-N}^{N} u_j^{n+1} \frac{1}{h^2}\delta_{ik}^{(2)} (\phi'(x_k))^2 + \frac{1}{h}\delta_{ik}^{(1)} \phi''(x_k))\} \\
&=(\sum_{j=-N}^{N} u_j^{n} \delta_{jk}^{(0)}) + (\sum_{j=-N}^{N} u_j^{n} \delta_{jk}^{(0)})(\sum_{j=-N}^{N} u_j^{n} \frac{1}{h}\delta_{jk}^{(1)} \phi'(x_k))(2\theta\delta t - \delta t) \\
&+(1-\theta)\delta t v(\sum_{j=-N}^{N} u_j^{n} \frac{1}{h^2}\delta_{ik}^{(2)} (\phi'(x_k))^2 + \frac{1}{h}\delta_{ik}^{(1)} \phi''(x_k)), \quad k=-N+1..N-1.
\end{aligned} \tag{3.11}
$$

Also for initial conditions we get:

$$\begin{aligned}\sum_{j=-N}^{N} u_j^{n+1} S(j,h)o\phi(x)|_{x=x_{-N}} &= u_1^{n+1}(x_{-N}) = g_a(t^{n+1}),\\ \sum_{j=-N}^{N} u_j^{n+1} S(j,h)o\phi(x)|_{x=x_{N}} &= u_1^{n+1}(x_{N}) = g_b(t^{n+1}).\end{aligned} \tag{3.12}$$

Numerical solution of Eq.(1.1), in the sinc points, can be approximated by

$$u^n(x_i) = \sum_{j=-N}^{N} u_j^n S(j,h)o\phi(x_i), \quad i=-N,..N. \tag{3.13}$$

Eq.(3.13) can be written in a matrix form as:

$$u^n = \mathbf{A}p^n, \tag{3.14}$$

where

$$\mathbf{A} = [A_{i,j} = S(j,h)o\phi(x_i)], \quad p^n = [u_{-N}^n, ..., u_N^n]^t. \tag{3.15}$$

The matrix $\mathbf{A}$ can be written in the following form [13]:

$$\mathbf{A} = \mathbf{A_d} + \mathbf{A_b}, \tag{3.16}$$

where

$$\mathbf{A_d} = \left\{ A_{i,j} = S(j,h)o\phi(x_i): \begin{array}{l} -N+1 \le i \le N-1, \\ -N \le j \le N \\ and\ 0\ elsewhere \end{array} \right\}$$
$$\mathbf{A_b} = \{A_{i,j} = S(j,h)o\phi(x_i): \ i=-N,N, \quad -N \le j \le N \ and\ 0\ elsewhere\} \tag{3.17}$$

For interior points $x_i \in (a,b)$ and boundary points $x_{-N} = a, x_N = b$ equations (3.11) and (3.12) can be written as:

$$\begin{aligned}&[\mathbf{A_d} + \mathbf{A_b} + \theta\delta t(\mathbf{D}+\mathbf{E}) - \theta\delta t v\mathbf{C}]p^{n+1}\\ &\quad = [\mathbf{A_d} + \mathbf{A_b} + \delta t(2\theta-1)\mathbf{E} + (1-\theta)\delta t v\mathbf{C}]p^n + \mathbf{F}^{n+1},\end{aligned} \tag{3.18}$$

where $\mathbf{B}$ and $\mathbf{C}$ are $(2N+1)(2N+1)$ matrices such that:

$$\mathbf{B} = \left\{ S'(j,h)o\phi(x_i) : \begin{array}{l} -N+1 \leq i \leq N-1, \\ -N \leq j \leq N \\ and\ 0\ elsewhere \end{array} \right\}$$
$$\mathbf{C} = \{S''(j,h)o\phi(x_i) : \ i = -N, N, \quad -N \leq j \leq N \ and\ 0\ elsewhere\} \tag{3.19}$$

and

$$\begin{aligned} &u_x^n = \mathbf{B}p^n, \\ &\mathbf{D} = u_x^n * \mathbf{A_d}, \\ &\mathbf{E} = u^n * \mathbf{B}, \\ &\mathbf{F}^{n+1} = [g_a(x_{-N}), ..., g_b(x_N)]^t. \end{aligned} \tag{3.20}$$

The symbol $*$ means that $i-th$ component of the vector u^n is multiplied to every element of $i-th$ row of matrix $\mathbf{B}$. Finally system (3.18) cab be shown in the following form

$$\mathbf{L}p^{n+1} = \mathbf{R}p^n + \mathbf{F}^{n+1}, \tag{3.21}$$

where

$$\begin{aligned} &\mathbf{L} = [\mathbf{A_d} + \mathbf{A_b} + \theta\delta t(\mathbf{D} + \mathbf{E}) - \theta\delta tv\mathbf{C}], \\ &\mathbf{R} = [\mathbf{A_d} + \mathbf{A_b} + \delta t(2\theta - 1)\mathbf{E} + (1-\theta)\delta tv\mathbf{C}]. \end{aligned} \tag{3.22}$$

4 Stability analysis

In this section, stability analysis of the method is considered. Following [13], Eq.(3.18) can be linearized by assuming u in the nonlinear term uu_x as locally constant. Suppose the error at $n-th$ time level between exact and approximate solution is given by:

$$e^n = u^n_{Exact} - u^n_{Approximate} \tag{4.1}$$

The error equation for the linearized Eq.(3.18) can be written as:

$$[H + \delta t\theta K]e^{n+1} = [H - \delta t(1-\theta t)K]e^{n}, \tag{4.2}$$

where

$$H = [\mathbf{A_d} + \mathbf{A_b}]\mathbf{A^{-1}} = I, \quad K = [\mathbf{D} + \mathbf{E} - v\mathbf{C}]\mathbf{A^{-1}}. \tag{4.3}$$

So we have

$$e^{n+1} = We^{n}, \quad W = [I + \delta t\theta K]^{-1}[I - \delta t(1-\theta)K]. \tag{4.4}$$

To numerical method be stable we must have $\|W\|_2 \leq 1$, consequently, $\rho(W)$, spectral radius of the matrix W, must be $\rho(W) \leq 1$. It results that :

$$|\frac{1 - \delta t(1-\theta)\lambda_K}{1 + \delta\theta\lambda_K}| \leq 1, \tag{4.5}$$

where λ_K is are eigenvalue of the matrix K.
For $\theta = 0.5$, the inequality (4.5) becomes

$$|\frac{1 - 0.5\delta t\lambda_K}{1 + 0.5\delta t\lambda_K}| \leq 1, \tag{4.6}$$

In the case of complex eigenvalue $\lambda_K = a_k + ib_k$, where a_k, b_k are real numbers, inequality (4.6) results

$$|\frac{(1 - 0.5\delta t a_k) - i(0.5\delta t b_k)}{(1 + 0.5\delta t a_k) + i(0.5\delta t b_k)}| \leq 1, \quad \Rightarrow a_k \geq 0. \tag{4.7}$$

In the case of real eigenvalue of λ_K, inequality (4.6) holds.
For $\theta = 0$, the inequality (4.5) becomes

$$|1 - \delta t\lambda_K| \leq 1, \quad \Rightarrow \quad \delta t \leq \frac{2}{\lambda_K}, \quad \lambda_K \geq 0. \tag{4.8}$$

Thus for $\theta = 0$ the scheme is conditionally stable. The stability of scheme for the other values can be investigate in a similar manner. The stability of the scheme and conditioning of the component matrices H, K of the matrix W depend on the weight parameter and the minimum distance between any two collocation points h in the domain set $[a, b]$.

5 Numerical Examples

At first, we give the following algorithm to compute numerical solution of Eq.(1.1):

Algorithm1:

Step1: Input $a, b, \delta t, N, \theta, \alpha, f(x), g_a(t), g_b(t), \phi(t)$

Step2: Set $n := 0$ calculate p^0 and $n := 1$.

Step3: While $n\delta t < T$ do the following statements

Calculate Matrices $\mathbf{L}, \mathbf{R}, \mathbf{F}$

Solve system (3.21) and obtain p^n, which is approximated solution at time level n.

$n := n + 1$.

end while

In this section, based on algorithm 1, two examples are presented to illustrate the effectiveness and importance of proposed method. All programs have been provided by Maple 13.

Example .1 We consider the following Burgers' equation [6,7]

$$\begin{aligned} &u_t + uu_x = vu_{xx}, \quad 0 < x < 1,\ t > 0 \\ &u(x,0) = \sin(\pi x), \quad 0 \le x \le 1, \\ &u(0,t) = 0, \quad t \ge 0, \\ &u(1,t) = 0, \quad t \ge 0, \end{aligned} \tag{5.1}$$

The exact Fourier series solution of this problem given by Cole [7] is:

$$u(x,t) = 2\pi v \frac{\sum_{n=1}^{\infty} a_n \exp(-n^2\pi^2 vt) n \sin(n\pi x)}{a_0 + \sum_{n=1}^{\infty} a_n \exp(-n^2\pi^2 vt)\cos(n\pi x)} \tag{5.2}$$

where the Fourier coefficients are :

$$\begin{aligned} a_0 &= \int_0^1 \exp\{-(2\pi v)^{-1}(1 - \cos(\pi x)\} dx, \\ a_n &= 2\int_0^1 \exp\{-(2\pi v)^{-1}(1 - \cos(\pi x)\} \cos(n\pi x) dx, \quad n = 1, 2, 3, ... \end{aligned} \tag{5.3}$$

To obtain results, we use Maple 13 package and take $x = 0.2, 0.4, 0.6, 0.8$ with $\delta = 0.01, v = 10$ by $N = 7$. The results are shown in Table 1.

x	$Approximate$	$Exact$
0.2	0.21742	0.21816
0.4	0.35387	0.35390
0.6	0.35503	0.35502
0.8	0.219867	0.21997

Table 1: Results of example 1 by Sinc method with $N = 7$ and $\delta = 0.01, v = 10$.

From these results, we conclude that the proposed method, to calculate the approximate numerical solution of the Burgers'equation, gives remarkable accuracy in comparison with the exact solution for some values of x.
In order to show relation between x and δt, we take different value $\delta t = 0.4, 0.6, 0.8$ and obtain approximate solution at $x = 0.25, 0.50, 0.75$. Based on result in table 2, the approximate solution is in good agreement with the exact solution. Also, the solutions are more accurate than the results in [6].

x	δt	$Approximate(Present\ Method)$	$Ref.[6]$	$Exact$
e	0.4	0.01358	0.01303	0.01357
0.25	0.6	0.00188	0.00178	0.00189
	0.8	0.00026		0.00026
	0.4	0.01927	0.01853	0.01924
0.50	0.6	0.00266	0.00252	0.00267
	0.8	0.00033		0.00037
	0.4	0.01356	0.01308	0.01363
0.75	0.6	0.00187	0.00178	0.00189
	0.8	0.00027		0.00026

Table 2: Results of example 1 by Sinc collocation method with $v = 1, \delta t = 0.4, 0.6, 0.8$.

Table 3, shows numerical results of example 1 by Sinc collocation method for different values $v = 1, 0.1, 0.01, \delta = 0.6$.

x	$v = 1$	$v = 0.1$	$v = 0.01$
0.25	0.00188	0.24357	0.27537
0.50	0.00267	0.45087	0.53532
0.75	0.00187	0.49267	0.77147

Table 3: Numerical results of example 1 by Sinc collocation method with $v = 1, 0.1, 0.01, \delta = 0.6$.

Example 2 We consider the following Burgers' equation [6,7]

$$\begin{aligned} &u_t + uu_x = vu_{xx}, \quad 0 < x < 1,\ t > 0 \\ &u(x,0) = 4x(1-x), \quad 0 \le x \le 1, \\ &u(0,t) = 0, \quad t \ge 0, \\ &u(1,t) = 0, \quad t \ge 0, \end{aligned} \tag{5.4}$$

The exact solution of this problem is given by [7] where

$$\begin{aligned} &a_0 = \int_0^1 \exp\{-x^2(3v)^{-1}(3-2x)\}dx, \\ &a_n = 2\int_0^1 \exp\{-x^2(3v)^{-1}(3-2x)\}\cos(n\pi x)dx, \quad n = 1,2,3,... \end{aligned} \tag{5.5}$$

Table 4, shows numerical solution for $x = 0.2, 0.4, 0.6, 0.8$ between exact and approximate solution.

x	*Approximate*	*Exact*
0.2	0.22514	0.22514
0.4	0.36524	0.36522
0.6	0.36648	0.36641
0.8	0.22741	0.22706

Table 4: Results of example 1 by Sinc method with $N = 7$ and

$$\delta = 0.01, v = 10.$$

Table 5 displays numerical solutions of example 2 for different values of δt. It is observed that the numerical solutions are seen to be satisfactorily in agreement with the exact ones.
Figure 1, shows approximate solutions with $\delta t = 0.1, 0.2, 0.4, v = 0.01$ for example 2.
Figure 2, shows approximate solutions with $v = 0.1, 0.01, 0.001$.

x	δt	*Approximate*	*Exact*
0.25	0.4	0.01402	0.01400
	0.6	0.00183	0.00195
	0.8	0.00023	0.00027
0.50	0.4	0.01978	0.01985
	0.6	0.00267	0.00276
	0.8	0.00034	0.00038
0.75	0.4	0.01421	0.01407
	0.6	0.00197	0.00195
	0.8	0.00023	0.00027

Table 5: Results of example 2 by Sinc collocation method with $N = 7$ and $\delta = 0.01, v = 10$.

6 Conclusion

In this paper, we applied the Sinc collocation method on the Burgers' equations. The results show that the Sinc method is a powerful mathematical tool to solve Burgers' equation. It is also a promising method to solve other nonlinear equations. The solutions obtained are shown graphically. In addition this method is portable to other area of problems and easy to programming.

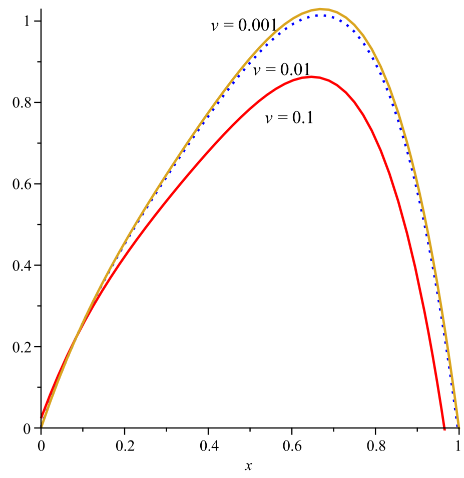

Fig. 1. Approximate solutions with $v = 0.1, 0.01, 0.001$ for example 2.

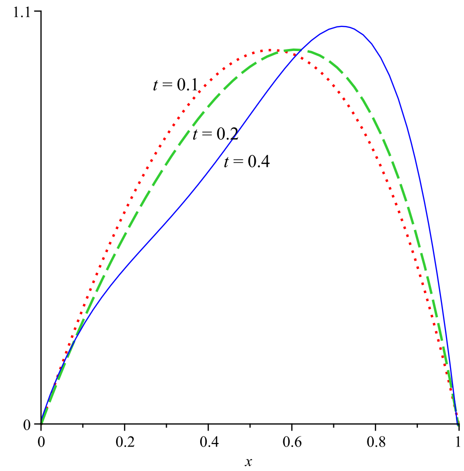

Fig. 2. Approximate solutions with $t = 0.1, 0.2, 0.4, v = 0.01$ for example 2.

References

[1] S. Abbasbandy, M. T. Darvishi, A numerical solution of Burgers'equation by modified Adomian method, Appl. Math. Comput, 163 (2005)1265-1272.

[2] M. A. Abdou, A. A. Soliman, Variational iteration method for solving Burgers and coupled Burgers' equations, J. Comput. Appl. Math. 181 (2005) 245-251.

[3] R. Bahadir, M. Saglam, A mixed finite difference and boundary element approach to one-dimensional Burgers' equation, Appl. Math. Comput, 160 (2005)663-673.

[4] J. M. Burgers, A mathematical model illustrating the theory of turbulence, Adv. Appl. Mech, 1 (1948) 171-199.

[5] E. Hopf, The partial differential equation $U_t + UU_x = cU_{xx}$, Commun. Pure Appl. Math, 3 (1950) 201-230.

[6] D. Khojasteh Salkuyeh, F. Saadati Sharafeh, On the numerical solution of the Burgers' equation, International Journal of Computer Mathematics 86 (2009)1334-1344.

[7] S. Kutluaya, A. Esena, I. Dagb, Numerical solutions of the Burgers' equation by the least-squares quadratic B-spline finite element method, J. Comput.Appl. Math, 167 (2004) 21-33.

[8] J. Lund, K. Bowers, Sinc Methods for Quadrature and Differential Equations, SIAM, Philadelphia, 1992.

[9] Z. S. Lü, An explicit Bäcklund transformation of Burgers' equation with applications, Commun. Theor. Phys, 44 (2005) 987-989.

[10] V. A. Matveev, M. A. Salle, Darboux Transformation and Solitons, Springer-Verlag, Berlin, Heidelberg, 1991.

[11] Özis, E. N. Aksan, A finite element approach for solution of Burgers'equation, Appl. Math. Comput, 139 (2003) 417-428.

[12] A. A. Soliman, The modified extended tanh-function method for solving Burgers-type equations, Physica A, 361 (2006) 394-404.

[13] Siraj-ul-Islam, Sirajul Haq, Arshed Ali, Meshfree method for the numerical solution of the RLW equation, J. Comput. Appl. Math, 223 (2009) 997-1012.

[14] F. Stenger, Numerical Methods Based on Sinc and Analytic Functions, Springer, 1993.

[15] J. W. Thomas, Numerical Partial Differential Equations: Finite Difference Methods, Springer-Verlag, New York, 1995.

[16] A. M. Wazwaz, The tanh and the sine-cosine methods for a reliable treatment of the modified equal width equation and its variants, Commun, Nonlinear Sci. Numer. Simul, 11 (2006) 148-160.