Available online at www.tjnsa.com
J. Nonlinear Sci. Appl. 9 (2016), 3727–3734
Research Article

Journal of Nonlinear Science and Applications
Print: ISSN 2008-1898 Online: ISSN 2008-1901

Approximations for Burgers' equations with C-N scheme and RBF collocation methods

Huantian Xie*, Jianwei Zhou, Ziwu Jiang, Xiaoyi Guo

School of Science, Linyi University, Linyi 276005, P. R. China.

Communicated by X.-J. Yang

Abstract

The Burgers' equation is one of the typical nonlinear evolutionary partial differential equations. In this paper, a mesh-free method is proposed to solve the Burgers' equation using the finite difference and collocation methods. With the temporal discretization of the equation using C-N scheme, the solution is approximated spatially by Radial Basis Function (RBF). The numerical results of two different examples indicate the high accuracy and flexibility of the presented method.

Keywords: Burgers' equation, collocation, Crank-Nicholson (C-N) scheme, multiquadric (MQ).
2010 MSC: 35L72, 65N35.

1. Introduction

In this paper, we consider the following nonlinear evolutionary partial differential equation:

$$\frac{\partial u(x,t)}{\partial t} + u(x,t)\frac{\partial u(x,t)}{\partial x} = \frac{1}{R}\frac{\partial^2 u(x,t)}{\partial x^2}, \tag{1.1}$$

where $R > 0$, interpreted as Reynolds number.

This equation is widely termed Burgers' equation, because it was treated by Burgers [4, 5] as a mathematical model for free turbulence after it was first introduced by Bateman [2]. Burgers' equation has been studied by many researchers for the following reasons:

(1) It contains the simplest form of nonlinear advection term uu_x and dissipation term u_{xx}/R for simulating the physical phenomena of wave motion[1, 17, 20–23, 25].

*Corresponding author

Email addresses: xiehuantian@163.com (Huantian Xie), zhoujianwei@lyu.edu.cn (Jianwei Zhou), jiangziwu@lyu.edu.cn (Ziwu Jiang), guoxiaoyi@lyu.edu.cn (Xiaoyi Guo)

Received 2015-09-17

(2) It can be solved analytically by exp-function method, variational iteration method and homotopic perturbation method[6, 14] so that numerical comparison can be made.

(3) Its shock wave behavior when the Reynolds number R is large. To date, the development of an innovative and robust numerical method for seeking accurate and efficient numerical solutions of Burgers' equation with large values of R, remains as a challenging task.

Various numerical methods were employed to obtain the solution of Burgers' equation and the solution methodologies commonly fall into the following classes: finite difference method [3, 10], finite element method [18], spectral methods and meshless methods [9, 12]. Recently, there has been an increasing interest in approximating the solutions of PDEs ([7, 11–13, 19, 24]) using Kansa method, one of the domain-type meshless methods, developed by Kansa [15, 16] in 1990. It is obtained by collocating the RBFs, particularly the multiquadric (MQ), for the numerical approximation of the solution. In contrast to the traditional meshed-based methods, the RBF collocation methods are mathematically simple and truly meshless. In this paper, a new numerical method is proposed to solve the Burgers' equation based on the finite difference and collocation method. The C-N scheme is employed for the temporal discretization of the equation, meanwhile, the numerical solution is approximated by the MQ radial basis function.

The paper is organized as follows: In Section 2, the radial basis functions approximation method is described. In Section 3, the method is applied to Burgers' equation with its temporal discretization by C-N scheme. The numerical experiments are presented in Section 4. Finally, a brief conclusion is summarized in Section 5.

2. Radial basis function approximation

The approximation of a distribution $u(\mathbf{x})$, using radial basis functions, may be written as a linear combination of N radial functions, usually it takes the following form

$$u(\mathbf{x}) \simeq \sum_{j=1}^{N} \lambda_j \varphi(\mathbf{x}, \mathbf{x}_j) + \psi(\mathbf{x}), \quad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \tag{2.1}$$

where N is the number of data points, $\mathbf{x} = (x_1, x_2, ..., x_d)$, d is the dimension of the problem, λ_j is the coefficient to be determined, φ is the radial basis function, and ψ is an additional polynomial. In the case without the additional polynomial, φ must be strictly positive definite to guarantee the solvability of the resulting system (e.g., Gaussian or inverse-multiquadrics). However, ψ is usually required if φ is conditionally positive definite, i.e., φ has a polynomial growth toward infinity, for example, thin plate splines and multiquadrics. Moreover, the polynomial in (2.1) is added for a special proof of nonsingularity of the extended interpolation system. Due to its good accuracy, multiquadrics will be used as radial basis function for the numerical scheme introduced in Section 3, which is defined as

$$\varphi(\mathbf{x}, \mathbf{x}_j) = \varphi(r_j) = \sqrt{r_j^2 + c^2}, \tag{2.2}$$

where $r_j = \parallel \mathbf{x} - \mathbf{x}_j \parallel$ is the Euclidean norm of $\mathbf{x} - \mathbf{x}_j$.

Herein $\mathcal{P}_q^d$ denotes the space of d-variate polynomials of order not exceeding q, and letting the polynomials $P_1, ..., P_m$ be the bases, then the polynomial $\psi(\mathbf{x})$, in (2.1), can be stated with

$$\psi(\mathbf{x}) = \sum_{i=1}^{m} \xi_i P_i(\mathbf{x}), \tag{2.3}$$

where $m = (q - 1 + d)!/(d!(q - 1)!)$.

To determine the coefficients $(\lambda_1, ..., \lambda_N)$ and $(\xi_1, ..., \xi_m)$, the collocation method is used. However, in addition to the N equations resulting from collocating (2.1) at the N points, extra m equations are required. This is insured by the m conditions for (2.1), *i.e.*,

$$\sum_{j=1}^{N} \xi_i P_i(\mathbf{x}_j) = 0,\ \ i = 1, ..., m. \tag{2.4}$$

In a similar representation as equation (2.1), for any linear partial differential operator $\mathcal{L}$, $\mathcal{L}u$ can be approximated by

$$\mathcal{L}u(\mathbf{x}) \simeq \sum_{j=1}^{N} \mathcal{L}\lambda_j \varphi(\mathbf{x}, \mathbf{x}_j) + \mathcal{L}\psi(\mathbf{x}). \tag{2.5}$$

3. Meshless method for the Burgers' equation

In this section, we consider the second-order nonlinear Burgers' equation

$$\frac{\partial u(x,t)}{\partial t} + u(x,t)\frac{\partial u(x,t)}{\partial x} = \varepsilon \frac{\partial^2 u(x,t)}{\partial x^2}, \qquad x \in [a,b] \subset \mathbb{R}, \quad t > 0, \tag{3.1}$$

with the initial and boundary conditions

$$u(x,0) = u_0(x), \tag{3.2}$$

$$u(a,t) = f_1(t), \quad u(b,t) = f_2(t), \quad t > 0, \tag{3.3}$$

where $\varepsilon = \frac{1}{R} > 0$, and $u_0(x), f_1(t), f_2(t)$ are given functions.

Equation (3.1) is discretized according to the following θ-weighted scheme

$$\frac{u^{n+1} - u^n}{\tau} + [\theta u^{n+1}\nabla u^{n+1} + (1-\theta)u^n \nabla u^n] = \varepsilon[\theta \nabla^2 u^{n+1} + (1-\theta)\nabla^2 u^n], \tag{3.4}$$

where ∇ is the gradient operator, $0 \le \theta \le 1$, $u^n = u(x, t^n)$, $t^n = t^{n-1} + \tau$ and τ is the time step size.

Moreover, the nonlinear term is linearized by

$$u^{n+1}\nabla u^{n+1} = u^{n+1}\nabla u^n + u^n \nabla u^{n+1} - u^n \nabla u^n. \tag{3.5}$$

Substituting (3.5) into (3.4), we obtain

$$u^{n+1} - \tau\varepsilon\theta\nabla^2 u^{n+1} + \tau\theta[u^{n+1}\nabla u^n + u^n\nabla u^{n+1}] = u^n + \tau(2\theta - 1)u^n\nabla u^n + \tau\varepsilon(1-\theta)\nabla^2 u^n. \tag{3.6}$$

Assuming that there are $N-2$ interpolation points, $u(x, t^n)$ can be approximated by

$$u^n(x) \simeq \sum_{j=1}^{N-2} \lambda_j^n \varphi(r_j) + \lambda_{N-1}^n x + \lambda_N^n. \tag{3.7}$$

To determine the interpolation coefficients $(\lambda_1, \lambda_2, ..., \lambda_{N-1}, \lambda_N)$, the collocation method is used by applying (3.6) at point $x_i, i = 1, 2, ..., N-2$. Thus, we have

$$u^n(x_i) \simeq \sum_{j=1}^{N-2} \lambda_j^n \varphi(r_{ij}) + \lambda_{N-1}^n x_i + \lambda_N^n, \tag{3.8}$$

where $r_{ij} = \sqrt{(x_i - x_j)^2}$. The additional conditions due to (2.4) are written as

$$\sum_{j=1}^{N-2} \lambda_j^n = \sum_{j=1}^{N-2} \lambda_j^n x_j = 0. \tag{3.9}$$

Combining (3.8) with (3.9) in a matrix form, we have

$$[u]^n = \mathbf{A}[\lambda]^n, \tag{3.10}$$

where $[u]^n = [u_1^n, \ \cdots\ , \ u_{N-2}^n\ 0\ 0]^T, [\lambda]^n = [\lambda_1^n, \ \cdots\ , \ \lambda_N^n]^T$ and $\mathbf{A} = [a_{ij}, 1 \le i, j \le N]$.

There are $p = N - 4$ internal (domain) points and two boundary points. Therefore, the $(N \times N)$ matrix $\mathbf{A}$ can be split into $\mathbf{A} = \mathbf{A}_d + \mathbf{A}_b + \mathbf{A}_e$, where

- $\mathbf{A}_d = [a_{ij}$ for $(2 \le i \le N-3, 1 \le j \le N)$ and 0 elsewhere];
- $\mathbf{A}_b = [a_{ij}$ for $(i = 1, N-2, 1 \le j \le N)$ and 0 elsewhere];
- $\mathbf{A}_e = [a_{ij}$ for $(N-1 \le i \le N, 1 \le j \le N)$ and 0 elsewhere].

Using the notation $\mathcal{L}\mathbf{A}$ to designate the matrix of the same dimension as $\mathbf{A}$ and containing the elements $\widetilde{a}_{ij} = \mathcal{L}a_{ij},\ 1 \le i, j \le N$, then (3.6) together with (3.3) can be stated, in the matrix form, as

$$\begin{aligned}&\{\mathbf{A}_d - \tau\varepsilon\theta\nabla^2\mathbf{A}_d + \tau\theta[diag(\mathbf{A}_d[\lambda]^n)\nabla\mathbf{A}_d + diag(\nabla\mathbf{A}_d[\lambda]^n)\mathbf{A}_d] + \mathbf{A}_b\}[\lambda]^{n+1}\\ &= \mathbf{A}_d[\lambda]^n + \tau\varepsilon(1-\theta)\nabla^2\mathbf{A}_d[\lambda]^n + \tau(2\theta-1)(\mathbf{A}_d[\lambda]^n).*(\nabla\mathbf{A}_d[\lambda]^n) + [F]^{n+1},\end{aligned} \tag{3.11}$$

where $diag(\mathbf{A}_d[\lambda]^n)$ is a diagonal matrix with $\mathbf{A}_d[\lambda]^n$ as its main diagonal and $[F]^n = [f_1^n\ 0\ \cdots\ 0\ f_{N-2}^n\ 0\ 0]^T$.

In (3.11), the accent $''.*''$ means component by component multiplication of two vectors. Equation (3.11) is obtained by combining (3.6), which applies to the domain points, while (3.3) applies to the boundary points.

Using (3.9) and the initial condition, represented by (3.2), $[\lambda]^0$ can be computed. Then (3.11) and (3.9) lead to $[u]^n$'s.

Remark 3.1. When choosing $N-2$ internal (domain) points and two boundary points, $[u]^n$ can also be got without using (3.9) [12].

Remark 3.2. Although (3.11) is valid for any value of $\theta \in$[0, 1], we will use $\theta = 1/2$ (the famous Crank-Nicholson scheme).

4. Numerical experiments

In this section, numerical results of our method for the Burgers' equation are presented. We use two different problems to show the accuracy and flexibility of the proposed method. In order to evaluate the numerical errors, we adopt three kinds of norms as defined by

$$L_\infty = \max_j |u_j - U_j|, \quad L_2 = \sqrt{\sum_{j=1}^{N} |u_j - U_j|^2}\ , \quad RMS = \sqrt{\frac{1}{N}\sum_{j=1}^{N} |u_j - U_j|^2}\ ,$$

where $u_j = u(x_j, T)$ is the exact analytical solution, and U_j is the numerical solution of u_j.

Example 4.1. In this example, we consider the second-order nonlinear Burgers' equation (1.1) with a large Reynolds number $R = 10000$, which has a monotone increasing solution. The analytical solution is given in [8] as

$$u(x,t) = \frac{\varepsilon}{1+\varepsilon t}(x + tan(\frac{x}{2(1+\varepsilon t)})). \tag{4.1}$$

The solution, evaluated at $t = 0$, is used as the initial condition, and the boundary functions are taken from the exact solution at $x = \pm 3$, respectively.

Table 1: L_∞, L_2 and RMS errors, with $\tau = 0.1, d_x = 0.12, x \in [-3, 3]$.

T	L_∞	L_2	RMS	c
1	2.1296×10^{-5}	3.2324×10^{-5}	4.5713×10^{-6}	3.72
2	1.1971×10^{-5}	2.7778×10^{-5}	3.9285×10^{-6}	3.55
2	1.7986×10^{-5}	3.4930×10^{-5}	4.9398×10^{-6}	3.54
4	2.4114×10^{-5}	5.5027×10^{-5}	7.7819×10^{-6}	3.75
5	3.0091×10^{-5}	5.0920×10^{-5}	7.2012×10^{-6}	3.60

All the computations were performed on Pentium(R) Dual-Core, 2.10 GHZ CPU and 2 GB of RAM. In Table 1, the L_∞, L_2, Root-Mean-Square (RMS) of errors between the exact and numerical solutions confirm the high accuracy of our method, and the shape parameter c found experimentally are also listed for $T = 1, 2, 3, 4$ and 5. In Figure 1, the graph of exact and numerical solutions for $T = 5$ and the absolute error graph are shown simultaneously. The space-time graph of the numerical solution up to $T = 5$ is also depicted in Figure 2.

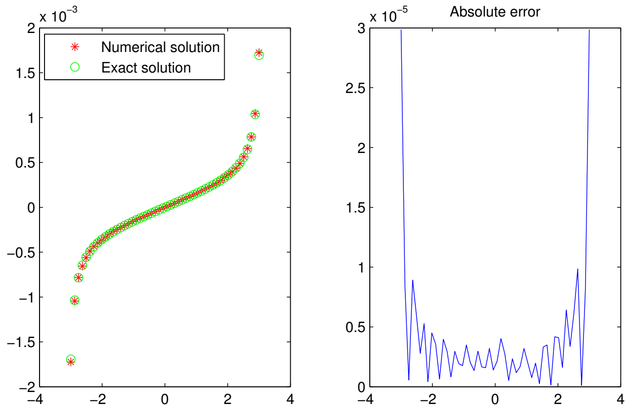

Figure 1: Exact and numerical solutions, and absolute error at $T = 5$ with $\tau = 0.1,\ d_x = 0.12, x \in [-3, 3]$.

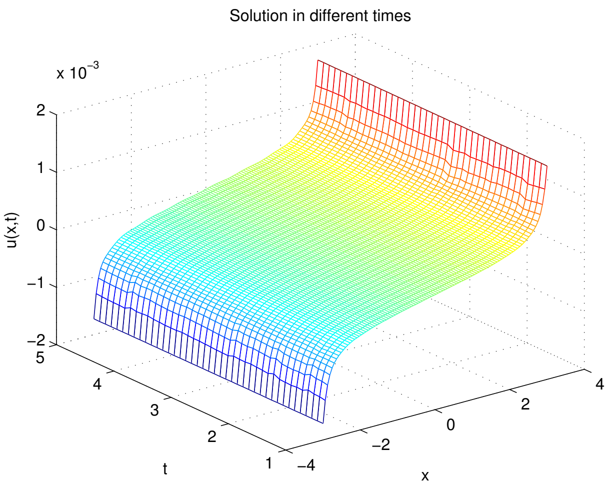

Figure 2: Space-time graph of the solution up to $T = 5$ with $\tau = 0.1, d_x = 0.12, x \in [-3, 3]$.

Example 4.2. The nonlinear Burgers' Equation with Reynolds number $R = 10$ is considered and it has a monotone decreasing solution. The exact solution is given by

$$U(x,t) = \frac{0.1e^{-A} + 0.5e^{-B} + e^{-C}}{e^{-A} + e^{-B} + e^{-C}}, \tag{4.2}$$

where

$$\begin{aligned} A &= 0.05R(x - 0.5 + 4.95t), \\ B &= 0.25R(x - 0.5 + 0.75t), \\ C &= 0.5R(x - 0.375). \end{aligned}$$

The required initial and boundary functions are taken from the exact solution. The L_∞ and L_2 errors and RMS of errors at $T = 1, 2, 3, 4$ and 5, and the values of the shape parameter c are listed in Table 2, respectively. The graph of exact and numerical solutions and the absolute error graph for $T = 5$ are drawn in Figure 3, which also indicate the good accuracy of the presented method. The space-time graph of the numerical solution up to $T = 5$ is shown in Figure 4. The space-time graphs for the two problems show the temporal stability of the meshless method for the Burgers' equation.

Table 2: L_∞, L_2 and RMS errors, with $\tau = 0.1, d_x = 0.16, x \in [-4, 4]$.

T	L_∞	L_2	RMS	c
1	1.9311×10^{-4}	3.9249×10^{-3}	5.5506×10^{-4}	1.19
2	2.9140×10^{-3}	8.2943×10^{-3}	1.1730×10^{-3}	1.70
2	2.9268×10^{-3}	9.4732×10^{-3}	1.3397×10^{-3}	1.51
4	2.6951×10^{-3}	8.5127×10^{-3}	1.2039×10^{-3}	1.27
5	4.8280×10^{-5}	1.3387×10^{-2}	1.8932×10^{-3}	1.56

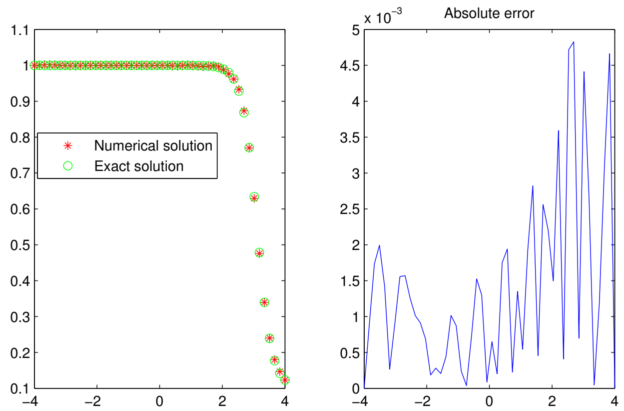

Figure 3: Exact and numerical solutions, and absolute error at $T = 5$ with $\tau = 0.1, d_x = 0.16, x \in [-4, 4]$.

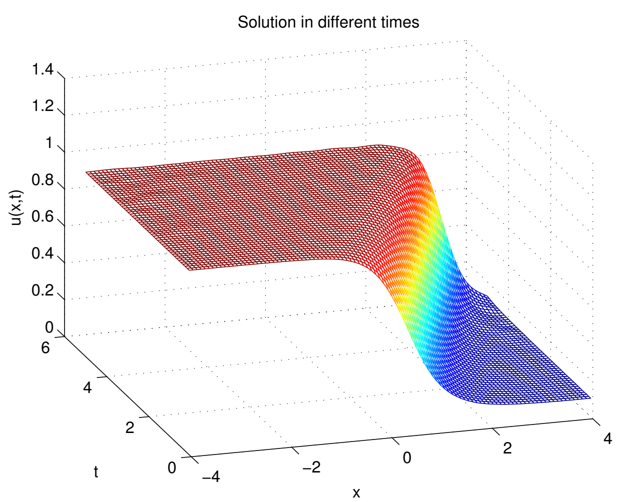

Figure 4: Space-time graph of the solution up to $T = 5$ with $\tau = 0.1, d_x = 0.12, x \in [-3, 3]$.

5. Conclusions

In this paper, we discussed one of the well-known nonlinear partial differential equations named the Burgers' equation. A meshless numerical method was proposed to solve the Burgers' equation based on the finite difference and RBF collocation. The C-N scheme was employed in the temporal discretization of the equation, and the numerical solution is approximated directly by the multiquadric (MQ) radial basis function. To demonstrate the accuracy and flexibility of this method, two different problems were considered in the numerical experiments, one with Reynolds number $R = 10000$ and a monotone increasing solution, and the other with Reynolds number $R = 10000$ and a monotone decreasing solution. The good performance of the presented method indicates the possibility of the extension of meshless method to other nonlinear partial differential equations.

Acknowledgements

The authors are grateful to the anonymous referee for their helpful suggestions and constructive comments. The work was partly supported by the National Natural Science Foundation of China (Grants nos. 11571157, 11402108 and 11301252), the Natural Science Foundation of Shandong Province (Grant no. BS2015DX012), AMEP of Linyi University and the Science Research Foundation for Doctoral Authorities of Linyi University(LYDX2015BS018).

References

[1] W. F. Ames, *Nonlinear Partial Differential Equations in Engineering*, Academic Press, New York-London, (1965). 1

[2] H. Bateman, *Some recent researches on the motion of fluids*, Mon. Weather Rev., **43** (1915), 163–170. 1

[3] H. P. Bhatt, A. Q. M. Khaliq, *Fourth-order compact schemes for the numerical simulation of coupled Burgers' equation*, Comput. Phys. Commun., **200** (2016), 117–138. 1

[4] J. M. Burgers, *Mathematical examples illustrating relations occurring in the theory of turbulent fluid motion*, Verh. Nederl. Akad. Wetensch. Afd. Natuurk. Sect., **17** (1939), 53 pages. 1

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

[6] J. D. Cole, *On a quasilinear parabolic equations occurring in aerodynamics*, Quart. Appl. Math., **9** (1951), 225–236. 1

[7] M. Dehghan, A. Shokri, *A numerical method for KdV equation using collocation and radial basis functions*, Nonlinear Dynam., **50** (2007), 111–120. 1

[8] D. J. Evans, A. R. Abdullah, *The group explicit method for the solution of Burgers' equation*, Computing, **32** (1984), 239–253. 4.1

[9] S. Haq, A. Hussain, M. Uddin, *On the numerical solution of nonlinear Burgers–type equations using meshless method of lines*, Appl. Math. Comput., **218** (2012), 6280–6290. 1

[10] I. A. Hassanein, A. A. Salama, H. A. Hosham, *Fourth-order finite difference method for solving Burgers' equation*, Appl. Math. Comput., **170** (2005), 781–800. 1

[11] Y.-C. Hon, K. F. Cheung, X.-Z. Mao, E. J. Kansa, *Multiquadric solution for shallow water equations*, J. Hydraul. Eng., **125** (1999), 524–533. 1

[12] Y.-C. Hon, X.-Z. Mao, *An efficient numerical scheme for Burgers' equation*, Appl. Math. Comput., **95** (1998), 37–50. 1, 3.1

[13] Y.-C. Hon, X.-Z. Mao, *A radial basis function method for solving options pricing model*, Financ. Eng., **8** (1999), 31–49. 1

[14] E. Hopf, *The partial differential equation $u_t + uu_x = \mu u_{xx}$*, Comm. Pure Appl. Math., **3** (1950), 201–230. 1

[15] E. J. Kansa, *Multiquadrics–a scattered data approximation scheme with applications to computational fluid dynamics-I*, Comput. Math. Appl., **19** (1990), 127–145. 1

[16] E. J. Kansa, *Multiquadrics–a scattered data approximation scheme with applications to computational fluid dynamics–II. Solution to parabolic, hyperbolic and elliptic partial differential equations*, Comput. Math. Appl., **19** (1990), 147–161. 1

[17] A. Kurt, Y. Cenesiz, O. Tasbozan, *On the solution of Burgers' equation with the new fractional derivative*, Open Phys., **13** (2015), 355–360. 1

[18] S. Kutluay, A. Esen, I. Dag, *Numerical solutions of the Burgers' equation by the least-squares quadratic B-spline finite element method*, J. Comput. Appl. Math., **167** (2004), 21–33. 1

[19] M. D. Marcozzi, S. Choi, C. S. Chen, *On the use of boundary conditions for variational formulations arising in financial mathematics*, Appl. Math. Comput., **124** (2001), 197–214. 1

[20] L. Xu, J.-H. Zhao, H. Liu, *Numerical simulation for the single-bubble electrospinning process*, Therm. Sci., **19** (2015), 1255–1259. 1

[21] X.-J. Yang, J. A. T. Machado, J. Hristov, *Nonlinear dynamics for local fractional Burgers' equation arising in fractal flow*, Nonlinear Dynam., **84** (2015), 3–7.

[22] X.-J. Yang, J. A. T. Machado, H. M. Srivastava, *A new numerical technique for solving the local fractional diffusion equation: two-dimensional extended differential transform approach*, Appl. Math. Comput., **274** (2016), 143–151.

[23] X.-J. Yang, H. M. Srivastava, C. Cattani, *Local fractional homotopy perturbation method for solving fractal partial differential equations arising in mathematical physics*, Rom. Rep. Phys., **67** (2015), 752–761. 1

[24] M. Zerroukat, H. Power, C. S. Chen, *A numerical method for heat transfer problem using collocation and radial basis functions*, Int. J. Numer. Methods Eng., **42** (1992), 1263–1278. 1

[25] D.-S. Zhang, D.-Z. Pan, H.-Y. Wang, W.-D. Shi, *Numerical prediction of cavitating flow around a hydrofoil using pans and improved shear stress transport k-omega model*, Therm. Sci., **19** (2015), 1211–1216. 1