Cubic B-spline quasi-interpolation and an application to numerical solution of generalized Burgers-Huxley equation

Lan-Yin Sun¹ and Chun-Gang Zhu²

Abstract
Nonlinear partial differential equations are widely studied in Applied Mathematics and Physics. The generalized Burgers-Huxley equations play important roles in different nonlinear physics mechanisms. In this paper, we develop a kind of cubic B-spline quasi-interpolation which is used to solve Burgers-Huxley equations. Firstly, the cubic B-spline quasi-interpolation is presented. Next we get the numerical scheme by using the derivative of the quasi-interpolation to approximate the spatial derivative of the dependent variable and modified Euler scheme to approximate the time derivative of the dependent variable. Moreover, the efficiency of the proposed method is illustrated by the agreement between the numerical solution and the analytical solution which indicate the numerical scheme is quite acceptable.

Keywords
Nonlinear physics mechanisms, Burgers-Huxley equation, numerical solution, cubic B-spline quasi-interpolation

Introduction
Nonlinear phenomena play a crucial role in various nonlinear fields of science which has undergone many studies.¹⁻⁵ It is known that various phenomena in scientific fields can be described by nonlinear partial differential equations. The Burgers-Huxley equations arise from the mathematical modeling of many nonlinear scientific phenomena.

Consider the following generalized Burgers-Huxley equation (1)

\[ \frac{\partial u}{\partial t} + \alpha u^\delta \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = \beta u(1-u^\delta)(u^\delta - \gamma), \quad (1) \]

where \( \alpha, \beta, \gamma \) and \( \delta \) are parameters, \( \beta \geq 0, \delta > 0, 0 < \gamma < 1 \). This equation describes the interaction between reaction mechanisms, convection effects and diffusion transport.⁶

- when \( \beta = 0, \delta = 1 \), equation (1) degenerates into the following Burgers equations

\[ \frac{\partial u}{\partial t} + \alpha u \frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} = 0. \quad (2) \]

This equation is a very important fluid dynamic model which has many applications in fields as gas dynamics, number theory, heat conduction, elasticity etc.

- when \( \alpha = 0, \delta = 1 \), equation (1) is reduced to the following Huxley equation which describes nerve

¹School of Mathematics and Statistics, Xinyang Normal University, Xinyang, P.R.China
²School of Mathematical Sciences, Dalian University of Technology, Dalian, P.R.China

Corresponding author:
Lan-Yin Sun, School of Mathematics and Statistics, Xinyang Normal University, No.237 Nanhu Road Shihe District, Xinyang, 464000, P.R.China.
Email: lysun@xynu.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
pulse propagation in nerve fibers and wall motion in liquid crystals,
\[
\frac{\partial u}{\partial t} + \frac{\partial^2 u}{\partial x^2} = \beta (1 - u(u - \gamma)).
\]
(3)

As we all know, the nonlinear diffusion equations (2) and (3) play an important role in nonlinear physics.

Since the Burgers’ equation was firstly discussed by Bateman in 1915, it had attracted many scholars’ attention.8–11 Hodgkin and Huxley12 used the Huxley equation to predict the quantitive behavior of a model nerve. The homotopy analysis method was presented to get the analytical solution of the Burgers-Huxley equation (2). In 2013, Kevorkian and Cole13 proposed a numerical scheme to solve the coupled highly dimensional nanofluid flow among the rotating circu-

The framework of the paper is organized as follows. Some preliminaries regarding B-spline quasi-interpolation are addressed in Sec.2. In Sec.3, the numerical scheme to solve the generalized Burgers-Huxley equation is proposed. The accuracy and efficiency of our method are verified with two numerical examples in Sec.4. Finally, the paper is completed with a conclusion.

Cubic B-spline quasi-interpolation

Given an interval \( I = [a, b] \), let \( S_d(x_n) \) denote the space of splines of degree \( d \) and \( C^{d-1} \) on the uniform partition \( x_n = \{x_i = a + ih, 0 \leq i \leq n\} \) with meshlength \( h = \frac{b-a}{n} \), where \( b = x_n \). With the following de-Boor-Cox formula,27
\[
B_{i,0}(x) = \begin{cases} 1, & u \in [x_i, x_{i+1}] \\ 0, & \text{otherwise} \end{cases}
\]
for \( d = 0 \) and
\[
B_{i,d}(x) = \begin{array}{c}
\frac{x-x_i}{x_i+d-x_i}B_{i,d-1}(x) \\
+ \frac{x_{i+d+1}-x}{x_{i+d+1}-x_{i+1}}B_{i+1,d-1}(x),
\end{array}
\quad x \in [x_i, x_{i+1}],
\]
(5)

for \( d \geq 1 \), the basis functions \( B_{j,d}(x), j = \{1, 2, \ldots, n + d\} \) of space \( S_d(x_n) \) can be presented. As usual, multiple knots \( a = x_0 = x_{-1} = \ldots = x_{-d} \) and \( b = x_n = x_{n+1} = \ldots = x_{n+d} \) are added at endpoints.

Univariate spline quasi-interpolations can be defined as operators of the form
\[
Qf(x) = \sum_{j \in J} \mu(j)B_{j,d}(x),
\]
where \( \{B_{j,d}(x), j \in J\} \) are the B-spline basis functions of \( S_d(x_n) \). We denote by \( \Pi_d \) the space of polynomials of total degree at most \( d \). In general, we impose that \( Q \) is
exact on the space $\Pi_d$, that is, $Qp = p$ for all $p \in \Pi_d$. Some authors impose further that $Q$ is a projector on the space of splines itself. As a consequence of this property, the approximation order is $O(h^{d+1})$ on smooth functions, $h$ being the maximum step length of the partition. The coefficients $\mu_j$ is a linear combination of discrete values of $f$ at some points in the neighborhood of $\text{supp}(B_{j,d}(x))$. The associated quasi-interpolation is called a discrete quasi-interpolation.

The main advantage of quasi-interpolation is that they have a direct construction without solving any system of linear equations. Moreover, they are local, in the sense that the value of $Qf(x)$ depends only on values of $f(x)$ in a neighborhood of $x$. Finally, they have a rather small infinity norm, so they are nearly optimal approximations. In this paper, we use cubic B-spline quasi-interpolation to construct the numerical scheme of PDE.

Given some values of an unknown function $f(x_i) = f_i, i = 0, 1, \ldots, n$, consider $C^2$ cubic B-spline quasi-interpolation.\textsuperscript{26}

$$Q_3f(x) = \sum_{j=1}^{n+3} \mu_j(f) B_{j,3}(x),$$

the coefficient functionals are respectively:

$$\begin{align*}
\mu_1(f) &= f_0, \\
\mu_2(f) &= \frac{1}{18} (7f_0 + 18f_1 - 9f_2 + 2f_3), \\
\mu_j(f) &= \frac{1}{6} (-f_{j-3} + 8f_{j-2} - 7f_{j-1}), \quad j = 3, \ldots, n + 1, \\
\mu_{n+2}(f) &= \frac{1}{18} (2f_{n-2} - 9f_{n-1} + 18f_n) + 7f_n, \\
\mu_{n+3}(f) &= f_n.
\end{align*}$$

For $f \in C^4(I)$, the error estimation is

$$||f(x) - Q_3f(x)|| = O(h^4).$$

Let $\mu(f) = [\mu_0(f), \mu_1(f), \ldots, \mu_{n+3}(f)]$, and $f = [f_0, f_1, \ldots, f_n]$, in matrix form

$$\mu(f) = P \cdot f,$$

where

$$P = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 1/18 & 1 & -1/2 & 1/9 & 0 & \cdots & 0 \\
-1/6 & 0 & 4/3 & -1/6 & 0 & 0 & \cdots & 0 \\
-1/6 & 0 & 4/3 & -1/6 & 0 & 0 & \cdots & 0 \\
& \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & -1/6 & 4/3 & -1/6 & 0 \\
0 & \cdots & 0 & 1/9 & -1/2 & 1 & 7/18 & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}.$$
Numerical scheme using cubic B-spline quasi-interpolation

In this section, we construct the numerical scheme for solving Burgers-Huxley equation (1) with the cubic B-spline quasi-interpolation in space and modified Euler method for time. This scheme reduces the equation into a system of first-order ordinary differential equation (ODE) which is solved by modified Euler scheme. The efficiency of the proposed method is illustrated by two numerical experiments, which confirm that obtained results are in good agreement with earlier studies. This scheme is an easy, economical and efficient technique for finding numerical solutions for various kinds of (non)linear physical models as compared to the earlier schemes.

Discretizing Burgers-Huxley equation (1) with modified Euler scheme in time, we obtain

\[
2\frac{u_i^{n+1} - u_i^n}{\tau} = (-\alpha u_i^{n+1} (u_i^n)^2 + (u_{xx})_i^n)
+ \beta u_i^n (1 - u_i^{n+1} (u_i^n - \gamma))
\]

(8)

\[
+ (-\alpha u_i^{n+1} (u_i^n)^2 + (u_{xx})_i^n)
+ \beta u_i^n (1 - u_i^{n+1} (u_i^n - \gamma)),
\]

where \( u_i^n = u(x_i, t_n) \), \( (u_{xx})_i^n = u_{xx}(x_i, t_n) \) and \( (u_{xx})_i^n = u_{xx}(x_i, t_n) \) are approximated by the derivatives of cubic B-spline quasi-interpolant \( Q_j(u(x_i, t_n)) \), \( \tau \) is the time step. To dump the dispersion of the scheme, we define switch function \( g_i^n \) as explained in Chen and Wu, whose values are 0 or 1 at discrete points \( (x_i, t_n) \) as

\[
g_i^n = \max\{0, 1 + \min\{0, \text{sgn}(u_i^n) \cdot \text{sgn}(u_{xx}^n)\}\},
\]

where \( k = i - \text{sgn}(u_i^n) \). Thus, the resulting numerical scheme is

\[
u_i^{n+1} = u_i^n + \tau \left\{ [-\alpha u_i^{n+1} (u_i^n)^2 + (u_{xx})_i^n] + \beta u_i^n (1 - u_i^{n+1} (u_i^n - \gamma)) \right\}
\]

with \( \text{sgn}(u_i^n) \cdot \text{sgn}(u_{xx}^n) \). The comparison of the numerical solution and the analytical solution at \( x = 1, x = 6.25, x = 24.5 \) for \( t = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 \) are shown in Figures 3 and 4. From Figure 4, the errors vary from \(-1 \times 10^{-5}\) to \(4 \times 10^{-5}\) which implies the scheme is feasible and efficient.

Conclusions and further work

In this paper, we develop a kind of cubic B-spline quasi-interpolation and use it to solve Burger-Huxley equation. The accuracy and efficiency of derived solutions and errors (Analytical-numerical) of equation (9) at \( t = 0, 3, 6, 9 \) using CBSQI are shown in Figure 1. Three-dimensional graphical output of numerical solutions and errors from equation (9) are shown in Figure 2 which indicates the errors vary between \(-1 \times 10^{-5}\) and \(4 \times 10^{-5}\). Moreover, in Tables 1–4, we compare the numerical solution with the analytical solution at \( x = -12, x = -5, x = 0.6, x = 4.6 \) for \( t = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 \), respectively. These results show that the numerical solution obtained by our proposed method is in good agreement with the analytical solution. It means that this scheme is valid.

Example 2 Consider the Burgers-Fisher equation

\[
\begin{cases}
u_t + \nu^2 u_x - u_{xx} = \nu(1 - u^2), & t > 0, \quad -5 \leq x \leq 25 \\
u(x, 0) = \left(\frac{1}{2} - \frac{1}{2} \tanh \left(\frac{1}{9} (3x - 10t)\right)\right)^{\frac{1}{2}}, & -5 \leq x \leq 25
\end{cases}
\]

with the exact solution [2]

\[
u(x, t) = \left(\frac{1}{2} - \frac{1}{2} \tanh \left(\frac{1}{9} (3x - 10t)\right)\right)^{\frac{1}{2}}.
\]

The versatility and the accuracy of the proposed method are measured by the difference between numerical solutions and analytical solutions. The numerical solutions and errors (Analytical-numerical) of equation (9) at \( t = 0, 3, 6, 9 \) using CBSQI are shown in Figure 1. Three-dimensional graphical output of numerical solutions and errors from equation (9) are shown in Figure 2 which indicates the errors vary between \(-1 \times 10^{-5}\) and \(4 \times 10^{-5}\). Moreover, in Tables 1–4, we compare the numerical solution with the analytical solution at \( x = -12, x = -5, x = 0.6, x = 4.6 \) for \( t = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 \), respectively. These results show that the numerical solution obtained by our proposed method is in good agreement with the analytical solution. It means that this scheme is valid.

Example 1. Consider the following Burgers-Huxley equation

\[
\begin{cases}
u_t + \nu^2 u_x - u_{xx} = \nu(1 - u^2), & t > 0, \quad -14 \leq x \leq 6 \\
u(x, 0) = \left(\frac{1}{2} + \frac{1}{2} \tanh \left(\frac{1}{2} x\right)\right)^{\frac{1}{2}}, & -14 \leq x \leq 6
\end{cases}
\]

with the analytical solution [2]

\[
u(x, t) = \left(\frac{1}{2} + \frac{1}{2} \tanh \left(\frac{1}{9} (3x - 10t)\right)\right)^{\frac{1}{2}}.
\]
numerical scheme have been nicely validated through numerical examples which confirm that obtained results agree well with the analytic solutions. There are some valuable aspects that deserve further exploration in our future work. (i) how to analyze the stability of the numerical scheme. (ii) how to generalize the quasi-

**Figure 1.** Numerical solutions and errors to equation (9) for $t = 0, 3, 6, 9$ with CBSQI.

**Figure 2.** Numerical solutions and errors to equation (9) with CBSQI in 3D.

| Table 1. Comparison of solution of equation (9) at $x = -12$. | Table 2. Comparison of solution of Equation (9) at $x = -5$. |
|----------------------|----------------------|
| $t$ | Numerical solution | Analytical solution | Error | $t$ | Numerical solution | Analytical solution | Error |
|------|---------------------|---------------------|-------|------|---------------------|---------------------|-------|
| 0.00 | 0.18313 | 0.18313 | 0.000000 | 0.00 | 0.185594 | 0.185594 | 0.000000 |
| 1.00 | 0.20464 | 0.20464 | 0.000000 | 1.00 | 0.206525 | 0.206525 | 0.000003 |
| 2.00 | 0.22867 | 0.22867 | 0.000000 | 2.00 | 0.229582 | 0.229577 | 0.000005 |
| 3.00 | 0.25553 | 0.25553 | 0.000000 | 3.00 | 0.254898 | 0.254891 | 0.000008 |
| 4.00 | 0.28558 | 0.28558 | 0.000000 | 4.00 | 0.282582 | 0.282570 | 0.000012 |
| 5.00 | 0.31905 | 0.31906 | 0.000001 | 5.00 | 0.312702 | 0.312686 | 0.000016 |
| 6.00 | 0.35650 | 0.35651 | 0.000001 | 6.00 | 0.345278 | 0.345258 | 0.000021 |
| 7.00 | 0.39834 | 0.39835 | 0.000001 | 7.00 | 0.380259 | 0.380234 | 0.000025 |
| 8.00 | 0.44506 | 0.44507 | 0.000001 | 8.00 | 0.417505 | 0.417475 | 0.000030 |
| 9.00 | 0.49725 | 0.49725 | 0.000001 | 9.00 | 0.456771 | 0.456737 | 0.000034 |
| 10.00 | 0.55551 | 0.55552 | 0.000001 | 10.00 | 0.497696 | 0.497658 | 0.000038 |
Table 3. Comparison of solution of equation (9) at $x = 0.6$.

| $t$  | Numerical solution | Analytical solution | Error  |
|-----|--------------------|---------------------|--------|
| 0.00| 0.773749           | 0.773749            | 0.000000 |
| 1.00| 0.806666           | 0.806675            | -0.000008 |
| 2.00| 0.836293           | 0.836302            | -0.000009 |
| 3.00| 0.862525           | 0.862533            | -0.000008 |
| 4.00| 0.885408           | 0.885415            | -0.000006 |
| 5.00| 0.905107           | 0.905111            | -0.000004 |
| 6.00| 0.921864           | 0.921867            | -0.000003 |
| 7.00| 0.935975           | 0.935976            | -0.000001 |
| 8.00| 0.947754           | 0.947753            | 0.000000 |
| 9.00| 0.957512           | 0.957511            | 0.000001 |
| 10.00| 0.965547          | 0.965546             | 0.000002 |

Figure 3. Numerical solutions and errors to equation (10) for $t = 0, 1.5, 3.5$ with CBSQI.

Table 4. Comparison of solution of equation (9) at $x = 4.6$.

| $t$  | Numerical solution | Analytical solution | Error  |
|-----|--------------------|---------------------|--------|
| 0.00| 0.977495           | 0.977495            | 0.000000 |
| 1.00| 0.981859           | 0.981858            | 0.000001 |
| 2.00| 0.985395           | 0.985394            | 0.000000 |
| 3.00| 0.988254           | 0.988254            | 0.000000 |
| 4.00| 0.990562           | 0.990561            | 0.000000 |
| 5.00| 0.992421           | 0.992421            | 0.000000 |
| 6.00| 0.993918           | 0.993917            | 0.000000 |
| 7.00| 0.995121           | 0.995120            | 0.000000 |
| 8.00| 0.996088           | 0.996087            | 0.000000 |
| 9.00| 0.996864           | 0.996863            | 0.000000 |
| 10.00| 0.997486          | 0.997486             | 0.000001 |

Figure 4. Numerical solutions and errors to equation (10) with CBSQI in 3D.
interpolation scheme to solve high dimension PDE. These research topics are more challenging.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work is partly supported by the National Natural Science Foundation of China (Nos. 11801490, 11671068).

ORCID iD
Lan-Yin Sun https://orcid.org/0000-0003-2297-5012

References
1. Wang X, Zhu Z and Lu Y. Solitary wave solutions of the generalised Burgers-Huxley equation. *J Phys A Math Gen* 1990; 23: 271–274.
2. Molabahrami A and Khani F. The homotopy analysis method to solve the Burgers–Huxley equation. *Nonlinear Anal Real World Appl* 2009; 10: 589–600.
3. Wazwaz AM. Travelling wave solutions of generalized forms of Burgers, Burgers–KdV and Burgers–Huxley equations. *Appl Math Comput* 2005; 169: 639–656.
4. Batista B, Noorani MSM and Hashim I. Application of variational iteration method to the generalized Burgers–Huxley equation. *Chaos Solitons Fractals* 2008; 36: 660–663.
5. Yefimova OY and Kudryashov N. Exact solutions of the Burgers-Huxley equation. *J Appl Math Mech* 2004; 3: 413–420.
6. Ablowitz MJ, Fuchssteiner B and Kruskal MD. *Topics in soliton theory and exactly solvable nonlinear equations: proceedings of the conference on nonlinear evolution equations, solitons and the inverse scattering transform*, Oberwolfach, Germany, July 27–August 2, 1986. Singapore: World Scientific, 1987.
7. Bateman H. Some recent researches on the motion of fluids. *Mon Weather Rev* 1915; 43: 163–170.
8. Wang M. Exact solutions for a compound KdV-Burgers equation. *Phys Lett A* 1996; 213: 279–287.
9. Su CH and Gardner CS. Korteweg-de Vries equation and generalizations. III. Derivation of the Korteweg-de Vries equation and Burgers equation. *J Math Phys* 1969; 10: 536–539.
10. Benton ER and Platzman GW. A table of solutions of the one-dimensional Burgers equation. *Q Appl Math* 1972; 30: 195–212.
11. Bertini L, Cancrini N and Jona-Lasinio G. The stochastic Burgers equation. *Commun Math Phys* 1994; 165: 211–232.
12. Hodgkin AL and Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol* 1952; 117: 500–544.
13. Kevkorkian J and Cole JD. *Perturbation methods in applied mathematics*. Vol. 34. Berlin: Springer Science & Business Media, 2013.
14. Malfliet W and Hereman W. The tanh method: I. Exact solutions of nonlinear evolution and wave equations. *Phys Scr* 1996; 54: 563.
15. Malfliet W and Hereman W. The tanh method: II. Perturbation technique for conservative systems. *Phys Scr* 1996; 54: 569–575.
16. Miller EL. *Predictor-corrector studies of Burgers’ model of turbulent flow*. PhD Thesis, University of Delaware, 1966.
17. Christie I and Mitchell A. Upwinding of high order Galerkin methods in conduction-convection problems. *Int J Numer Methods Eng* 1978; 12: 1764–1771.
18. Zhu C-G and Wang R-H. Numerical solution of Burgers’ equation by cubic B-spline quasi-interpolation. Appl Math Comput 2009; 208: 260–272.
19. Griffiths G and Schiesser WE. Traveling wave analysis of partial differential equations: numerical and analytical methods with MATLAB and Maple. Burlington, USA: Academic Press, 2010.
20. Li C-Y and Zhu C-G. A multilevel univariate cubic spline quasi-interpolation and application to numerical integration. Math Methods Appl Sci 2010; 33: 1578–1586.
21. Bhatti M, Ellahi R, Zeeshan A, et al. Numerical study of heat transfer and Hall current impact on peristaltic propulsion of particle-fluid suspension with compliant wall properties. Mod Phys Lett B 2019; 33: 1950439.
22. Bhatti MM, Shahid A, Abbas T, et al. Study of activation energy on the movement of gyrotactic microorganism in a magnetized nanofluids past a porous plate. Processes 2020; 8: 328.
23. Zhang L, Arain M, Bhatti M, et al. Effects of magnetic Reynolds number on swimming of gyrotactic microorganisms between rotating circular plates filled with nanofluids. Appl Math Mech 2020; 41: 637–654.
24. Goldman R. Pyramid algorithms: a dynamic programming approach to curves and surfaces for geometric modeling. San Francisco: Elsevier, 2002.
25. Ali A, Gardner G and Gardner L. A collocation solution for Burgers’ equation using cubic B-spline finite elements. Comput Methods Appl Mech Eng 1992; 100: 325–337.
26. Sablonniere P. Univariate spline quasi-interpolants and applications to numerical analysis. arXiv preprint math/0504022 2005.
27. De Boor C, Höllig K and Riemenschneider S. Box splines. Vol. 98. New York: Springer Science & Business Media, 2013.
28. De Boor C, De Boor C, Mathématicien E-U, et al. A practical guide to splines. Vol. 27. New York: Springer-Verlag, 1978.
29. Lyche T and Schumaker LL. Local spline approximation methods. J Approx Theory 1975; 15: 294–325.
30. Chen R and Wu Z. Applying multiquadric quasi-interpolation to solve Burgers’ equation. Appl Math Comput 2006; 172: 472–484.