The exponential cubic B-spline algorithm for Fisher equation

Idiris Dag, Ozlem Ersoy*

Eskişehir Osmangazi University, Department of Mathematics-Computer, Faculty of Science and Art, Eskişehir, Turkey

A R T I C L E   I N F O

Article history:
Received 18 February 2015
Revised 3 January 2016
Accepted 21 February 2016

PACS:
02.70.Jn
47.35.Fg

Keywords:
Collocation methods
Exponential cubic B-spline
Fisher’s equation

A B S T R A C T

In this study, exponential B-spline collocation method is set up for solving Fisher’s equation. Integration of Fisher’s equation is managed by use of the exponential cubic B-spline in space and the Crank–Nicolson method in time. The effect of reaction and diffusion is observed by studying three test problems. A comparison is performed between the obtained numerical results and some earlier results using $L_{\infty}$ and relative error norms.

The exponential B-splines are piecewise polynomial functions including a free parameter. It and its properties are introduced by [5]. Although the use of it in the numerical methods is not as wide as the well-known B-splines, there are few papers dealing with the use of the exponential B-splines for finding solutions of the differential equations: Some variants of the exponential B-spline collocation methods have been set up to numerical solutions of the Singularly perturbed problems in the studies [4,7,8]. Time dependent partial differential equations, Convection–Diffusion, Korteweg–de Vries (KdV), generalized Burgers, equal width and Generalized Long Wave equations, have been solved numerically by way of the exponential B-spline collocation method recently [14,16–19].

The numerical solutions of the differential equations have investigated using spline functions. So spline functions are used to build up numerical methods for finding solutions of differential equations. Publications also exist dealing with the numerical solutions of Fisher’s equation by way of spline-related techniques. We want to mention some of them: the Galerkin scheme whose trial function is consist of combination of the cubic B-splines is set up to find numerical solutions of Fisher’s equation over the finite elements [9], the Galerkin method is used with quadratic B-spline base functions to obtain the numerical solutions of Fisher’s equation [10], a cubic B-spline collocation method is given to solve Fisher’s equation [11], the numerical solution of Fisher’s equation is given by using collocation method based on the modified cubic B-spline method [13], cubic B-spline quasi-interpolation is established to obtain solutions of Fisher’s equation numerically [12], the quintic B-spline collocation method is proposed to get solution of Fisher’s equation in the study [15].

1. Introduction

The reaction–diffusion equations have an important role in modeling some physical phenomena. They are used in many fields such as biology, chemistry and engineering. Because of the complexities in finding their solutions, approximate solutions of reaction–diffusion equations have become a central tool in their study. In one space dimension, the nonlinear reaction–diffusion equations may be written in the following form:

$$U_t = \lambda U_{xx} + \phi(U)$$

where $U = U(x,t)$ is a time dependent real-valued function. $\lambda U_{xx}$ is called diffusivity term where the coefficient $\lambda$ is a non-negative constant and the function $\phi(U)$ describes the reaction term. One of the most popular special case of the Eq. (1) is given by

$$U_t = \lambda U_{xx} + \beta U(1 - U), \quad -\infty < x < \infty, \ t > 0$$

where $\beta$ is a real parameter. This equation is known as a Fisher’s equation, which was introduced by Fisher [1] who describe it to model the kinetic advancing rate of an advantageous gene. Since then works have been done to extend the model to take into account the other biological, chemical and physical events. Fisher’s equation is also defined to represents the evolution of the population due to the two competing physical processes, flame propagation, nuclear reactor, auto-catalytic chemical reactions, logistic growth models and neurophysiology.

* Corresponding author. Tel.: +902222393750.
E-mail address: ozersoy@ogu.edu.tr (O. Ersoy).

http://dx.doi.org/10.1016/j.chaos.2016.02.031
0960-0779/© 2016 Elsevier Ltd. All rights reserved.
In this study, we investigate the solvability of Fisher’s equations by the exponential cubic B-spline finite element methods. We have mentioned that solutions of Fisher’s equation have been obtained by the polynomial B-spline based methods. As known, generalization of the polynomial B-splines is the exponential B-splines including a free parameter which gives different bell-shaped functions. Thus, an approximation function consisting of combination of the exponential B-splines are used in the collocation method over the interval. Our aim is to see the effect of the approximation method accompanied the exponential B-spline functions and compare results of both present methods and B-spline-related method. The free parameter of the exponential B-spline functions is determined experimentally by scanning values with small increment in some predetermined intervals to get the least error when the collocation method is run for getting solutions of Fisher’s equation. Fully-integration of Fisher’s equation is achieved by using Crank–Nicolson method for the time discretization and exponential cubic B-spline collocation method for the space discretization. The efficiency of the collocation method together with the exponential B-splines is searched on solutions of Fisher’s equation.

The initial and two boundary conditions for Fisher’s equation (2) are given as

\[ U(x, 0) = f(x), x \in [-\infty, \infty] \]  

\[
\lim_{x \to -\infty} U(x, t) = 1, \lim_{x \to +\infty} U(x, t) = 0 \quad \text{or} \\
\lim_{x \to -\infty} U(x, t) = 0.
\]

In the literature, conditions (3) and together (4) are commonly referred as nonlocal conditions, while conditions (3) and (5) are usually known as local conditions.

2. Exponential cubic B-spline collocation method

Knots are equally distributed over the problem domain \([a, b]\) as

\[
\pi : a = x_0 < x_1 < \cdots < x_N = b
\]

with mesh spacing \(h_1 = (b - a)/N\) and \(h = (b - a)/N\). The exponential cubic B-splines, \(B_i(x)\), at knots including knots \(x_i, i = -3, -2, -1, N + 1, N + 2, N + 3\) outside problem domain can be defined as

\[
B_i(x) = \begin{cases} 
  b_0 \left( x_{i-2} - x_i - \frac{1}{p} (\sinh(p(x_{i-2} - x_i))) \right) \\
  a_1 + b_1 \left( x_{i-2} - x_i + c_1 \exp(p(x_{i-2} - x_i)) + d_1 \exp(-p(x_{i-2} - x_i)) \right) \\
  b_2 \left( x_{i-2} - x_{i+2} - \frac{1}{p} (\sinh(p(x_{i+2} - x_{i+2}))) \right) \\
  0 
\end{cases}
\]

where

\[
a_1 = \frac{\beta c + \delta}{\beta c - s}, \quad b_1 = \frac{p}{2} \left[ \frac{c(c - 1) + s^2}{(\beta c - s)(1 - c)} \right], \quad b_2 = \frac{p}{2(\beta c - s)},
\]

\[
c_1 = 1 \left[ \frac{\exp(-p)(1 - c) + s(\exp(-p) - 1)}{(\beta c - s)(1 - c)} \right],
\]

\[
d_1 = 1 \left[ \frac{\exp(p)(c - 1) + s(\exp(p) - 1)}{(\beta c - s)(1 - c)} \right]
\]

and

\[
c = \cosh(\phi h), \quad s = \sinh(\phi h), \quad p \text{ is a free parameter.}
\]

\([B_{-1}(x), B_0(x), \ldots, B_{N+4}(x)]\) form a basis for the functions defined over the interval \([a, b]\). Each basis function \(B_i(x)\) is twice continuously differentiable. The values of \(B_i(x), B_i'(x)\) and \(B_i''(x)\) at the knots \(x_i\), computed from Eq. (6), are documented in Table 1.

![Table 1](image-url)

Table 1

<table>
<thead>
<tr>
<th>(x)</th>
<th>(x_{i-2})</th>
<th>(x_{i-1})</th>
<th>(x_i)</th>
<th>(x_{i+1})</th>
<th>(x_{i+2})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(B_i)</td>
<td>0</td>
<td>(e^{-\phi x_i})</td>
<td>1</td>
<td>(e^{-\phi x_i})</td>
<td>0</td>
</tr>
<tr>
<td>(B_i')</td>
<td>0</td>
<td>(2(x_i - c))</td>
<td>0</td>
<td>(2(x_i - c))</td>
<td>0</td>
</tr>
<tr>
<td>(B_i'')</td>
<td>0</td>
<td>(p^2)</td>
<td>(\phi^2)</td>
<td>(p^2)</td>
<td>(\phi^2)</td>
</tr>
</tbody>
</table>

Now suppose that an approximate solution \(U_n\) is written in terms of the exponential B-splines as

\[ U_n(x, t) = \sum_{i=1}^{N+1} \delta_i B_i(x) \]

where \(\delta_i\) are time dependent parameters to be determined from the collocation procedure, the boundary and initial conditions. Evaluation of Eq. (7), its first and second derivatives at knots \(x_i\) using the Table 1 yields the nodal values \(U_i\) in terms of parameters

\[
U_i = U(x_i, t) = \frac{s - ph}{2(\beta c - s)} \delta_{i-1} + \delta_i + \frac{s - ph}{2(\beta c - s)} \delta_{i+1},
\]

\[
U_i' = U'(x_i, t) = \frac{p(1 - c)}{2(\beta c - s)} \delta_{i-1} + \frac{p(1 - c)}{2(\beta c - s)} \delta_{i+1},
\]

\[
U_i'' = U''(x_i, t) = \frac{p^2 s}{2(\beta c - s)} \delta_{i-1} - \frac{p^2 s}{2(\beta c - s)} \delta_{i+1} + \frac{p^2 s}{2(\beta c - s)} \delta_{i+1}.
\]

The Crank–Nicolson scheme is used to discretize time variables of the unknown \(U\) in Fisher’s equation so that one obtains the time discretized form of the equation as

\[
\frac{U_n^{n+1} - U_n^n}{\Delta t} = \lambda \left( \frac{(U_{n+2} - U_{n+1})}{2\Delta t} \right) + \beta \frac{U_{n+1} + U_n^n}{2} - \beta \left( \frac{(U_{n+2} - U_n^n)^2}{2} \right)
\]

where \(U_{n+1} = U(x, t_{n+1})\) is the solution of the equation at the \((n+1)\)th time level. Here, \(t_{n+1} = t_n + \Delta t\) and \(\Delta t\) is the time step, superscript denote nth time level, \(t = n\Delta t\). The nonlinear term \((U^2)_{n+1}\) in Eq. (9) may be linearized by using the following term [2]:

\[
(U^2)_{n+1} = 2U^nU_{n+1} - (U^n)^2
\]

Thus we get

\[
2U^n\Delta t U_{n+1} - \beta \Delta t U_{n+1} + 2\beta \Delta t U_{n+1} = 2U^n + \lambda \Delta t U_{n+1} + \beta \Delta t U^n
\]

Substitution of (7) into (11) leads to the fully-discretized equation:

\[
\chi_1 \delta_{m-1} + \chi_2 \delta_{m} + \chi_3 \delta_{m+1} = \chi_4 \delta_{m-1} + \chi_5 \delta_{m} + \chi_6 \delta_{m+1}
\]

where

\[
\chi_1 = (2 - \beta \Delta t + 2\beta \Delta t) \alpha - \lambda \Delta t \gamma
\]

\[
\chi_2 = (2 - \beta \Delta t + 2\beta \Delta t) \alpha + 2\lambda \Delta t \gamma
\]

\[
\chi_3 = (2 + \beta \Delta t) \alpha + \lambda \Delta t \gamma
\]
\[ \chi_t = (2 + \beta \Delta t) - 2\lambda \Delta t \gamma \]
\[ L = \alpha \delta_{i-1} + \delta_i + \alpha \delta_{i+1} \]
\[ \alpha = \frac{s - ph}{2(phc - s)}, \quad \gamma = \frac{p^2 s}{2(phc - s)}. \]

The system consists of \( N + 1 \) linear equation in \( N + 3 \) unknown parameters \( \delta^{n+1}_0, \delta^{n+1}_1, \ldots, \delta^{n+1}_{N+1} \). To make solvable the system, boundary conditions \( U_0 = \sigma_1 \), \( U_\pi = \sigma_2 \) are used to find two additional linear equations:
\[ \delta_{-1} = \frac{1}{\alpha} (\sigma_1 - \delta_0 - \alpha \delta_1), \quad (13) \]
\[ \delta_{N+1} = \frac{1}{\alpha} (\sigma_2 - \alpha \delta_{N-1} - \delta_N). \]

(13) can be used to eliminate \( \delta_{-1}, \delta_{N+1} \) from the system (12) which then becomes a solvable matrix equation for the unknown \( \delta^{n+1}_0, \ldots, \delta^{n+1}_{N} \). A tridiagonal system of equations can be solved with Thomas algorithm.

Initial parameters \( \delta^0_0, \delta^0_1, \ldots, \delta^0_{N+1} \) can be determined from the initial condition and first space derivative of the initial conditions at the boundaries as the following:
1. \( U_N(x_N, 0) = U(x, 0), \quad i = 0, \ldots, N \)
2. \( U_N(x_N, 0) = U'(x_N) \)
3. \( U_N(x_N, 0) = U'(x_N) \).

3. Numerical tests

Numerical method described in the previous section is tested on three problems to demonstrate the robustness and numerical accuracy.

The discrete \( L_\infty \) error norm
\[ L_\infty = \| U - U_N \|_\infty = \max_j | U_j - (U_N)_j | \]
and relative error norm
\[ \text{Rel} = \left( \frac{\sum_{j=1}^{N} |U_j^{n+1} - U_j^n|^2}{\sum_{j=1}^{N} |U_j^n|^2} \right)^{1/2} \]
are used to measure error between the analytical and numerical solutions.

(a) A particular solution of Eq. (2) was found by Ablowitz and Zeptetla [3], which is given by
\[ U(x, t) = \left( 1 + \exp \left( \frac{\beta}{6} x - \frac{5\beta}{6} t \right) \right)^{-2} \]

The numerical solution of Eq. (2) with initial conditions (14) with \( t = 0 \) and boundary conditions replaced by the artificial boundary conditions \( U(a, t) = 1, U(b, t) = 0 \) over the problem domain \( [a, b] = [-0.2, 0.8] \) has been found for \( \lambda = 0.1 \) and \( \beta = 1 \). \( N = 40, 120 \) with \( \Delta t = 0.0001 \) at times \( t = 0.002, 0.003, 0.004, 0.005, 0.006, 0.007 \) in Fig. 1, at times \( t = 0.001, 0.002, 0.003, 0.004, 0.005 \) in Fig. 2 and times \( t = 0.001, 0.0015, 0.002, 0.0025, 0.003, 0.0035 \) in Fig. 3, agreeing with same parameters in the study [11].

For \( N = 64 \) results are presented in Table 2 to compare with Refs. [6,10,15] in terms of \( L_\infty \) error norm, at different time steps.

In Figs. 4 and 5, \( L_\infty \) error norm is depicted for \( p = 1 \) and \( \beta = 10.000 \) at \( t = 5.0 \times 10^{-4}, 1.0 \times 10^{-3}, 1.5 \times 10^{-3}, 2.0 \times 10^{-3}, 2.5 \times 10^{-3}, 3.0 \times 10^{-3}, 3.5 \times 10^{-3} \) for \( N = 64 \) and \( N = 120 \), respectively.

(b) Secondly, the initial pulse profile
\[ U(x, 0) = \sec h^2 (10x) \]
is chosen as the initial condition for our first numerical experiment together with boundary condition (5), \( U(-50, t) = U(50, t) = 0 \). In numerical calculations, the constants in Eq. (2) are selected as \( \lambda = 0.1 \) and \( \beta = 1 \). For the discretization of space and time, the parameters \( h = 0.025 \) and \( \Delta t = 0.05 \) are used. Then, the algorithm is run up to time \( t = 40 \) over the domain \([-50, 50] \). In this test problem, Fig. 6 is drawn for \([a, b] = [-2, 2] \) at \( t = 0, 0.1, 0.2, 0.3, 0.4 \) and 0.5, while Fig. 7 is drawn for \([a, b] = [-6, 6] \) at \( t = 0, 1, 2, 3, 4 \) and 5. In Fig. 8 \([a, b] = [-50, 50] \) at \( t = 0, 5, 10, 15, 20, 25, 30, 35 \) and 40. In early times of the run, diffusion is more effective than reaction as seen in Fig. 6 in which the crest of the pulse declines until the pulse reaches the minimum value approximately...
equal to 0.33. As time passes, reaction terms start to influence over the diffusion illustrated in Fig. 7 and the domination of the diffusion is seen in Fig. 8 for the solution behavior in which the pulse has reached the maximum value $U = 1$ and becomes flatter. In the

Table 2
Maximum error norms at some different times for the first test problem.

| Parameters: $\lambda = 1$, $\beta = 10000$, $N = 64$, $\Delta t = 5 \times 10^{-6}$ and $x \in [-0.2, 0.8]$ |
|---|---|---|---|---|
| Time | $t = 0.0005$ | $t = 0.0015$ | $t = 0.0025$ | $t = 0.0035$ |
| Present $p = 1$ | $1.10 \times 10^{-2}$ | $1.49 \times 10^{-1}$ | $3.44 \times 10^{-1}$ | $5.08 \times 10^{-1}$ |
| Present | $3.54 \times 10^{-3}$ | $7.63 \times 10^{-2}$ | $2.04 \times 10^{-2}$ | $1.52 \times 10^{-2}$ |
| Various $p$ | $(p = 0.000000300)$ | $(p = 0.000000192)$ | $(p = 0.000000089)$ | $(p = 0.000000089)$ |
| Ref [15] | $2.05 \times 10^{-4}$ | $2.30 \times 10^{-3}$ | $5.49 \times 10^{-3}$ | $9.04 \times 10^{-3}$ |
| CN, Ref [6] | $2.55 \times 10^{-3}$ | $1.62 \times 10^{-2}$ | $8.65 \times 10^{-2}$ | $6.98 \times 10^{-2}$ |
| ASD, Ref [6] | $1.03 \times 10^{-2}$ | $1.25 \times 10^{-1}$ | $2.80 \times 10^{-1}$ | $4.48 \times 10^{-1}$ |
| PPS, Ref [6] | $1.07 \times 10^{-2}$ | $4.93 \times 10^{-2}$ | $9.37 \times 10^{-2}$ | $9.44 \times 10^{-2}$ |
| DSC, Ref [6] | $3.13 \times 10^{-6}$ | $3.90 \times 10^{-6}$ | $7.82 \times 10^{-6}$ | $3.42 \times 10^{-6}$ |

Table 3
Relative errors at some different times for the second test problem.

| Parameters: $\lambda = 0.1$, $\beta = 1$, $N = 64$, $\Delta t = 0.05$ and $x \in [-50, 50]$ |
|---|---|---|---|---|
| Method | $t = 5$ | $t = 10$ | $t = 15$ | $t = 20$ |
| Present $p = 1$ | $0.113 \times 10^{-2}$ | $1.268 \times 10^{-1}$ | $9.152 \times 10^{-3}$ | $3.228 \times 10^{-3}$ |
| Present | $7.730 \times 10^{-3}$ | $1.364 \times 10^{-3}$ | $9.087 \times 10^{-4}$ | $6.412 \times 10^{-4}$ |
| (Various $p$) | $(p = 1.4235)$ | $(p = 0.1773)$ | $(p = 1.92731)$ | $(p = 1.35700)$ |
| Ref [10] | $1.386 \times 10^{-2}$ | $7.860 \times 10^{-3}$ | $6.054 \times 10^{-3}$ | $5.090 \times 10^{-3}$ |

![Figure 4](image1.png) **Fig. 4.** $L_{\infty}$ error norm for $N = 64$. 

![Figure 5](image2.png) **Fig. 5.** $L_{\infty}$ error norm for $N = 120$. 

![Figure 6](image3.png) **Fig. 6.** Solutions at early times.

Table 3 errors at some different times for the second test problem is documented together with error of the quadratic Galerkin method.

(c)Lastly, Fisher’s equation is studied with local boundary condition (5) and initial condition given by

$$U(x, 0) = \begin{cases} e^{10(x+1)}, & x < -1 \\ e^{10(x-1)}, & x > 1 \\ 1, & -1 \leq x \leq 1 \end{cases}$$

This example is considered their studies [11,15]. In this test problem, the parameters are chosen as $\Delta t = 0.05$ and $N = 40$. In Fig. 9, numerical solutions are drawn for $[a, b] = [-4, 4]$ at $t = 0.1$, $0.2$, $0.3$, $0.4$ and $0.5$, while, in Fig. 10, is drawn for $[a, b] = [-6, 6]$ at $t = 1$, $2$, $3$, $4$, and $5$. In early times of the run, although, generally, effect of the reaction–diffusion is small, reaction is more effective than diffusion because there is change from the sharpness to smoothness near $x = \pm 1$ in the solutions. In Fig. 11, over the interval $[a, b] = [-30, 30]$ at $t = 5$, $10$, $15$, $20$, $25$ and $30$ solution profiles are depicted from which solution becomes flatter and flatter due to diffusion effect.

In the Table 4 errors at some different times for the last test problem is documented.
Table 4
Relative errors at some different times for the last test problem.

<table>
<thead>
<tr>
<th>Method</th>
<th>$t = 5$</th>
<th>$t = 10$</th>
<th>$t = 15$</th>
<th>$t = 20$</th>
<th>$t = 40$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Present</td>
<td>$1.033 \times 10^{-2}$</td>
<td>$7.458 \times 10^{-3}$</td>
<td>$5.967 \times 10^{-3}$</td>
<td>$5.088 \times 10^{-3}$</td>
<td>$3.490 \times 10^{-3}$</td>
</tr>
<tr>
<td>Ref. [10]</td>
<td>$9.435 \times 10^{-3}$</td>
<td>$6.917 \times 10^{-3}$</td>
<td>$5.614 \times 10^{-3}$</td>
<td>$4.825 \times 10^{-3}$</td>
<td>$3.352 \times 10^{-3}$</td>
</tr>
</tbody>
</table>

4. Conclusion

A collocation algorithm is suggested to solve Fisher's equation. Combination of the exponential B-spline functions is used to construct an approximation function for the collocation method. The fully-integration of Fisher's equation leads to an algebraic iterative procedure in the form of three band matrix equation. Advantage of the present algorithm is both to solve the matrix equation of having small dimension than methods mentioned in the paper easily. Generally, there is a slight decrease for the error with free parameters used just around zero value seen in Tables 2 and 3. The discrete singular decomposition method provides the least error than the method listed in the Table 2. The proposed method provides less error than the finite difference method and quadratic B-spline Galerkin method and higher than the quintic B-spline collocation method. As a results, this alternative method gives fairly good solutions when compared results with existing results carried out with some other numerical methods.
Acknowledgment

This paper has been presented at the 3rd international Eurasian conference on mathematical sciences and applications IECMSA_2014 Vienna-Austria.

References
