An element-free Galerkin (EFG) method for generalized Fisher equations (GFE)*

Shi Ting-Yu(时婷玉)[a], Cheng Rong-Jun(程荣军)[b], and Ge Hong-Xia(葛红霞)[a]†

[a] Faculty of Science, Ningbo University, Ningbo 315211, China
[b] Ningbo Institute of Technology, Zhejiang University, Ningbo 315100, China

(Received 7 November 2012; revised manuscript received 15 December 2012)

A generalized Fisher equation (GFE) relates the time derivative of the average of the intrinsic rate of growth to its variance. The exact mathematical result of the GFE has been widely used in population dynamics and genetics, where it originated. Many researchers have studied the numerical solutions of the GFE, up to now. In this paper, we introduce an element-free Galerkin (EFG) method based on the moving least-square approximation to approximate positive solutions of the GFE from population dynamics. Compared with other numerical methods, the EFG method for the GFE needs only scattered nodes instead of meshing the domain of the problem. The Galerkin weak form is used to obtain the discrete equations, and the essential boundary conditions are enforced by the penalty method. In comparison with the traditional method, numerical solutions show that the new method has higher accuracy and better convergence. Several numerical examples are presented to demonstrate the effectiveness of the method.

Keywords: element-free Galerkin (EFG) method, meshless method, generalized Fisher equation (GFE)

PACS: 02.60.Lj, 03.65.Ge **DOI:** 10.1088/1674-1056/22/6/060210

1. Introduction

In 1930, Fisher[1] proposed an equation for population dynamics that relates the time derivative of the average of the intrinsic rate of growth of the different alleles to its variance. It has become known as the fundamental law of population genetics. The generalized form of the Fisher equation (GFE) holds for temporal functions, which are different from zero with continuous second-order derivatives. The exact results of the Fisher equation are independent of the detailed kinetics of the process. They are valid whether the evolution equations are linear or nonlinear, or local or nonlocal in space and time.[2] The use of the GFE is new in many fields, such as chemistry, biology, biotechnology, chemical engineering, materials science, etc.[3]

The properties of the Fisher equation have been studied theoretically by many authors and the travelling wave solutions of the Fisher equation have been studied by many computational approaches. One of the first numerical solutions was presented in Ref. [4] with a pseudospectral approach. The moving mesh method for numerical solutions of the Fisher equation was used by Qiu and Sloan.[5] Wazwaz and Gorguis[6] found the exact solutions to the Fisher equation and to a nonlinear diffusion equation of the Fisher type by employing the Adomian decomposition method. The Galerkin finite element method,[7] and sinc collocation method[8] have a good use in the Fisher equation. Mickens[9] proposed a best finite-difference scheme for the Fisher equation, and it was used by many others.[10–15]

We consider the following Fisher equation,

$$u_t = [(a_0u + a_1)u_x]_x + (a_2 - a_3u)u, \tag{1}$$

where u is the biological population density. The first term on the right-hand side of Eq. (1) is called the diffusion term, which denotes the migration of the population. If the diffusion term is 0, the distribution of the population in space is uniform, u is just involved with time. Otherwise the population is necessary from high density to low density diffusion. The second term on the right-hand side of Eq. (1) is called the reaction term, which represents the propagation of the population, in which a_2 is the intrinsic natural growth rate, it is equal to the birth rate minus mortality, $a_2 - a_3u$ is actual increase. General requirement is $a_2a_3 > 0$. When $a_0 = 0, a_2 = a_3$, the equation is called the classic Fisher equation. The Neumann boundary condition and initial condition are respectively given as follows:

$$u_x(a,t) = g_1(t),\ u_x(b,t) = g_2(t), \tag{2}$$

$$u(x,0) = u_0(x),\ x \in (a,b). \tag{3}$$

The original EFG method was developed by Belytschko *et al.*,[16] which is only based on nodes, thus no mesh generation or remeshing is required. It has been shown that, compared with the finite element method, the EFG method has the advantages of high accuracy, rapid convergence, and a smooth stress solution can be obtained without post-processing.[16]

*Project supported by the National Natural Science Foundation of China (Grant No. 11072117), the Natural Science Foundation of Ningbo City (Grant Nos. 2012A610038 and 2012A610152), the Scientific Research Fund of Education Department of Zhejiang Province, China (Grant No. Z201119278), and K.C. Wong Magna Fund in Ningbo University.

†Corresponding author. E-mail: gehongxia@nbu.edu.cn

© 2013 Chinese Physical Society and IOP Publishing Ltd

The variational method is used to obtain the corresponding discrete equations, the essential boundary conditions are enforced by the penalty method. The EFG method is greatly developed and widely used in various fields, for example, for solving linear hyperbolic equation and KdV equations,[17,18] for simulation of stationary shallow water flowing in rivers,[19] analysis of thin plates,[20] and so on. Recently an improved interpolating EFG method has been studied and used to two-dimensional potential problems[21–24] and others.[25–30] In the present paper, the EFG method is used to obtain the numerical solution of the GFE.[31]

The paper is organized in the following way: in Section 2, we describe the EFG method, including the approximation method and the weight function. In Section 3, we use the EFG method to discretize the equation of the problem. In order to present a clear overview of the method, in Section 4, some examples are solved. Section 5 gives the conclusion of this paper.

2. Moving least-square (MLS) approximation

Moving least square (MLS) approximation was proposed originally by mathematicians for data fitting and surface structures. Lancaster and Salkauskas pioneered the use of surface generation using the MLS, and then Belytschko applied it to the mesh-free method (FEM).

The requirements of MLS approximation usually contain three parts: a monomial basis, a weight function of a compact support associated with each node, and a set of coefficients depending on the node position.

In order to obtain $u^h(x)$, which is the approximation function of $u(x)$, Lancaster and Salkauskas defined a local approximation

$$u^h(x,\bar{x}) = \sum_{k=1}^{m} p_k(x)a_k(\bar{x}) \equiv \boldsymbol{p}(x)\boldsymbol{a}(\bar{x}), \tag{4}$$

where $\bar{x}$ is a calculated point and x is the node in the compacted supported domain of $\bar{x}$, $\boldsymbol{p}^{\mathrm{T}}(x)$ is a complete monomial basis of order, as a linear basis is used $\boldsymbol{p}^{\mathrm{T}}(x) = (1,x)$, or a quadratic basis $\boldsymbol{p}^{\mathrm{T}}(x) = (1,x,x^2)$, $\boldsymbol{a}(\bar{x})$ is an unknown coefficient vector, and it is the vector forms of $a_k(\bar{x})$.

To get the unknown coefficients $\boldsymbol{a}(\bar{x})$, we define the following weighted discrete L_2 norm

$$\begin{aligned}\boldsymbol{J} &= \sum_{i=1}^{N} w(x-x_i)[u^h(x_i,\bar{x}) - u_i]^2 \\ &= \sum_{i=1}^{N} w(x-x_i)[\boldsymbol{p}^{\mathrm{T}}(x_i)\boldsymbol{a}(\bar{x}) - u_i]^2,\end{aligned} \tag{5}$$

where $w(x-x_i)$ is a weight function, x_i is the node, N is the number of nodes in the compacted supported domain of x, u_i is the nodal parameter of u at $x = x_i$.

By minimizing $\boldsymbol{J}$ in Eq. (5), we have

$$\frac{\partial \boldsymbol{J}}{\partial \boldsymbol{a}} = \boldsymbol{A}(x)\boldsymbol{a}(x) - \boldsymbol{B}(x)\boldsymbol{U} = 0, \tag{6}$$

where

$$\boldsymbol{A}(x) = \sum_{i=1}^{N} w(x-x_i)\boldsymbol{p}(x_i)\boldsymbol{p}^{\mathrm{T}}(x_i), \tag{7}$$

$$\begin{aligned}\boldsymbol{B}(x) = {} & (w(x-x_1)\boldsymbol{p}(x_1),\ w(x-x_2)\boldsymbol{p}(x_2),\ \ldots, \\ & w(x-x_N)\boldsymbol{p}(x_N)),\end{aligned} \tag{8}$$

$$\boldsymbol{U} = (u_1,\ u_2,\ \ldots,\ u_N)^{\mathrm{T}}. \tag{9}$$

Then Eq. (6) is equal to

$$\boldsymbol{a}(x) = \boldsymbol{A}^{-1}(x)\boldsymbol{B}(x)\boldsymbol{U}, \tag{10}$$

from Eqs. (4) and (10), the approximate function of $u(x)$ can be obtained as follows:

$$\begin{aligned}u(x) \approx u^h(x) &= \sum_{k=1}^{m} p_k(x)a_k(x) \\ &= \sum_{i=1}^{N} \Phi_i(x)u_i(t) = \boldsymbol{\Phi}(x)\boldsymbol{U}(t),\end{aligned} \tag{11}$$

where $\boldsymbol{\Phi}$ is a matrix of shape function, defined as

$$\begin{aligned}\boldsymbol{\Phi} &= (\Phi_1(x),\ \Phi_2(x),\ \ldots,\ \Phi_N(x)) \\ &= \boldsymbol{p}^{\mathrm{T}}(x)\boldsymbol{A}^{-1}(x)\boldsymbol{B}(x).\end{aligned} \tag{12}$$

The derivative of the unknown $u(x)$ is approximated by

$$\frac{\partial^k}{\partial x^k}u^h(x) = \sum_{i=1}^{N}\left[\frac{\partial^k}{\partial x^k}\Phi_i(x)\right]u_i(t). \tag{13}$$

The weight function used in the MLS plays a very important role in the EFG method, so the selection of an appropriate weight function is essential. In Eq. (7), $w(x-x_i)$ is the weight function which should be greater than zero within a small neighborhood of x_i, called the domain of influence of node x_i, and in the rest of the solution domain, the weight function should be equal to zero. The cubic spline weight function used in the present analysis can be written as a function of normalized radius r as follows:

$$w(x-x_i) = w(r) = \begin{cases} \dfrac{2}{3} - 4r^2 + 4r^3, & r \le \dfrac{1}{2}, \\ \dfrac{4}{3} - 4r + 4r^2 - \dfrac{4}{3}r^3, & \dfrac{1}{2} < r \le 1, \\ 0, & r > 1 \end{cases} \tag{14}$$

with

$$r = \frac{d_i}{d_{m_i}},\ \ d_i = \|x - x_i\|,\ \ d_{m_i} = d_{\max}c_i, \tag{15}$$

where $d_{\max}$ is a scaling parameter, the distance c_i is chosen so that the matrix $A(x)$ is not singular.

3. Numerical scheme

To establish the numerical approximation scheme, we consider N points: $x_i = (i-1)\Delta x, i = 1,2,3,\ldots,N$, where Δx is the spatial step size and $\Delta x = (b-a)/(N-1)$. $x_1 = a$, $x_N = b$ are the boundary points. The grid points in the time interval $[0,T]$ are labeled as $t_j = n\Delta t, n = 0,1,2,3,\ldots$, where Δt is the time interval.

The EFG method will be implemented for obtaining that discrete equation. To get the discrete equation, we use a weak form of the equation, and employ the penalty method to deal with the boundary conditions.

By making variation, the weak form of Eq. (1) is

$$\int \delta u(u_t - a_0(u_x)^2 - a_0 u u_{xx} - a_1 u_{xx} - a_2 u + a_3 u^2)\,\mathrm{d}\Gamma + \alpha\delta u(u-\overline{u})|\Gamma_u = 0, \tag{16}$$

where $\delta u \in C_0^\infty(\Omega)$ is a variational function, Γ refers to the domain of x, Γ_u refers to the boundaries, and $\overline{u}$ are the values of u in the boundaries. α is a penalty parameter and usually equals $10^3 \sim 10^8$.

Equation (16) can be written as

$$\int \delta u u_t\,\mathrm{d}\Gamma - \int \delta u a_1 u_{xx}\,\mathrm{d}\Gamma - \int \delta u a_2 u\,\mathrm{d}\Gamma - \int \delta u(a_0 u_x^2 + a_0 u u_{xx} - a_3 u^2)\,\mathrm{d}\Gamma + \alpha\delta u(u-\overline{u})|\Gamma_u = 0. \tag{17}$$

According to Eqs. (11) and (13), we can obtain the approximations of u^h, $\partial u^h/\partial x$, $\partial^2 u^h/\partial x^2$, and $\partial u^h/\partial t$

$$u^h = \sum_{i=1}^{N} \Phi_i(x)u_i(t) = \boldsymbol{\Phi}(x)\boldsymbol{U}(t), \tag{18}$$

$$\frac{\partial u^h}{\partial x} = \sum_{i=1}^{N} \frac{\partial \Phi_i(x)}{\partial x}u_i(t) = \sum_{i=1}^{N} \Phi_{i,x}(x)u_i(t) = \boldsymbol{\Phi}_x(x)\boldsymbol{U}(t), \tag{19}$$

$$\frac{\partial^2 u^h}{\partial x^2} = \sum_{i=1}^{N} \frac{\partial^2 \Phi_i(x)}{\partial x^2}u_i(t) = \sum_{i=1}^{N} \Phi_{i,xx}(x)u_i(t) = \boldsymbol{\Phi}_{xx}(x)\boldsymbol{U}(t), \tag{20}$$

$$\frac{\partial u^h}{\partial t} = \sum_{i=1}^{N} \Phi_i(x)\frac{\partial u_i(t)}{\partial t} = \boldsymbol{\Phi}(x)\dot{\boldsymbol{U}}(t), \tag{21}$$

where

$$\boldsymbol{\Phi}_x(x) = \frac{\partial \boldsymbol{\Phi}(x)}{\partial x} = \left(\frac{\partial \Phi_1(x)}{\partial x}, \frac{\partial \Phi_2(x)}{\partial x}, \frac{\partial \Phi_3(x)}{\partial x}, \cdots, \frac{\partial \Phi_N(x)}{\partial x}\right) = (\Phi_{1,x}(x), \Phi_{2,x}(x), \Phi_{3,x}(x), \cdots, \Phi_{N,x}(x)), \tag{22}$$

$$\boldsymbol{\Phi}_{xx}(x) = \frac{\partial^2 \boldsymbol{\Phi}(x)}{\partial x^2} = \left(\frac{\partial^2 \Phi_1(x)}{\partial x^2}, \frac{\partial^2 \Phi_2(x)}{\partial x^2}, \frac{\partial^2 \Phi_3(x)}{\partial x^2}, \cdots, \frac{\partial^2 \Phi_N(x)}{\partial x^2}\right) = (\Phi_{1,xx}(x), \Phi_{2,xx}(x), \Phi_{3,xx}(x), \cdots, \Phi_{N,xx}(x)), \tag{23}$$

and $\boldsymbol{U}(t_i)$ is the parameters when $t = \mathrm{i}\Delta t$.

Substituting Eqs. (18)–(23) into the weak form (17) yields

$$\delta\boldsymbol{U}^{\mathrm{T}}\boldsymbol{G}\dot{\boldsymbol{U}}(t) + \delta\boldsymbol{U}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{U}(t) - \delta\boldsymbol{U}^{\mathrm{T}}\int \boldsymbol{\Phi}^{\mathrm{T}}(a_0(u_x^2) + a_0(uu_{xx}) - a_3(u^2))\,\mathrm{d}\Gamma - \delta\boldsymbol{U}^{\mathrm{T}}\boldsymbol{F} = 0, \tag{24}$$

where $\boldsymbol{G}$ and $\boldsymbol{K}$ are $N\times N$ matrices, whose values of the i-th row, j-th column are respectively

$$\boldsymbol{G}_{ij} = \int \Phi_i(x)\Phi_j(x)\,\mathrm{d}\Gamma, \tag{25}$$

$$\boldsymbol{K}_{ij} = a_1\int \Phi_{i,x}(x)\Phi_{j,x}(x)\,\mathrm{d}\Gamma - a_2\int \Phi_i(x)\Phi_j(x)\,\mathrm{d}\Gamma + \alpha[\Phi_i(x)\Phi_j(x)|_{x=a} + \Phi_i(x)\Phi_j(x)|_{x=b}], \tag{26}$$

$\boldsymbol{F}$ is an $N\times 1$ matrix, whose j-th row is

$$\boldsymbol{F}_j = \alpha(\Phi_j(x)\overline{u}|_{x=a} + \Phi_j(x)\overline{u}|_{x=b}). \tag{27}$$

According to the arbitrariness of $\delta\boldsymbol{U}^{\mathrm{T}}(t)$, Eq. (24) becomes

$$\boldsymbol{G}\dot{\boldsymbol{U}}(t) + \boldsymbol{K}\boldsymbol{U}(t) - \int \boldsymbol{\Phi}^{\mathrm{T}}(a_0(u_x^2) + a_0(uu_{xx}) - a_3(u^2))\,\mathrm{d}\Gamma - \boldsymbol{F} = 0. \tag{28}$$

Forming the time discretization of Eq. (28) with the central difference method, we obtain

$$\boldsymbol{G}\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t} + \boldsymbol{K}\frac{\boldsymbol{U}^{n+1}+\boldsymbol{U}^n}{2} - \int \boldsymbol{\Phi}^{\mathrm{T}}\left(a_0\frac{(u_x^2)^{n+1}+(u_x^2)^n}{2} + a_0\frac{(uu_{xx})^{n+1}+(uu_{xx})^n}{2} - a_3\frac{(u^2)^{n+1}+(u^2)^n}{2}\right)\mathrm{d}\Gamma - \frac{\boldsymbol{F}^{n+1}+\boldsymbol{F}^n}{2} = 0, \tag{29}$$

where

$$\boldsymbol{U}^n = \boldsymbol{U}(n\Delta t) = (u_1(n\Delta t), u_2(n\Delta t), \ldots, u_N(n\Delta t)). \tag{30}$$

The nonlinear termes $(u_x^2)^{n+1}$, $(uu_{xx})^{n+1}$, and $(u^2)^{n+1}$ in the above Eq. (29) are lineralized by using the following term:[32]

$$(uv)^{n+1} = u^{n+1}v^n + u^n v^{n+1} - u^n v^n, \tag{31}$$

then we have

$$(u_x^2)^{n+1} = 2u_x^{n+1}u_x^n - u_x^n u_x^n, \tag{32}$$

$$(uu_{xx})^{n+1} = u^{n+1}u_{xx}^n + u^n u_{xx}^{n+1} - u^n u_{xx}^n, \tag{33}$$

$$(u^2)^{n+1} = 2u^{n+1}u^n - u^n u^n. \tag{34}$$

Substituting Eqs. (32), (33), and (34) into Eq. (29), we have

$$\boldsymbol{G}\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^n}{\Delta t} + \boldsymbol{K}\frac{\boldsymbol{U}^{n+1}+\boldsymbol{U}^n}{2} - \boldsymbol{H}\boldsymbol{U}^{n+1} - \frac{\boldsymbol{F}^{n+1}+\boldsymbol{F}^n}{2} = 0, \tag{35}$$

where $\boldsymbol{H}$ is an $N\times N$ matrix, whose value is

$$\boldsymbol{H} = \int \boldsymbol{\Phi}^{\mathrm{T}}(a_0\boldsymbol{U}^n(\boldsymbol{\Phi}_x^{\mathrm{T}}\boldsymbol{\Phi}) + a_0\boldsymbol{U}^n\frac{\boldsymbol{\Phi}_{xx}^{\mathrm{T}}\boldsymbol{\Phi} + \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}_{xx}}{2}$$

$$-a_3\boldsymbol{U}^n(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}))\mathrm{d}\Gamma. \tag{36}$$

Now we can obtain the discrete equations about $\boldsymbol{U}^{n+1}$,

$$\hbar\boldsymbol{U}^{n+1} = \ell\boldsymbol{U}^n + \boldsymbol{\Theta}, \tag{37}$$

where

$$\hbar = 2\boldsymbol{G} + \Delta t\boldsymbol{K} + 2\Delta t\boldsymbol{H}, \tag{38}$$

$$\ell = 2\boldsymbol{G} - \Delta t\boldsymbol{K}, \tag{39}$$

$$\boldsymbol{\Theta} = \frac{\boldsymbol{F}^{n+1} + \boldsymbol{F}^n}{2}. \tag{40}$$

Equation (37) is the discrete expression of the Fisher equations. Through Eq. (37), we can get any moment parameters $\boldsymbol{U}^{n+1}$, by substituting $\boldsymbol{U}^{n+1}$ into Eq. (11), the approximation of the exact solution of Fisher equations is known.

4. Numercial examples

In this section, some examples are solved with our EFG method, and the Fisher equation is also demonstrated by some examples (its exact solutions are known). In our examples, a regular arrangement of nodes is used, and a cell structure is to set the quadrature cells equal to the intervals between the nodes. In each integration cell, a 4-Gauss quadrature scheme is used to evaluate the stiffness matrix, and the equations' exact solutions are known. For the purpose of the error estimation and convergence studies, we define the error norms L_2 and L_∞ as follows:

$$L_2 = \|u - u^h\|_2 = \sqrt{\Delta x \sum_{j=1}^{N} (u - u^h)^2}, \tag{41}$$

$$L_\infty = \|u - u^h\|_\infty = \max|u - u^h|, \tag{42}$$

where Δx is the spatial step size.

Example 1

We consider Eq. (1) with $a_0 = -1$, $a_1 = 1$, $a_2 = a_3 = 2$, and the boundary conditions

$$u_x\left(-\frac{\pi}{2}, t\right) = 0, \quad u_x\left(\frac{\pi}{2}, t\right) = 0 \tag{43}$$

with the initial condition as

$$u(x,0) = \frac{1}{3}[2 + \sin(x)], \tag{44}$$

and the exact solution as

$$u(x,t) = \frac{1}{3}[2 + \tanh(t) - (1 - \tanh(t))\sin(x)],$$
$$-\frac{\pi}{2} \le x \le \frac{\pi}{2}, t \ge 0. \tag{45}$$

we solve the problem by the present method for penalty factor $\alpha = 1 \times 10^7$, scaling parameter $d_{\max} = 2.5$, the time step length $\Delta t = 0.001$, spatial interval $[-\pi/2, \pi/2]$, and the number of collocation points $N = 16$. The weight function is chosen to be a cubic spline function and the bases are chosen to be linear. The results are listed in Tables 1 and 2 and plotted in Fig. 1.

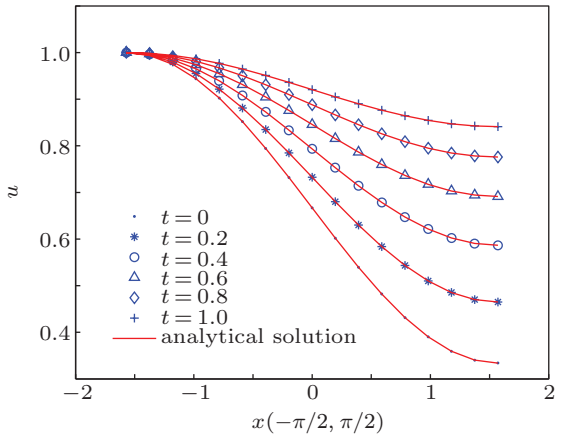


Fig. 1. (color online) Numerical simulations of example 1, when $\Delta t = 0.001$, $N = 16$.

Table 1. Error norms of example 1 with EFG method, when $\alpha = 10^7$, $\Delta t = 0.001$, $N = 16$.

t (EFG)	0.4	0.6	0.8	1.0
L_∞	9.7×10^{-5}	8.1×10^{-5}	5.8×10^{-5}	6.6×10^{-5}
L_2	1.2445×10^{-5}	1.4980×10^{-5}	1.6436×10^{-5}	1.7220×10^{-5}

Table 2. Error norms of example 1 with RBF[33] method which used MQ radial basis function, when $c = 1.17$, $N = 16$.

t (RBF)	0.4	0.6	0.8	1.0
L_∞	4.073×10^{-4}	4.784×10^{-4}	5.536×10^{-4}	6.292×10^{-4}
L_2	1.805×10^{-4}	2.120×10^{-4}	2.453×10^{-4}	2.788×10^{-4}

Example 2

For the choice of parameters $a_0 = 0$, $a_1 = 1$, $a_2 = a_3 = 2$, $(x,t) \in [-15,15] \times [0,2.5]$. The exact solution of the above problem is given by

$$u(x,t) = \left[1 + \exp\left(\sqrt{\frac{1}{3}}x - \frac{5}{3}t\right)\right]^{-2}.$$

Computation was done with scaling parameter $d_{\max} = 4.5$, the time step length $\Delta t = 0.001$, spatial interval $[-15,15]$, and the number of collocation points $N = 30$. In Fig. 2, the exact solutions of this problem are plotted in comparison with numerical results when $t = 0$, 0.5, 1.0, 1.5, 2.0, 2.5. Some results are displayed in Table 3.

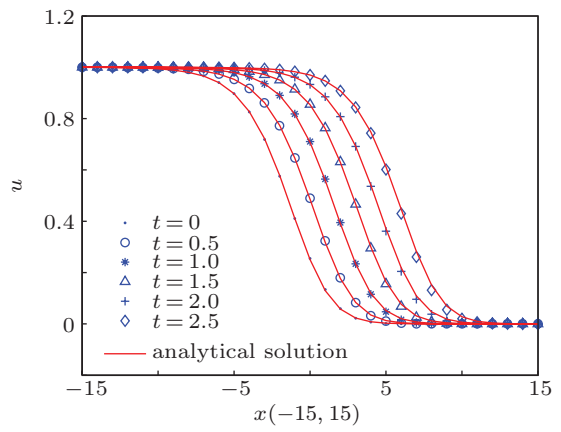


Fig. 2. (color online) Numerical simulation of example 2.

Table 3. Error norms of example 2, when $\alpha = 10^5$, Δt=0.001, $N = 30$.

Time	0.5	1.0	1.5	2.0	2.5
L_∞	5.4×10^{-3}	4.5×10^{-3}	8.05×10^{-3}	1.141×10^{-2}	2.16×10^{-2}
L_2	2.6944×10^{-4}	3.0508×10^{-4}	4.5808×10^{-4}	7.9828×10^{-4}	1.3×10^{-3}

To do numerical simulations, we consider the following two problems with $a_0 = 0$, $a_1 = 0.1$, $a_2 = a_3 = 1$, and $\Delta x = 1/16$, the boundary condition is $u(a,t) = u(b,t) = 0, t > 0$.

Example 3

For Eq. (1), we take the following initial condition $u(x,0) = \mathrm{sech}^2(10x)$, and perform the numerical simulation, the relationship, between u and x are shown in Figs. 3–5.

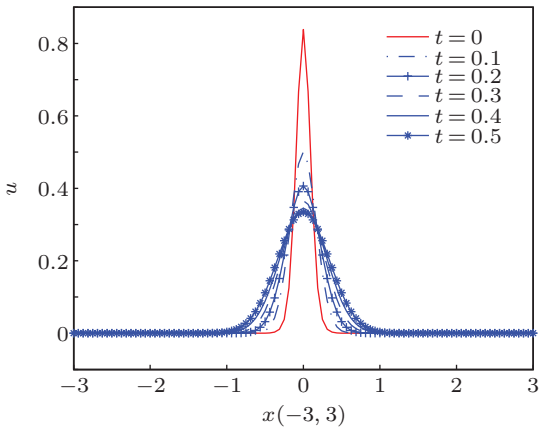


Fig. 3. (color online) The early numerical solution changes of example 3.

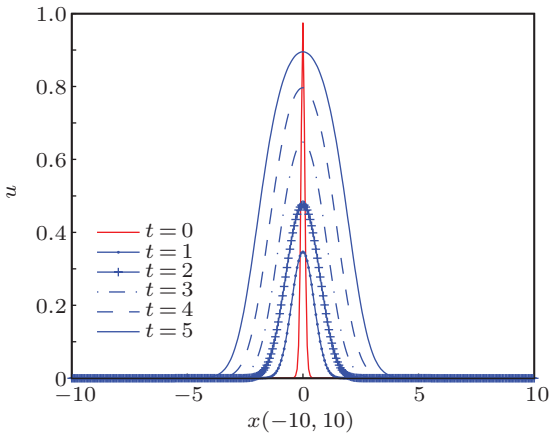


Fig. 4. (color online) The numerical solution changes in a short time of example 3.

From the above figures of example 3, we find that, at the beginning the diffusion term $u_{xx} \ll 0$, and the reaction term $u(1-u)$ is also very small. Therefore the diffusion term controls the reaction term, the function value reaches its maximum value and then drops quickly. As time goes on, the reaction term starts to control the diffusion term, and the function value begins to slowly increase and the peak value also increases gradually until reaching its maximum value $u = 1$, the image becomes more and more gentle.

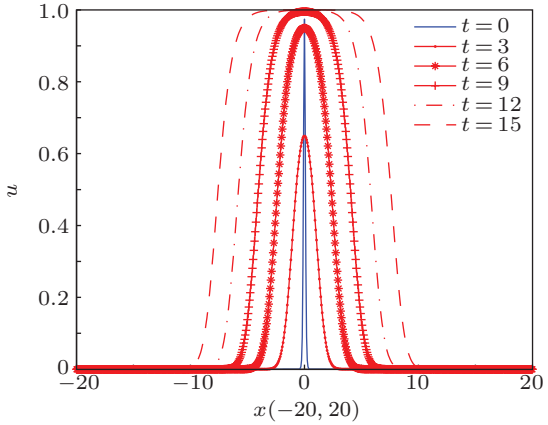


Fig. 5. (color online) The numerical solution changes in a long time of example 3.

Example 4

For Eq. (1), we take the following initial condition

$$u(x,0) = \begin{cases} \mathrm{e}^{10(x+1)}, & x < -1, \\ 1, & -1 \le x \le 1, \\ \mathrm{e}^{-10(x+1)}, & x > 1, \end{cases} \tag{46}$$

and the relationships between u and x are shown in Figs. 6–8.

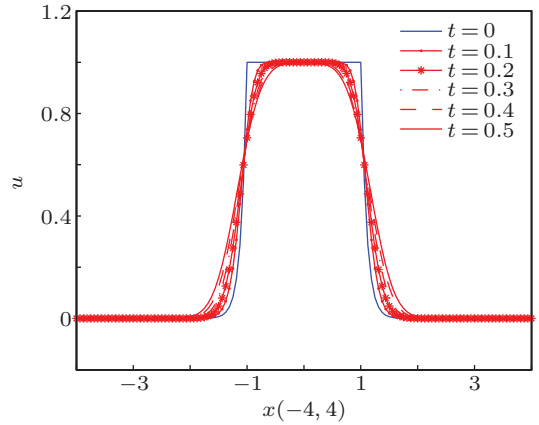


Fig. 6. (color online) The early numerical solution changes of example 4.

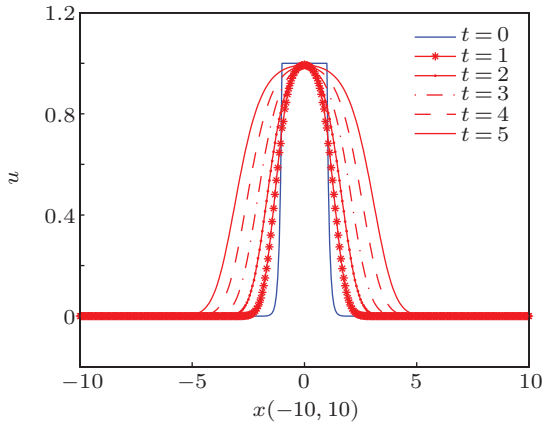


Fig. 7. (color online) The numerical solution changes in a short time of example 4.

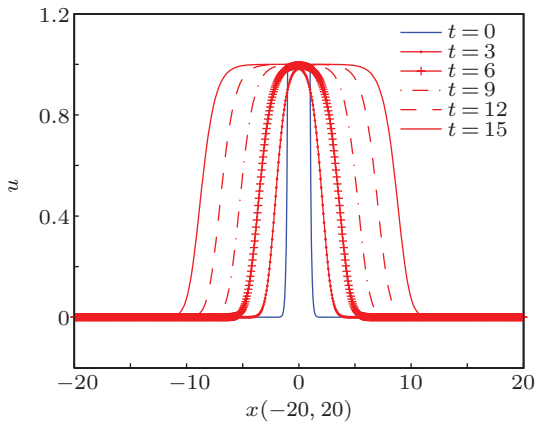


Fig. 8. (color online) The numerical solution changes in a long time of example 4.

The above four examples tell us that, at the beginning, diffusion and reaction terms have a little influence on the solution of the function. With time going on, the influence of reaction term increases bigger and bigger. After quite a long time, the peak value reaches its maximum $u = 1$, the image becomes more and more gentle.

5. Conclusions

In this paper, we have applied the element-free Galerkin (EFG) method to the solution of a class of generalized Fisher equations. We verified the effectiveness of the method by four numerical examples. At the same time we simulated some biological mathematics in reaction diffusion phenomena. The results are consistent with the actual ideal effect. We can obtain more accurate results with relatively less computation time. In comparison with the exact solution and other numerical methods to study Fisher equation, our EFG method has higher accuracy and better convergence.

References

[1] Fisher R A 1930 *The General Theory of Natural Selection* (Oxford: Oxford University Press) pp. 22–47
[2] Vlad M O, Szedlacsek S E, Pourmand N, Cavalli-Sforza L L, Oefner P and Ross J 2005 *Proc. Natl. Acad. Sci. USA* **102** 9848
[3] Ross J 2010 *Proc. Natl. Acad. Sci. USA* **107** 12777
[4] Gazdag J and Canosa J 1974 *J. Appl. Probab.* **11** 445
[5] Qiu Y and Sloan D M 1998 *J. Comput. Phys.* **146** 726
[6] Wazwaz A M and Gorguis A 2004 *Appl. Math. Comput.* **154** 609
[7] Tang S and Weber R O 1991 *J. Austr. Math. Soc. Sci. B* **33** 27
[8] Al-Khaled K 2001 *J. Comput. Appl. Math.* **137** 245
[9] Mickens R E 1994 *Numer. Meth. Part. Differ. Eqns.* **10** 581
[10] El-Azab M S 2007 *Appl. Math. Comput.* **186** 579
[11] Mittal R C and Ceeta A 2010 *Int. J. Comput. Math.* **87** 3039
[12] Sahin A, Dag I and Saka B 2008 *Kybernetes* **37** 326
[13] Macías-Díaz J E and Puri A 2012 *J. Comput. Appl. Math.* **218** 5829
[14] Zhang R P and Zhang L W 2012 *Chin. Phys. B* **21** 090206
[15] Alfaro M and Coville J 2012 *Appl. Math. Lett.* **25** 2095
[16] Belytschko T, Lu Y Y and Gu L 1994 *Int. J. Numer. Methods Engng.* **37** 229
[17] Wang J F, Sun F X and Cheng R J 2010 *Chin. Phys. B* **19** 060201
[18] Cheng R J and Ge H X 2009 *Chin.Phys. B* **18** 4059
[19] Du C 2000 *Comput. Methods Appl. Mesh. Engng.* **182** 89
[20] Kryl P and Belytschko T 1995 *Comput. Mech.* **17** 26
[21] Bai F N, Li D M, Wang J F and Cheng Y M 2012 *Chin. Phys. B* **21** 020204
[22] Cheng Y M, Wang J F and Bai F N 2012 *Chin. Phys B.* **21** 090203
[23] Wang J F, Sun F X and Cheng Y M 2012 *Chin. Phys B* **21** 090204
[24] Cheng Y M, Li R X and Peng M J 2012 *Chin. Phys B* **21** 090205
[25] Cheng Y M and Peng M J 2005 *Sci. China G: Phys. Mech. & Astron.* **35** 435 in Chinese
[26] Cheng Y M and Li J H 2005 *Acta Phys. Sin.* **54** 4463 (in Chinese)
[27] Qin Y X and Cheng Y M 2006 *Acta Phys. Sin.* **55** 3215 (in Chinese)
[28] Cheng R J and Cheng Y M 2007 *Acta Phys. Sin.* **56** 5569 (in Chinese)
[29] Dai B D and Cheng Y M 2007 *Acta Phys. Sin.* **56** 597 (in Chinese)
[30] Cheng R J and Cheng Y M 2008 *Acta Phys. Sin.* **57** 6037 (in Chinese)
[31] Ge H X, Liu Y Q and Cheng R J 2010 *Chin. Phys. B* **21** 010206
[32] Cheng R J and Cheng Y M 2011 *Chin. Phys. B* **20** 070206
[33] Xiao Y M and Wu Y J 2011 *J. Lanzhou University* **47** 2 (in Chinese)