Engineering with Computers (2021) 37:975–990
https://doi.org/10.1007/s00366-019-00867-1

ORIGINAL ARTICLE

An efficient spectral-Galerkin method for solving two-dimensional nonlinear system of advection–diffusion–reaction equations

Farhad Fakhar-Izadi[1]

Received: 4 June 2019 / Accepted: 18 September 2019 / Published online: 1 October 2019

Abstract
The spectral Legendre–Galerkin method for solving a two-dimensional nonlinear system of advection–diffusion–reaction equations on a rectangular domain is presented and compared with analytical solution. The proposed method is based on the Legendre–Galerkin formulation for the linear terms and computation of the nonlinear terms in the Chebyshev–Gauss–Lobatto points. The main difference of the spectral Legendre–Galerkin method presented in the current paper with the classic Legendre–Galerkin method is in treating the nonlinear terms and imposing boundary conditions. Indeed, in the spectral Legendre–Galerkin method the nonlinear terms are efficiently handled using the Chebyshev–Gauss–Lobatto points and also the boundary conditions are imposed strongly as collocation methods. Combination of the proposed method with a semi-implicit time integration method such as the Leapfrog–Crank–Nicolson scheme leads to reducing the complexity of computations and obtaining a linear algebraic system of equations. Efficiency and spectral accuracy of the proposed method are demonstrated numerically by some examples.

Keywords Advection–diffusion–reaction · Legendre–Galerkin method · Chebyshev pseudo-spectral · Leapfrog–Crank–Nicolson time integrator · Fast Fourier transform (FFT) · Schur matrix decomposition method

1 Introduction

In this paper, we consider the following coupled system of advection–diffusion–reaction equations:

$$\begin{cases} \mathcal{L}\mathbf{u} = \mathcal{F}(t,\mathbf{x},\mathbf{u}), & \text{in } \Omega\times(0,T],\\ \mathcal{B}\mathbf{u} = \mathbf{g}, & \text{on } \partial\Omega\times(0,T],\\ \mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}), & \text{in } \Omega\times\{0\}, \end{cases} \tag{1.1}$$

where Ω is a bounded domain in $\mathbb{R}^2$ with a smooth or piecewise smooth boundary, $\mathbf{u} = (u_1, u_2, \dots, u_m)$ is vector of concentration of the physical or biological species, $\mathcal{F} = (F_1, F_2, \dots, F_m)$ is a vector including reaction terms in form of

$$F_i(t,\mathbf{x},\mathbf{u}) = -R_i(u_1,u_2,\dots,u_m) - f_i(\mathbf{x},t), \quad i = 1,2\dots,m,$$

in which

$$R_i(u_1,u_2,\dots,u_m) = \sum_{k,j=1}^{m} b_{i,kj}u_k u_j + \sum_{j=1}^{m} c_{i,j}u_j, \tag{1.2}$$

and $b_{i,kj}$, $c_{i,j}$ are kinetic coefficients. Also in (1.1), $\mathcal{B} = (B_1, B_2, \dots, B_m)$ is vector of boundary operators and $\mathbf{g}$, $\mathbf{u}_0(\mathbf{x})$ are vectors of boundary and initial conditions, respectively. Further, $\mathcal{L} = (\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_m)$ is a m-dimensional vector with components of $\mathcal{L}_k = \frac{\partial}{\partial t} - L_k$ in which L_k is defined as follows

$$L_k = -\mathbf{v}\cdot\nabla u_k + d_k\Delta u_k,$$

where $\mathbf{v}\in\mathbb{R}^2$ is velocity vector in the advection term and $d_i, (i = 1,2,\dots,m)$ are positive parameters of diffusion.

Since R_i in (1.2) involves the product of concentrations, (1.1) is a nonlinear system of differential equations.

Many phenomena in various fields of science such as physics, biology, ecology and biochemistry are modeled by a system of advection–diffusion–reaction. Transport of air pollutants [33, 34, 74, 75], the influences of the unidirectional flow on spatial patterns of community composition and species replacement in the river ecosystems [39], formation of

✉ Farhad Fakhar-Izadi
f.fakhar@aut.ac.ir; ffizadii@gmail.com

[1] Department of Mathematics and Computer Science, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran

complex spatial structures in systems of interacting chemical species [50, 51], groundwater and surface water [76], ash-fall from volcano [45], transport of water vapour in the Earth's atmosphere [63, 84, 85], computer graphics [13] and many other biological patterns are some events which arise through the processes of advection–diffusion–reaction.

The nonlinear system of advection–diffusion–reaction is composed of three distinct terms: advection terms which describe transport of each component due to velocity field $\mathbf{v}$, diffusion (turbulent) terms which are related to random motion of each component due to the turbulent nature of the flow field and reaction terms which describe interaction of the involved species [13]. If $\mathbf{v} = 0$ in (1.1), then we are concerned with a nonlinear system of diffusion–reaction equations which has many applications, particularly, in population dynamics of interacting species like as diffusive Lotka–Volterra system [37].

There are a lot of works in relation to the theoretical and numerical aspects of such systems like (1.1). As mentioned in [61], the system of advection–diffusion–reaction equations (1.1) admits a rich class of solutions ranging from a relatively simple linear advection to nonlinear reaction–diffusion waves leading to the formation of complex, spatially nonhomogeneous patterns. Unique solvability of quasi-linear parabolic system is analyzed in [32]. An optimal control problem for the Lotka–Volterra system with diffusion is considered in [28] and also it has been proved that this system has a unique positive solution with some bounded properties. Existence of the bistable traveling wave solution of reaction–diffusion equations that model the interaction of n mutualist species is proved in [29] and also uniqueness of their bistable traveling wave solution is shown by homotopy approach incorporated with the Liapunov–Schmidt method. Furthermore, [39] gives some analytical discussion on advection–diffusion–reaction equations, coupled by Lotka–Volterra interaction terms, about the effects of dispersal patterns on competing species. Spatial structures and front propagation of the reaction–diffusion systems are considered in [6]. An advection–diffusion–reaction model for the pattern formulation is proposed in [48]. Also behavior of the model, especially the effects of the advection term on a simple reaction–diffusion system is studied by numerical calculation.

Numerical solution of nonlinear advection–diffusion–reaction systems also has been an interesting subject for many researchers and, therefore, some numerical methods such as finite difference, spectral methods, finite elements and finite volume are applied for solving them. Finite difference methods due to their simplicity are applied for system of advection–diffusion–reaction equations in different papers [7, 20, 25, 36, 47, 52–56, 80, 81]. In [82] high-order compact finite difference is proposed for (1.1) and Richardson extrapolation is also used for obtaining fourth-order accuracy in time. A compact finite difference method is presented in [36] using Crank–Nicolson technique in time discretization and fourth-order Padé approximation to space derivative. Asymptotic behavior of the finite difference solutions of nonlinear integro-differential reaction–diffusion equations is considered in [83]. Nonstandard finite difference schemes for one-space dimension single nonlinear reaction–diffusion partial differential equation with linear advection are extended in [46]. [86] is devoted to generalize quasilinearization method for solving nonlinear advection–diffusion–reaction systems. A finite volume algorithm which first approximates convection and diffusive fluxes and then solves the resulting ODE system is proposed in [61] for the solution of the advection–diffusion–reaction equations on the sphere. A fully discrete H^1-Galerkin method with quadrature is proposed in [15] for nonlinear single parabolic advection–diffusion–reaction equation. Also in [24] high-order finite volume scheme is derived for nonlinear system of advection–diffusion–reaction with stiff algebraic source terms. The system (1.1) is introduced in [37] and a finite element method together with its error analysis is considered for it. A Legendre spectral element method is proposed in [8] for one-dimensional predator–prey system on a large spatial domain. An efficient pseudo-spectral Legendre Galerkin method for solving a nonlinear partial integro-differential equation arising in population dynamics is introduced in [12]. An implicit–explicit Runge–Kutta–Chebyshev (RKC) method which treats diffusion and advection terms explicitly and the highly stiff reaction terms implicitly is proposed in [79]. In [4] a second-order exponential integrator for semidiscretized advection–diffusion–reaction equations is obtained that is five times faster than a classical second-order implicit solver. An operator splitting algorithm which splits homogeneous equations into advection, diffusion and reactions and then solves them by backward method of characteristic, finite element method and an explicit Runge–Kutta, respectively, is presented in [31] for a system of one-dimensional advection–diffusion–reaction equations. Also in [13] a numerical operator splitting for time integration of three-dimensional advection–diffusion–reaction system is implemented and three different methods of second-order accuracy are derived for solving, separately, each term that appeared in the model. Finite element solution of reaction–diffusion equations with advection and nonconforming finite element methods with subgrid viscosity for solving single advection–diffusion–reaction equation are considered in [38] and [11], respectively. Discontinuous hp-finite element methods for second-order elliptic and parabolic advection–reaction equations are considered in [26]. Performance of three different time integration methods, i.e. Euler backward, second-order Rosenbrock and implicit–explicit Runge–kutta–Chebyshev is examined in [78]. More details about time integration of (1.1) are presented in [30]. A multiscale variational method is proposed in [27] to solve the advection–diffusion equation. A couple implicit integration factor (IIF) method and its higher dimensional

analog compact IIF with weighted essentially non-oscillatory (WENO) methods using the operator splitting approach is presented in [88] to solve advection–diffusion–reaction equations. In [57] numerical method using establishing a difference scheme based on singular perturbed theory and Green's function is investigated for the system of reaction–diffusion equations in one-dimension and some estimates of the derivatives of the solution are obtained.

Depending on various applications, Eq. (1.1) can be imposed with different boundary conditions. In this paper we apply spectral Legendre–Galerkin method for nonlinear system (1.1) on a rectangular domain Ω with homogeneous Dirichlet boundary conditions. Nonhomogeneous essential boundary conditions can be achieved by lifting an arbitrary function satisfying the boundary conditions and modifying the right-hand side of the equation.

Galerkin methods [2, 5, 16, 22, 77] are one of the most important weighted residual methods which are able to solve many kinds of operator equations and have been applied by many authors in various fields of science and engineering, see, e.g., [3, 17, 18, 21, 23, 44, 69–72, 87].

The key property of the Galerkin approach is selecting a finite dimensional subspace of the Hilbert space (trial function space) for approximation of solution and imposing orthogonality relation of the obtained error to a finite dimensional space (test function space) which is the same to the trial space. If the trial and test spaces in the Galerkin method are different, the obtained method is called Petrov–Galerkin [2]. The main deficiency of the Galerkin methods is that their implementation for the nonlinear problems is rather difficult. However, using orthogonal polynomials as a basis for the approximation (trial) space makes these methods easier to be implemented by no need to compute the integral terms arising in inner product for the linear terms. Chebyshev–Galerkin methods due to the non-uniform weight function for the Chebyshev polynomials give rise to a complex and non-symmetric system of equations [66]. Therefore, in this paper we use the Legendre polynomials as the basis of the approximation space in the Galerkin method. These polynomials with the unit weight function are very efficient in the computations. For approximating the nonlinear terms in the Galerkin method, a nice approach based on pseudo-spectral methods was proposed in [66]. However, the idea of using Chebyshev nodes in the Legendre collocation method was first introduced by Don et. al [9] for the parabolic and hyperbolic equations, but implementation of these methods in [66] and [9] is different. For using this approach three essential steps must be performed in the Legendre–Galerkin method. The first step is computing the nonlinear terms at the Chebyshev–Gauss–Lobatto points; the second step is using the fast Fourier transform (FFT) between the physical and spectral Chebyshev spaces and the third step is applying a proper algorithm for finding the coefficients of the Legendre expansion from Chebyshev one [66].

The Legendre–Gauss–Lobatto points are not available in an explicit form, so there is not any fast Legendre transform between function values at the Legendre–Gauss–Lobatto points (physical space) and the Legendre expansion coefficients (spectral space) and also computational complexity of this process is $O(N^2)$ [1, 66, 67]. On the other hand, the Chebyshev–Gauss–Lobatto points are given in the explicit form and the fast Chebyshev transform or the fast Fourier transform (FFT) between their physical and spectral spaces can be efficiently performed in $O(N\log_2 N)$ operations. Therefore, weakness of the Legendre–Galerkin method for approximating the nonlinear terms is treated using the Chebyshev–Gauss–Lobatto points. Indeed, the Chebyshev pseudo-spectral Legendre–Galerkin method which encompasses advantages of the both Chebyshev–Galerkin and Legendre–Galerkin methods is used in this paper. The contribution of paper is generalization of the Chebyshev–Legendre–Galerkin method to a two-dimensional nonlinear system of equations. In the proposed method, the linear parts of the problem are approximated by the Legendre–Galerkin method and the nonlinear parts are computed by the interpolation operator at the Chebyshev–Gauss–Lobatto nodes. Therefore, easy computation of nonlinear terms is combined with the good stability properties of the Legendre–Galerkin methods. Combination of the Chebyshev pseudo-spectral Legendre–Galerkin method with a classic semi-implicit time integration scheme such as Leapfrog–Crank–Nicolson which treats the linear parts implicitly and the nonlinear parts explicitly yields a matrix equation that can be converted to a linear algebraic system of equations at each time step . Also in some cases of (1.1), including pure reaction–diffusion system (i.e. $\mathbf{v} = 0$) or non-diffusive case of it (i.e. $d_1 = 0, d_2 = 0$), the resulting matrix equation, due to symmetry of the coefficient matrix, can be solved efficiently by the Schur matrix decomposition or generalized eigenvalue problem methods.

The Chebyshev–Legendre–Galerkin or Petrov–Galerkin methods for solving many kinds of problems have been applied by some authors. In [66] Shen introduced this kind of Chebyshev–Legendre–Galerkin method and applied it for the elliptic problems. Also he proposed an efficient algorithm for transforming between coefficients of the Chebyshev expansion and of the Legendre expansion. The class of spectral Galerkin methods based on Legendre and Chebyshev polynomials for direct solution of the second- and fourth-order equations are proposed in [67, 68], respectively. The Chebyshev pseudo-spectral Legendre–Petrov–Galerkin method is considered for the modified Kawahara equation in [65] and fourth-order differential equations in [73]. Also in [43] the Legendre–Petrov–Galerkin method is applied for the Korteweg-de Vries equation and in [42] this method is proposed for the third-order differential equations. A Legendre Galerkin–Chebyshev collocation method is developed

for Burgers-like equations in [35]. Second-order equations are solved by the Chebyshev–Petrov–Galerkin method in [10]. Chebyshev–Legendre spectral method is presented for the nonlinear conservative laws in [40, 41].

The remainder of the paper is organized as follows: In the next section we propose the Chebyshev pseudo-spectral Legendre–Galerkin method for solving nonlinear system (1.1) on a rectangular domain Ω. Some numerical results are presented in Sect. 3 which demonstrate the spectral accuracy and efficiency of the proposed method. Finally Sect. 4 includes some concluding remarks.

2 Chebyshev–Legendre Galerkin method

In this section the Chebyshev–Legendre Galerkin method is proposed for a two-dimensional nonlinear system (1.1) on a rectangular domain Ω with homogeneous boundary condition as follows:

$$\begin{cases} \mathcal{L}\mathbf{u} = \mathcal{F}(t,\mathbf{x},\mathbf{u}), & \text{in } \Omega\times(0,T],\\ \mathbf{u} = 0, & \text{on } \partial\Omega\times(0,T],\\ \mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}), & \text{in } \Omega\times\{0\}, \end{cases} \tag{2.1}$$

where without loss of generality we suppose that $\Omega = I^2$, where $I = (-1,1)$ and operators of $\mathcal{L}$, and $\mathcal{F}$ are defined similar to the previous.

In the case of nonhomogeneous boundary conditions in Eq. (1.1), we can decompose the exact solution into a known lifted function $\mathbf{u}^d$ which satisfies at the boundary conditions and an unknown function $\mathbf{u}^h$ which is zero on the boundaries. By setting $\mathbf{u} = \mathbf{u}^h + \mathbf{u}^d$ such that $\mathcal{B}\mathbf{u}^d = \mathbf{g}$ and $\mathcal{B}\mathbf{u}^h = 0$, we have

$$\begin{cases} \mathcal{L}\mathbf{u}^h = \mathcal{F}^*(t,\mathbf{x},\mathbf{u}^h), & \text{in } \Omega\times(0,T],\\ \mathcal{B}\mathbf{u}^h = 0, & \text{on } \partial\Omega\times(0,T],\\ \mathbf{u}^h(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) - \mathbf{u}^d(\mathbf{x},0), & \text{in } \Omega\times\{0\}, \end{cases} \tag{2.2}$$

in which $\mathcal{F}^*(t,\mathbf{x},\mathbf{u}^h) = \mathcal{F}(t,\mathbf{x},\mathbf{u}^h + \mathbf{u}^d) - \mathcal{L}\mathbf{u}^d$.

2.1 Preliminaries

If $L_n(x)$ denotes the nth Legendre polynomial which has orthogonal property with respect to L^2 inner product on the interval $[-1,1]$ with the unit weight function, then the Legendre-Gauss quadrature formula is as follows:

$$\int_{-1}^{1} f(y)dy \approx \sum_{j=0}^{N} f(y_j)w_j,$$

where distinct nodes $y_0 < y_1 < \cdots < y_N$ are $N+1$ roots of $L_{N+1}(x)$ in $(-1,1)$ and $\{w_j\}_{j=0}^{N}$ are the corresponding weights. While, explicit formulae for the quadrature nodes are not known, the quadrature weights can be expressed by the following relation:

$$w_j = \frac{2}{(1-y_j^2)[L'_{N+1}(y_j)]^2}, \quad j = 0,1,\ldots,N.$$

In the rest of the current paper we need to introduce the Sobolev space. Let Ω be an open subset in $\mathbb{R}^n$ and $w(\mathbf{x})$ be a positive weight function on Ω. For any non-negative integer m, $H_w^m(\Omega)$ is the weighted Sobolev space with norm $\|.\|_{\Omega,m,w}$ that is defined as

$$H_w^m(\Omega) = \{u \in L_w^2(\Omega);\, D^\alpha u \in L_w^2(\Omega),\ \forall|\alpha| \le m\},$$

where for a given multi-index $\alpha = (\alpha_1,\ldots,\alpha_n) \in \mathbb{Z}_+^n$, we have

$$D^\alpha u = \frac{\partial^{|\alpha|}u}{\partial x_1^{\alpha_1},\ldots,\partial x_n^{\alpha_n}}, \quad |\alpha| = \sum_{i=1}^{n}\alpha_i.$$

In particular, the norm and inner product of $L_w^2(\Omega)$ $(= H_w^0(\Omega))$ are denoted by $\|.\|_{\Omega,w}$, and $\langle,\rangle_{\Omega,w}$, respectively. We will drop the subscript w in all notations, whenever $w \equiv 1$.

2.2 Implementation

We suppose $V_N^0 = \mathbb{P}_N(\Omega) \cap H_0^1(\Omega)$ is the trial (test) function space, in which $\mathbb{P}_N(\Omega)$ is the space of all algebraic two variables polynomials of degree at most N with respect to each variable and

$$H_0^1(\Omega) = \{v \in H^1(\Omega); v|_{\partial\Omega} = 0\}.$$

From computational aspects and for the sake of efficiency it is essential to use the combinations of orthogonal polynomials (with respect to the inner product $\langle,\rangle_\Omega$) as the basis of V_N^0. With this choice, the inner products arisen in the linear terms of the weak form are computed easily using orthogonality properties of basis and without any integration. For the nonlinear terms, as mentioned earlier, we use the Chebyshev–Gauss–Lobatto points which enable us to approximate these terms by the Legendre polynomials expansion in the case of the Legendre–Galerkin method [66]. The natural choice for basis of V_N^0 is the tensor product of basis functions in the one dimensional case. Indeed, it can be trivially proved that

$$V_N^0 = span\{\psi_{m,n}(x,y);\, 0 \le m,n \le N-2\},$$

in which $\psi_{m,n}(x,y) = \phi_m(x)\phi_n(y)$ and

$$\phi_n(x) = \frac{1}{\sqrt{4n+6}}(L_n(x) - L_{n+2}(x)).$$

Multiplying both sides of ith equation of (2.1) by $w \in H_0^1(\Omega)$ and integration by parts over Ω, we have the following weak form of (2.1):

$$\begin{cases} \langle \frac{\partial u_i}{\partial t}, w\rangle_\Omega + \mathbf{v}\cdot\langle \nabla u_i, w\rangle_\Omega + d_i\langle \nabla u_i, \nabla w\rangle_\Omega + \langle R_i(\mathbf{u}), w\rangle_\Omega + \langle f_i, w\rangle_\Omega = 0, \quad t\in(0,T], \\ \langle u_i(0), w\rangle_\Omega = \langle u_{0i}, w\rangle_\Omega, \end{cases}$$

where $u_i(t) \in H_0^1(\Omega)$ and $1 \le i \le m$.

Also the semi-discrete Chebyshev–Legendre–Galerkin method for problem (2.1) is to find $U_i^N(t) \in V_N^0$ such that for any $w \in V_N^0$

$$\begin{cases} \langle \frac{\partial U_i^N}{\partial t}, w\rangle_\Omega + \mathbf{v}\cdot\langle \nabla U_i^N, w\rangle_\Omega + d_i\langle \nabla U_i^N, \nabla w\rangle_\Omega \\ +\langle I_N^C R_i(\mathbf{U}^N(t)), w\rangle_\Omega + \langle I_N^C f_i(t), w\rangle_\Omega = 0, \qquad t\in(0,T], \quad i=1,2,\dots,m, \\ \langle U_i^N(0), w\rangle_\Omega = \langle I_N^C u_{0i}, w\rangle_\Omega, \end{cases} \tag{2.3}$$

in which I_N^C is the interpolation operator at $\{(\theta_i,\theta_j)\}_{i,j=0}^{N-2}$ on $\Omega = I^2$ where $\theta_i = \cos(\frac{i\pi}{N-2})$ are the Chebyshev–Gauss–Lobatto nodes on I. Also $\mathbf{U}^N(t)$ is the vector of approximate solutions including $U_i^N(t)$, i.e.

$$\mathbf{U}^N(t) = \begin{pmatrix} U_1^N(t) \\ U_2^N(t) \\ \vdots \\ U_m^N(t) \end{pmatrix} \approx \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_m(t) \end{pmatrix}.$$

in which $\hat{t}_k$, and $\tilde{t}_k$ stand for the following definition:

$$h(\hat{t}_k) = \frac{h(t_{k+1}) - h(t_{k-1})}{2\delta t}, \quad h(\tilde{t}_k) = \frac{h(t_{k+1}) + h(t_{k-1})}{2}.$$

Now for each $1 \le i \le m$, the terms on the right-hand side of (2.4) which include the interpolation operator I_N^C must be approximated by the Legendre expansion. This work is done by employing the nice and efficient algorithm proposed in [66]. This algorithm for $R_i(\mathbf{U}^N(t_k))$ is as follows:

Algorithm 1 Finding coefficients of the Legendre expansion

1: Compute $R_i(\mathbf{U}^N(t_k))$ at the Chebyshev-Gauss-Lobatto points $\{(\theta_i,\theta_j)\}_{i,j=0}^{N-2}$
2: Use the inverse FFT to obtain the Chebyshev expansion coefficients of $R_i(\mathbf{U}^N(t_k))$ from its values obtained by step 1.
3: Apply a suitable algorithm to find the Legendre expansion coefficients of $R_i(\mathbf{U}^N(t_k))$ from its Chebyshev coefficients obtained by step 2.

For linearization of the aforementioned weak form in (2.3), we use the semi-implicit second-order Leap-frog–Crank–Nicolson time integration method which treats the linear part of equation implicitly and the nonlinear part explicitly. Also this semi-implicit method has larger stability region than explicit ones and has been used extensively in the computational applied sciences. Therefore, for a given time step δt, the fully discrete Chebyshev–Legendre–Galerkin method for problem (2.1) is to find $U_i^N(t_k) \in V_N^0$ such that for any $w \in V_N^0$

$$\begin{cases} \langle U_i^N(\hat{t}_k), w\rangle_\Omega + \mathbf{v}\cdot\langle \nabla U_i^N(\tilde{t}_k), w\rangle_\Omega + d_i\langle \nabla U_i^N(\tilde{t}_k), \nabla w\rangle_\Omega \\ +\langle I_N^C R_i(\mathbf{U}^N(t_k)), w\rangle_\Omega + \langle I_N^C f_i(\tilde{t}_k), w\rangle_\Omega = 0, \qquad t\in(0,T], \quad i=1,2,\dots,m, \\ \langle U_i^N(t_1), w\rangle_\Omega = \langle I_N^C(u_{0i} + \delta t \frac{\partial u_i}{\partial t}|_{t=0}), w\rangle_\Omega, \\ \langle U_i^N(0), w\rangle_\Omega = \langle I_N^C u_{0i}, w\rangle_\Omega, \end{cases} \tag{2.4}$$

For finding coefficients of the Legendre expansion of $f_i(\tilde{t}_k)$, and $u_{0i} + \delta t \frac{\partial u_i}{\partial t}|_{t=0}$, for each i, a similar algorithm can be used.

As mentioned in step 3 of the presented algorithm, we must apply a suitable algorithm to find coefficients of the Legendre expansion from coefficients of the Chebyshev one. Alpert and Rohklin [1] introduced an $O(N)$ algorithm which is efficient for large N and is based on the fast multipole method [19]. But Shen [66] proposed a similar algorithm

whose complexity is almost $\min\{\frac{1}{2}N^2, CN\}$ in one-dimensional space where C is a large constant. Hence, for moderate values of N, Shen's algorithm is more efficient than the algorithm introduced in [1]. In one-dimensional case, the second and third steps of Algorithm 1 can be done in about $(\frac{5}{2}N\log_2 N + 4N) + \min\left\{\frac{1}{2}N^2, CN\right\} \sim O(N\log_2 N)$ as computed in [66]. Since multi-dimensional transform in the tensor product form is performed through a sequence of one-dimensional transforms, the two-dimensional version of Algorithm 1 can be done in $O(N^2\log_2 N)$ operations for $N > m$.

In the following, we describe extension of the Shen's algorithm [66] to the two-dimensional case which can be applied in step 3 of the presented algorithm for finding the Legendre expansion coefficients from the coefficients of Chebyshev expansion.

If $\tilde{b}_{i,j} = \langle L_i, T_j\rangle_\Omega$, then the recurrence relation for computing $\tilde{b}_{i,j}$ is proposed in [66] as follows:

$$\tilde{b}_{i,j+1} = \frac{2i+2}{2i+1}\tilde{b}_{i+1,j} + \frac{2i}{2i+1}\tilde{b}_{i-1,j} - \tilde{b}_{i,j-1}, \tag{2.5}$$

and with defining $b_{i,j} = (i+\frac{1}{2})\tilde{b}_{i,j}$, the vector of coefficients of Legendre expansion, $\mathbf{g}^*$, is obtained as follows:

$$\mathbf{g}^* = B\mathbf{f},$$

where $B = (b_{i,j})$, and $\mathbf{f}$ is vector of coefficients of Chebyshev expansion.

Now suppose

$$\sum_{i,j=0}^{N-2} h_{i,j}T_i(x)T_j(y) = \sum_{k,l=0}^{N-2} m_{k,l}L_k(x)L_l(y),$$

and $H = (h_{i,j})$, $M = (m_{k,l})$. Multiplying both sides of the above relation in $L_p(x)L_q(y)$ and then integrating on Ω with respect to the Legendre weight, we obtain

$$\sum_{i,j=0}^{N-2} h_{i,j}\langle T_iT_j, L_pL_q\rangle_\Omega = \sum_{k,l=0}^{N-2} m_{k,l}\langle L_kL_l, L_pL_q\rangle_\Omega,$$

or in an equivalent form we can write

$$\sum_{i,j=0}^{N-2} h_{i,j}\langle T_i, L_p\rangle_I\langle T_j, L_q\rangle_I = \sum_{k,l=0}^{N-2} m_{k,l}\langle L_k, L_p\rangle_I\langle L_l, L_q\rangle_I. \tag{2.6}$$

If $c_{i,j} = \langle L_i, L_j\rangle_I$, then the matrix form of (2.6) is as follows:

$$\tilde{B}H\tilde{B}^t = CMC,$$

in which $C = (c_{i,j})$, $\tilde{B} = (\tilde{b}_{i,j})$ where $\tilde{b}_{i,j}$ will be obtained by the recurrence relation presented in (2.5) and $\tilde{B}^t$ is transpose of $\tilde{B}$. Also, due to the orthogonality property of the Legendre polynomials, C is a diagonal matrix and its inverse will be easily computed. Therefore, the Legendre expansion coefficients can be found from Chebyshev expansion coefficients by

$$M = C^{-1}\tilde{B}H\tilde{B}^tC^{-1}. \tag{2.7}$$

By expanding

$$U_i^N(x,y,t) = \sum_{k,j=0}^{N-2} \hat{u}_{i,k,j}^N(t)\psi_{k,j}(x,y), \quad i = 1,2\ldots,m,$$

and take $w = \psi_{l,m}$ in (2.4); then we have

$$\begin{cases} \sum_{k,j=0}^{N-2} \hat{u}'^N_{i,k,j}(t)\langle\psi_{k,j},\psi_{l,m}\rangle_\Omega + \mathbf{v}\cdot\left(\sum_{k,j=0}^{N-2} \hat{u}^N_{i,k,j}(t)\langle\nabla\psi_{k,j},\psi_{l,m}\rangle_\Omega\right) + d_i\sum_{k,j=0}^{N-2} \hat{u}^N_{i,k,j}(t)\langle\nabla\psi_{k,j},\nabla\psi_{l,m}\rangle_\Omega \\ \quad + \sum_{k,j=0}^{N-2} \mathscr{R}^*_{i,k,j}(t)\langle L_kL_j,\psi_{l,m}\rangle_\Omega + \sum_{k,j=0}^{N-2} \mathscr{F}^*_{i,k,j}(t)\langle L_kL_j,\psi_{l,m}\rangle_\Omega = 0, \\ \sum_{k,j=0}^{N-2} \hat{u}^N_{i,k,j}(t_1)\langle\psi_{k,j},\psi_{l,m}\rangle_\Omega = \sum_{k,j=0}^{N-2} \mathscr{K}^*_{i,k,j}\langle L_kL_j,\psi_{l,m}\rangle_\Omega, \qquad i = 1,2,\ldots,m, \\ \sum_{k,j=0}^{N-2} \hat{u}^N_{i,k,j}(0)\langle\psi_{k,j},\psi_{l,m}\rangle_\Omega = \sum_{k,j=0}^{N-2} \mathscr{I}^*_{i,k,j}\langle L_kL_j,\psi_{l,m}\rangle_\Omega, \end{cases}$$

or in an equivalent form

$$\begin{cases} \sum_{k,j=0}^{N-2} \hat{u}'^{N}_{i,k,j}(t)\langle\phi_k,\phi_l\rangle_I\langle\phi_j,\phi_m\rangle_I + v_1\sum_{k,j=0}^{N-2}\hat{u}^{N}_{i,k,j}(t)\langle\phi'_k,\phi_l\rangle_I\langle\phi_j,\phi_m\rangle_I + v_2\sum_{k,j=0}^{N-2}\hat{u}^{N}_{i,k,j}(t)\langle\phi_k,\phi_l\rangle_I\langle\phi'_j,\phi_m\rangle_I \\ +d_i\left(\sum_{k,j=0}^{N-2}\hat{u}^{N}_{i,k,j}(t)\langle\phi'_k,\phi'_l\rangle_I\langle\phi_j,\phi_m\rangle_I + \sum_{k,j=0}^{N-2}\hat{u}^{N}_{i,k,j}(t)\langle\phi_k,\phi_l\rangle_I\langle\phi'_j,\phi'_m\rangle_I\right) \\ +\sum_{k,j=0}^{N-2}\mathscr{R}^{*}_{i,k,j}(t)\langle L_k,\phi_l\rangle_I\langle L_j,\phi_m\rangle_I + \sum_{k,j=0}^{N-2}\mathscr{F}^{*}_{i,k,j}(t)\langle L_k,\phi_l\rangle_I\langle L_j,\phi_m\rangle_I = 0, \quad i=1,2,\dots,m, \\ \sum_{k,j=0}^{N-2}\hat{u}'^{N}_{i,k,j}(t_1)\langle\phi_k,\phi_l\rangle_I\langle\phi_j,\phi_m\rangle_I = \sum_{k,j=0}^{N-2}\mathscr{K}^{*}_{i,k,j}\langle L_k,\phi_l\rangle_I\langle L_j,\phi_m\rangle_I, \\ \sum_{k,j=0}^{N-2}\hat{u}^{N}_{i,k,j}(0)\langle\phi_k,\phi_l\rangle_I\langle\phi_j,\phi_m\rangle_I = \sum_{k,j=0}^{N-2}\mathscr{I}^{*}_{i,k,j}\langle L_k,\phi_l\rangle_I\langle L_j,\phi_m\rangle_I, \end{cases}$$

in which

where matrices $\mathscr{R}^*_i(t)$ and $\mathscr{F}^*_i(t)$ are Legendre expansion coefficients of $R_i(\mathbf{U}^N(t))$ and $f_i(t)$, respectively, at time t and $\mathscr{K}^*_i$ is matrix of Legendre expansion coefficients of $u_{0i} + \delta t\frac{\partial u_i}{\partial t}|_{t=0}$, and $\mathscr{I}^*_i$ is matrix of u_{0i} that all will be obtained by (2.7).

Let us denote $\alpha_{m,n} = \langle\phi_n,\phi_m\rangle_\Omega$, $\beta_{m,n} = \langle\nabla\phi_n,\nabla\phi_m\rangle_\Omega$, $\eta_{m,n} = \langle\nabla\phi_n,\phi_m\rangle_\Omega$, and $\gamma_{m,n} = \langle L_n,\phi_m\rangle_\Omega$; then for $0 \le m,n \le N-2$, we get

$$\alpha_{m,n} = \begin{cases} c_n c_m(\frac{2}{2n+1} + \frac{2}{2n+5}), & n = m, \\ -\frac{2c_n c_m}{2m+5}, & n = m+2, \\ -\frac{2c_n c_m}{2n+5}, & n = m-2, \\ 0, & \text{otherwise}, \end{cases}$$

$$\beta_{m,n} = \begin{cases} 1, & n = m, \\ 0, & \text{otherwise}, \end{cases}$$

$$\eta_{m,n} = \begin{cases} -2c_m c_n, & n = m-1, \\ 2c_m c_n, & n = m+1, \\ 0, & \text{otherwise}, \end{cases}$$

$$\gamma_{m,n} = \begin{cases} \frac{2c_m}{2n+1}, & n = m, \\ -\frac{2c_n}{2m+5}, & n = m+2, \\ 0, & \text{otherwise}, \end{cases}$$

where $c_n = \frac{1}{\sqrt{4n+6}}$.

Therefore, with defining $V^N_i(t) = (\hat{u}^N_{i,k,j}(t))_{k,j}$, which encompasses coefficients in approximation of u_i as a matrix, and supposing $X^k_i = V^N_i(t_{k+1})$, and $Y^k_i = V^N_i(t_{k-1})$, at each time step in (2.4) we must solve m matrix equations as follows:

$$AX^k_iA + \tau_1 DX^k_iA + \tau_2 AX^k_iD^t + \sigma_i(X^k_iA + AX^k_i) = W_i, \quad i = 1,2\dots,m, \tag{2.8}$$

$$W_i = AY^k_iA - \tau_1 DY^k_iA - \tau_2 AY^k_iD^t - \sigma_i(Y^k_iA + AY^k_i) - \delta tC(2\mathscr{R}^*_i(t_k) + \mathscr{F}^*_i(t_{k-1}) + \mathscr{F}^*_i(t_{k+1}))C^t,$$

is known for each i, at time step t_{k+1} and $\tau_1 = v_1\delta t$, $\tau_2 = v_2\delta t$, $\sigma_i = d_i\delta t$. Also matrices $A = (\alpha_{m,n})$, $D = (\eta_{m,n})$, $C = (\gamma_{m,n})$ are defined in the previous. In other words, at any time step and for each i we are dealing with a matrix equation in the form

$$AXA + \tau_1 DXA + \tau_2 AXD^t + \sigma(XA + AX) = W, \tag{2.9}$$

where W is a known matrix. We can also rewrite the above matrix equation as a standard linear system using the tensor product notation as follows:

$$(A\otimes A + \tau_1 D\otimes A + \tau_2 A\otimes D + \sigma I\otimes A + \sigma A\otimes I)\mathbf{x} = \mathbf{w},$$

where $\mathbf{x}$, and $\mathbf{w}$ are vectorization of unknown coefficients matrix X and known matrix W, respectively. Indeed, we have $\mathbf{x} = vec(X)$, $\mathbf{w} = vec(W)$. Also $\otimes$ denotes the tensor product of matrices, i.e. $A\otimes D = (\alpha_{i,j}D)_{i,j=0,1,\dots,N-2}$.

Remark If we are concerned with a diffusion–reaction system, i.e. the velocity vector is equal to zero ($\mathbf{v} = 0$) in (1.1), then (2.8) can be solved efficiently by the matrix decomposition methods. Considering that A is symmetric and using the real version of the Schur decomposition, there exists an orthogonal matrix Q such that

$$Q^tAQ = T,$$

in which T is a diagonal matrix formed from the eigenvalues of A. Therefore, with defining $X = ZQ^t$ and multiplying both sides of (2.9) by Q and also using orthogonal property of Q, we obtain

$$AZT + \sigma(ZT + AZ) = WQ. \tag{2.10}$$

The pth column of Eq. (2.10) can be written as

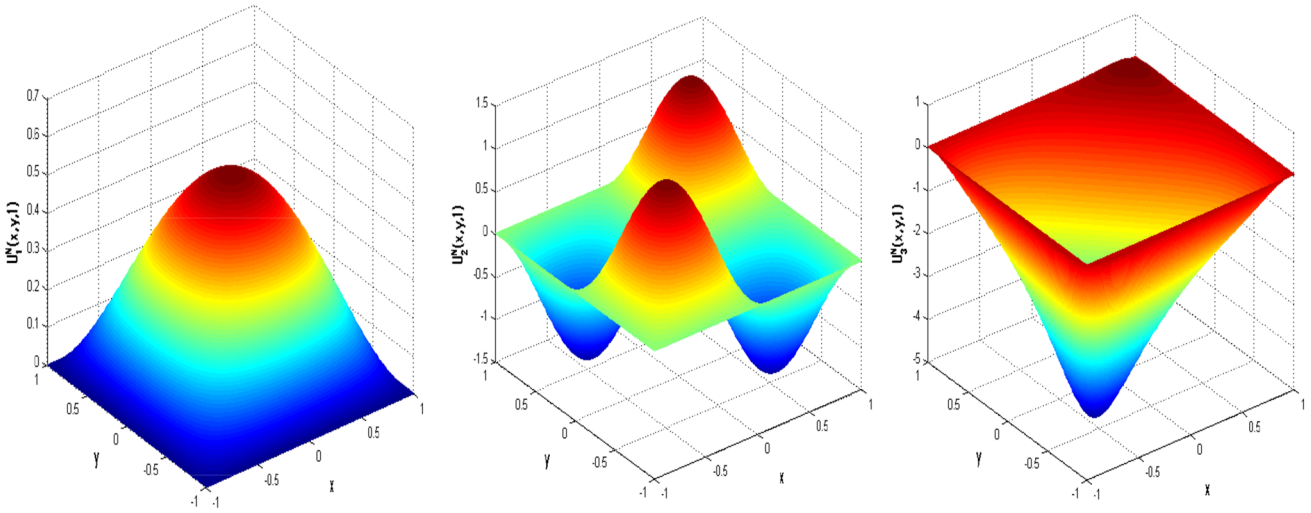

Fig. 1 Numerical solutions of (3.1) for $N = 30$, $\delta t = 10^{-5}$ at $t = 1$

Table 1 The L_∞, L_2 errors of (3.1) at $t = 1$ for $\delta t = 10^{-5}$

N	10	15	20	25
$\|e_1(1)\|_\infty$	4.844371 (−5)	1.341858 (−6)	1.340895 (−7)	1.340895 (−7)
$\|e_2(1)\|_\infty$	2.037579 (−4)	5.579877 (−7)	5.531989 (−7)	5.531989 (−7)
$\|e_3(1)\|_\infty$	5.511335 (−4)	4.338563 (−8)	4.331084 (−8)	4.331085 (−8)
$\|e_1(1)\|_2$	1.889930 (−5)	1.178489 (−7)	1.178485 (−7)	1.178485 (−7)
$\|e_2(1)\|_2$	1.443047 (−4)	5.614549 (−7)	5.613424 (−7)	5.613424 (−7)
$\|e_3(1)\|_2$	4.557040 (−4)	3.604120 (−8)	3.595470 (−8)	3.595470(−8)

$$\lambda_p AZ_p + \sigma(\lambda_p Z_p + AZ_p) = (WQ)_p, \tag{2.11}$$

where the index p indicates the pth column of the matrix and λ_p is the pth eigenvalue of A. Finally (2.9) will be equivalent to solving $N-1$ systems of linear algebraic equations in the form of (2.11), simultaneously. We must note that the matrix A in (2.9) is fixed and at each time step remains unchanged; hence it is enough to decompose A only once to compute matrices Q, and T for using at each time step.

Therefore, the proposed scheme is as simple as finite difference method but with higher order accuracy (exponential convergence) in spatial coordinates and second-order convergence in time that are confirmed by the numerical results. Also, unlike the most paper which are concerned with nonlinear system (1.1), there is no iteration in this method.

Table 2 The L_∞, L_2 errors of (3.1) at $t = 1$ for $N = 30$

δt	$\|e_1(1)\|_\infty$	Rate	$\|e_2(1)\|_\infty$	Rate	$\|e_3(1)\|_\infty$	rate
10^{-2}	9.841238 (−5)	–	5.263786 (−4)	–	6.600534 (−5)	–
10^{-3}	1.305413 (−5)	2.02	5.505243 (−5)	2.26	4.552919 (−6)	2.67
10^{-4}	1.337658 (−6)	2.28	5.529554 (−6)	2.30	4.351232 (−7)	2.35
10^{-5}	1.340895 (−7)	2.30	5.531989 (−7)	2.30	4.331085 (−8)	2.31
δt	$\|e_1(1)\|_2$	Rate	$\|e_2(1)\|_2$	Rate	$\|e_3(1)\|_2$	Rate
10^{-2}	8.409377 (−5)	–	5.288837 (−4)	–	5.202314 (−5)	–
10^{-3}	1.144884 (−5)	1.99	5.581146 (−5)	2.25	3.668486 (−6)	2.65
10^{-4}	1.175420 (−6)	2.28	5.610489 (−6)	2.30	3.601070 (−7)	2.32
10^{-5}	1.178485 (−7)	2.30	5.613424 (−7)	2.30	3.595470 (−8)	2.30

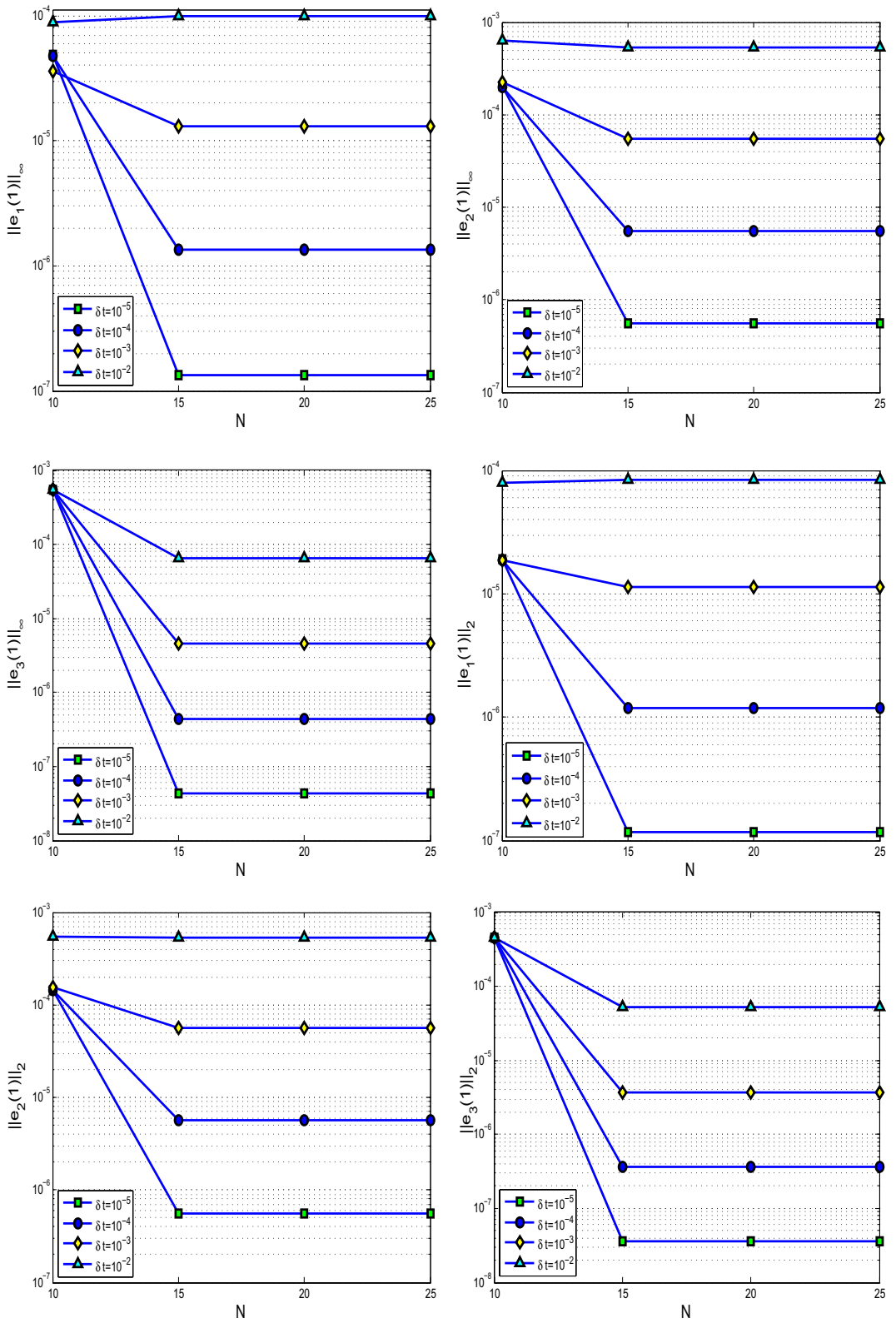

Fig. 2 Diagram of error $\|e(i)\|_\infty$, and $\|e(i)\|_2$, $(i = 1, 2, 3)$, versus N for (3.1)

3 Numerical experiments

In this section we implement Chebyshev–Legendre–Galerkin method for some examples of nonlinear system of advection–diffusion–reaction equations. Computations are performed for various advection vector $\mathbf{v}$, diffusion and kinetic coefficients, d_i, $b_{i,j,k}$ and $c_{i,j}$. The right-hand sides (f_i's) are obtained by applying the differential operator to the exact solutions. The numerical experiments confirm the second-order convergence in time and the spectral accuracy in space.

In each example, graphs of relative errors in two norms L_∞, and L_2 are plotted at $t = 1$ with different time steps and different values of the spectral discretization parameter N. Errors are defined at the uniform grid $x_{i,j} = (x_i, x_j)$ in $[-1, 1]^2$ and time t as follows:

$$\|e_i(t)\|_\infty = \frac{\|U_i^N(.,t) - u_i(.,t)\|_\infty}{\|u_i(.,t)\|_\infty},$$
$$\|e_i(t)\|_2 = \frac{\|U_i^N(.,t) - u_i(.,t)\|_2}{\|u_i(.,t)\|_2}, \quad i = 1, 2, \ldots, m$$

where

$$x_j = -1 + \frac{2j}{\ell}, \quad 0 \le j \le \ell, \quad \ell = 1000,$$

and $U_i^N(x,t)$, and $u_i(x,t)$ are the ith numerical and exact solutions, respectively.

Example 1 (advection–diffusion–reaction system) This example is devoted to solve system of advection–diffusion–reaction (1.1) for $m = 3$, velocity vector $\mathbf{v} = (1, 1)$, and $d_i = i$, $(i = 1, 2, 3)$. Indeed, we consider the following system:

$$\begin{cases} \frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x} + \frac{\partial u_1}{\partial y} - \Delta u_1 + R_1(u_1, u_2, u_3) + f_1(\mathbf{x}, t) = 0, \\ \frac{\partial u_2}{\partial t} + \frac{\partial u_2}{\partial x} + \frac{\partial u_2}{\partial y} - 2\Delta u_2 + R_2(u_1, u_2, u_3) + f_2(\mathbf{x}, t) = 0, \\ \frac{\partial u_3}{\partial t} + \frac{\partial u_3}{\partial x} + \frac{\partial u_3}{\partial y} - 3\Delta u_3 + R_3(u_1, u_2, u_3) + f_3(\mathbf{x}, t) = 0, \end{cases} \tag{3.1}$$

in which $R_1 = u_1u_2 + u_1u_3$, $R_2 = u_1 + u_2 + u_3$, and $R_3 = u_2u_3$ and the exact solutions are supposed to be

$$\begin{cases} u_1(x, y, t) = (1 - x^2)(1 - y^2)\cos(t), \\ u_2(x, y, t) = (1 + e^{-t})\sin(\pi x)\sin(\pi y), \\ u_3(x, y, t) = (1 - x^6)(1 - y^6)\sinh(x + y - t^2). \end{cases}$$

Also f_1, f_2 and f_3 are smooth source functions which are chosen corresponding to the exact solutions.

For different definitions of reaction terms R_i, such systems like (3.1) have many applications in various fields of science. In [48] the modeling of synchronous oscillation in

Table 3 The L_∞, L_2 errors of (3.2) at $t = 1$ for $\delta t = 10^{-5}$

N	$\|e_1(1)\|_\infty$	$\|e_1(1)\|_2$	$\|e_2(1)\|_\infty$	$\|e_2(1)\|_2$
10	7.011377 (−3)	4.812797 (−3)	8.206886 (−2)	6.227988 (−2)
15	2.061999 (−5)	8.810581 (−6)	1.700423 (−3)	1.323492 (−3)
20	6.432047 (−8)	2.621419 (−8)	3.402656 (−5)	2.760502 (−5)
25	1.587940 (−8)	1.060456 (−8)	6.400573 (−7)	5.090038 (−7)
30	1.560151 (−8)	1.057616 (−8)	9.315529 (−8)	9.152636 (−8)

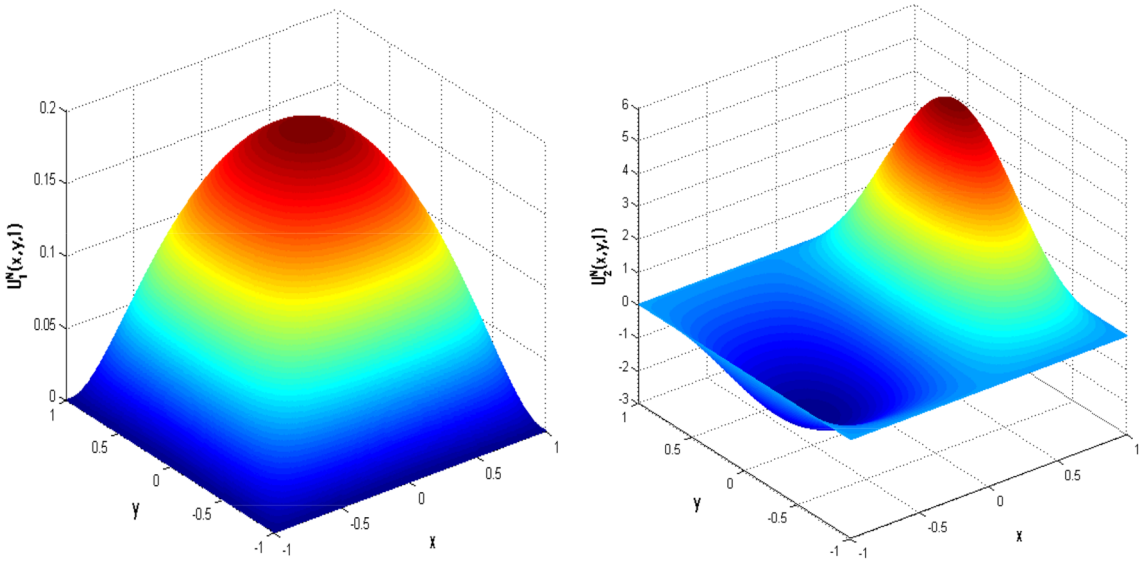

Fig. 3 Numerical solutions of (3.2) for $N = 30$, $\delta t = 10^{-5}$ at $t = 1$

Table 4 The L_∞, L_2 errors of (3.2) at $t = 1$ for $N = 30$

δt	$\|e_1(1)\|_\infty$	Rate	$\|e_1(1)\|_2$	Rate	$\|e_2(1)\|_\infty$	Rate	$\|e_2(1)\|_2$	rate
10^{-2}	3.437175 (−5)	–	2.268715 (−5)	–	5.283309 (−5)	–	5.449173 (−5)	–
10^{-3}	1.743721 (−6)	2.98	1.176901 (−6)	2.96	8.228104 (−6)	1.86	8.753732 (−6)	1.83
10^{-4}	1.576551 (−7)	2.40	1.068199 (−7)	2.40	8.547295 (−7)	2.26	9.085420(−7)	2.27
10^{-5}	1.560151 (−8)	2.31	1.057616 (−8)	2.31	9.315529 (−8)	2.22	9.152636 (−8)	2.30

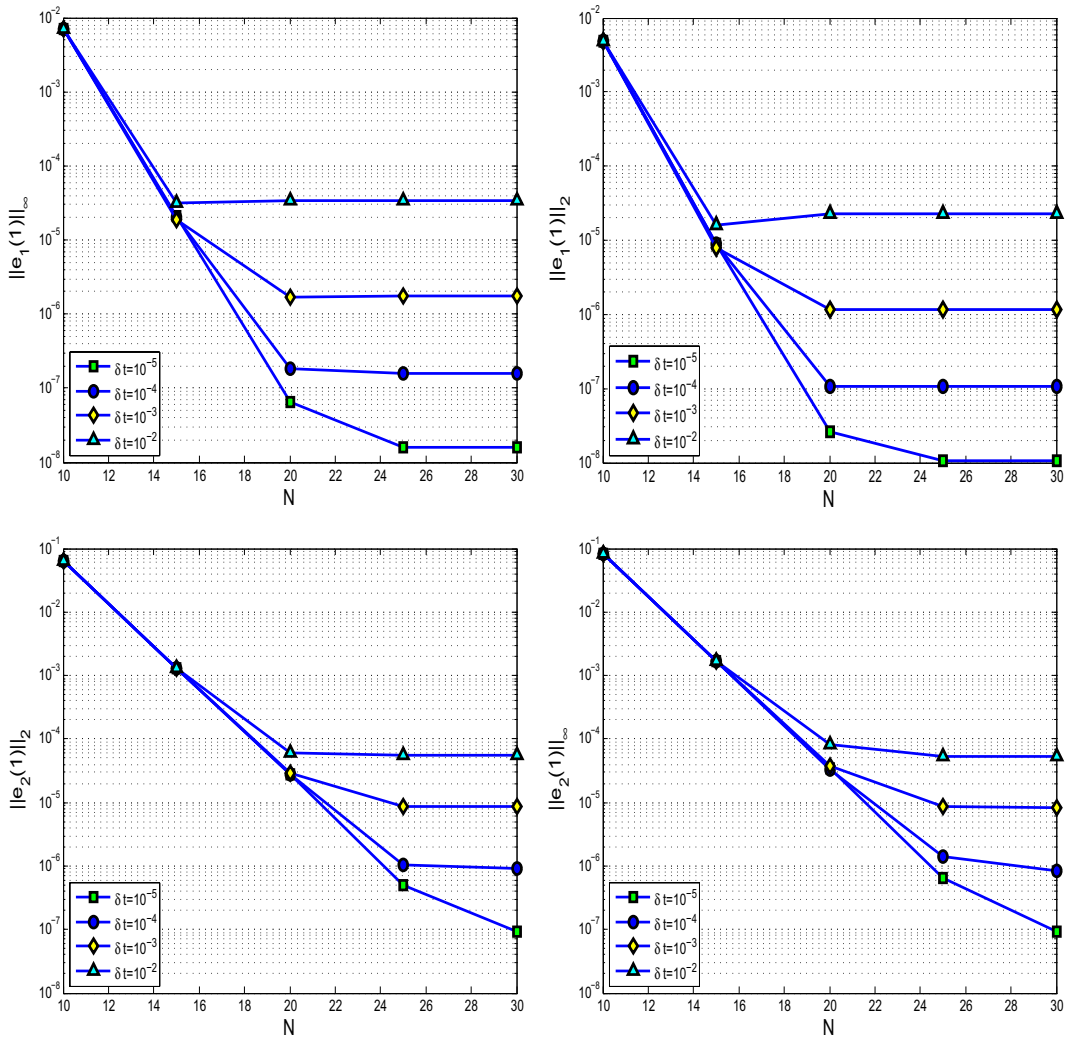

Fig. 4 Diagram of error $\|e(i)\|_\infty$, and $\|e(i)\|_2$, $(i = 1, 2)$, versus N for (3.2)

the plasmodium is stated by the above system and the quantities u_1, u_2, and u_3 are interpreted by concentrations of chemical substance in the free compartment of ectoplasm, non-free compartment of ectoplasm and endoplasmic compartment, respectively. Also in [6] a mathematical framework for the main qualitative features of plankton patchiness is presented in which u_1 is maximum phytoplankton content supported by a parcel of water in the absence of grazing, u_2 is distribution of phytoplankton and u_3 is mortality.

In Fig. 1 the numerical solutions of (3.1) for $N = 30$ and $\delta t = 10^{-5}$ are plotted at $t = 1$. Also The behavior of L_∞, and L_2 errors at $t = 1$ versus N are plotted in Fig. 2 for different time steps. Furthermore, the error values are shown in Tables 1, 2 confirming that the order of convergence (rate) in time is two and in space is spectral (exponential).

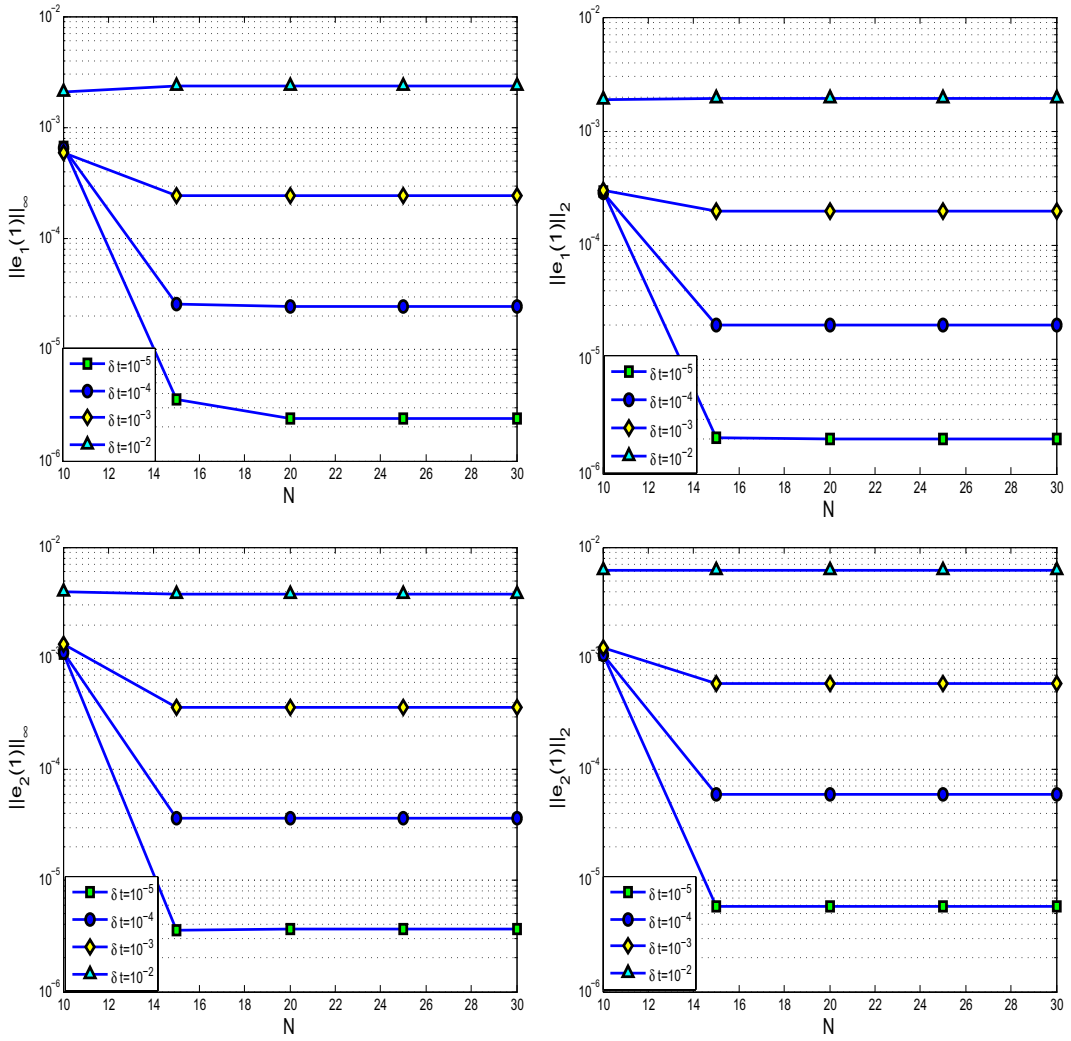

Fig. 5 Diagram of error $\|e(i)\|_\infty$, and $\|e(i)\|_2$, $(i = 1, 2)$, versus N for (3.3)

Table 5 The L_∞, L_2 errors of (3.3) at $t = 1$ for $\delta t = 10^{-5}$

N	$\|e_1(1)\|_\infty$	$\|e_1(1)\|_2$	$\|e_2(1)\|_\infty$	$\|e_2(1)\|_2$
10	6.750152 (−4)	2.948005 (−4)	1.107468 (−3)	1.068549v(−3)
15	3.608261 (−6)	2.085370 (−6)	3.535145 (−6)	5.912632 (−6)
20	2.424811 (−6)	2.019912 (−6)	3.638494 (−6)	5.907836 (−6)
25	2.429047 (−6)	2.019875 (−6)	3.634412 (−6)	5.907862 (−6)
30	2.429038 (−6)	2.019876 (−6)	3.634416 (−6)	5.907862 (−6)

In all tabular results, we use the abbreviation notation $x(n)$ instead of $x \times 10^n$.

Example 2 (predator–prey system) In this example, generalized predator–prey system with many applications in populations dynamics and ecology is considered as follows:

$$\begin{cases} \frac{\partial u_1}{\partial t} = D\Delta u_1 + ru_1(1 - \frac{u_1}{v_0}) - cu_2 g(u_1) - f_1(\mathbf{x}, t), \\ \frac{\partial u_2}{\partial t} = D\Delta u_2 - bu_2 + au_2 g(u_1) - f_2(\mathbf{x}, t), \end{cases} \tag{3.2}$$

Table 6 The L_∞, L_2 errors of (3.3) at $t = 1$ for $N = 30$

δt	$\|e_1(1)\|_\infty$	Rate	$\|e_1(1)\|_2$	Rate	$\|e_2(1)\|_\infty$	Rate	$\|e_2(1)\|_2$	Rate
10^{-2}	2.342734 (−3)	–	1.948764 (−3)	–	3.745957 (−3)	–	6.107081 (−3)	–
10^{-3}	2.420442 (−4)	2.27	2.012799 (−4)	2.27	3.645450 (−4)	2.33	5.927519 (−4)	2.33
10^{-4}	2.428250 (−5)	2.30	2.019228 (−5)	2.30	3.635420 (−5)	2.30	5.909649 (−5)	2.31
10^{-5}	2.429038 (−6)	2.30	2.019876 (−6)	2.30	3.634416 (−6)	2.30	5.907862 (−6)	2.30

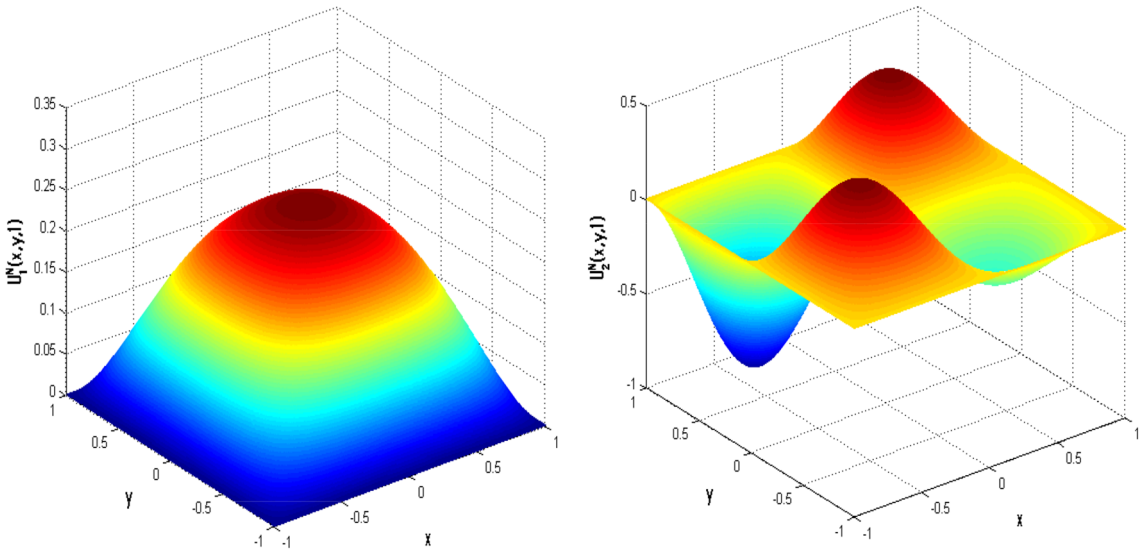

Fig. 6 Numerical solutions of (3.3) for $N = 30$, $\delta t = 10^{-5}$ at $t = 1$

where r, c, a, b, v_0 are positive parameters and D is the diffusivity. Also u_2, and u_1 are concentrations of predators and preys, respectively. As mentioned in [6], the function $g(\cdot)$ represents the prey consumption rate per predator as a function of the maximal consumption rate, c. Parameters a and r denote maximal per capita predator and prey birth rates; for predators, that is the birth rate when the prey density is very high, while for prey, it is the birth rate at very low prey density. The per capita predator death rate is denoted by b, and v_0 is the prey carrying capacity.

We suppose

$$\begin{cases} u_1(x,y,t) = \frac{(e^{x^2-1}-1)(e^{y^2-1}-1)}{1+t^2}, \\ u_2(x,y,t) = (e^{\sin(\pi x)}-1)(e^{\cos(\frac{\pi}{2}y)}-1)(1+t^4), \end{cases}$$

are the exact solutions of (3.2) with taking $g(u) = u + u^3$, and the parameters $D = 1.5$, $c = r = v_0 = 0.5$, $a = b = 1$. Also f_1 and f_2 are defined according to the exact solutions.

For $N = 30$ and $\delta t = 10^{-5}$ the numerical solutions at $t = 1$ are drawn in Fig. 3. In Tables 3 and 4, error values for different spectral discretization parameter, N, and different time steps are listed, respectively. As can be predicted, the rate of convergence in time is approximately two. The behavior of L_∞, and L_2 errors at $t = 1$ versus N are depicted in Fig. 4 for different time steps.

Example 3 (Fitzhugh–Nagumo system) As the last example, we investigate Fitzhugh–Nagumo system that originally has been introduced as a mathematical model for the neural activity [14, 49]. This model is presented as

$$\begin{cases} \frac{\partial u_1}{\partial t} - D\Delta u_1 - u_1(u_1-1)(a-u_1) + u_2 + f_1(\mathbf{x},t) = 0, \\ \frac{\partial u_2}{\partial t} - bu_1 + \gamma u_2 + f_2(\mathbf{x},t) = 0, \end{cases} \tag{3.3}$$

where u_1 is the activator, u_2 is the inhibitor, and a, b, γ are positive parameters.

We take the exact solutions as

$$\begin{cases} u_1(x,y,t) = \frac{(1-x^4)(1-y^4)}{(1+t^2+y^2)(1+t^2+x^2)}, \\ u_2(x,y,t) = \frac{\sin(\pi x)\sin(\pi y)}{e^{tx}+e^{-ty}}, \end{cases}$$

and define the source functions f_1 and f_2 accordingly.

The numerical solutions at $t = 1$ for $N = 30$ and $\delta t = 10^{-5}$ are plotted in Fig. 5. Error values for different spectral discretization parameter, N, are presented in Table 5 and also in Table 6 they are listed for different time steps. The behavior of L_∞, and L_2 errors at $t = 1$ versus N are drawn in Fig. 6 for different time steps.

The numerical results presented in all examples confirm the second-order convergence in time and the spectral accuracy in space, but as can be seen in each example, there exists a certain value of N such N_0 whose increasing spectral discretization parameter, N, greater than N_0, has no significant effect on decreasing error and rate of convergence will be almost equal to zero. This fact is due to the time discretization error, because according to the error bound of the proposed method which is approximately in form of $\kappa((\delta t)^2 + N^{-m+1})$, only increasing spectral discretization parameter, N, or decreasing time step δt solely, will not reduce the obtained error and for this purpose we must increase N, and decrease δt, simultaneously.

4 Conclusion

In this paper we used the Legendre–Galerkin method combined with the Chebyshev-collocation technique for spatial discretization and Leapfrog–Crank–Nicolson scheme

for time advancing to solve a two-dimensional nonlinear advection–diffusion–reaction system. As can be seen, the advantage of this method over the classic Galerkin methods is its efficiency in solving nonlinear equations by converting the equation to solve a linear system of algebraic equations or a matrix equation that can be solved by decomposition methods in some cases. Numerical examples demonstrate the efficiency and spectral accuracy of the proposed method. Also the results presented in each example indicate that only increasing spectral discretization parameter N or decreasing time step δt will not reduce the obtained error and, therefore, we must increase N and decrease δt, simultaneously.

Acknowledgements The author is grateful to the reviewers for carefully reading this paper and for their comments and suggestions which have improved the paper.

References

1. Alpert BK, Rokhlin V (1991) A fast algorithm for the evaluation of Legendre expansions. SIAM J Sci Stat Comput 12:158–179
2. Boyd JP (2000) Chebyshev and Fourier spectral methods, 2nd edn. Dover, New York
3. Boyd JP (1994) Time-marching on the slow manifold: the relationship between the nonlinear Galerkin method and implicit timestepping algorithms. Appl Math Lett 7:95–99
4. Caliari M, Vianello M, Bergamaschi L (2007) The LEM exponential integrator for advection–diffusion–reaction equations. J Comput Appl Math 210(1–2):56–63
5. Canuto C, Hussaini MY, Quarteroni A, Zang TA (1988) Spectral methods in fluid dynamics. Springer, New York
6. Cencini M, Lopez C, Vergni D (2003) Reaction–diffusion systems: front propagation and spatial structures. Lect Notes Phys 636:187–210
7. Dehghan M (2004) Numerical solution of the three-dimensional advection–diffusion equation. Appl Math Comput 150:5–19
8. Dehghan M, Sabouri M (2013) A Legendre spectral element method on a large spatial domain to solve the predator–prey system modeling interacting populations. Appl Math Model 37:1028–1038
9. Don WS, Gottlieb D (1994) The Chebyshev–Legendre method: implementing Legendre methods on Chebyshev points. SIAM J Numer Anal 31:1519–1534
10. Elbarbary E (2008) Efficient Chebyshev–Petrov–Galerkin method for solving second-order equations. J Sci Comput 34:113–126
11. El Alaoui L, Ern A (2006) Nonconforming finite element methods with subgrid viscosity applied to advection–diffusion–reaction equations. Numer Methods Partial Differ Equ 22(5):1106–1126
12. Fakhar-Izadi F, Dehghan M (2013) An efficient pseudo-spectral Legendre Galerkin method for solving a nonlinear partial integro-differential equation arising in population dynamics. Math Methods Appl Sci 36(12):1485–1511
13. Fazio R, Jannelli A (2009) Second order numerical operator splitting for 3D advection–diffusion–reaction models. In: Kreiss G et al (eds) Numerical mathematics and advanced applications. Springer, Berlin, pp 317–324
14. Fitzhugh R (1961) Impulses and physiological states in theoretical models of nerve membranes. J Biophys 1:445–466
15. Ganesh M, Mustapha K (2006) A fully discrete H^1-Galerkin method with quadrature for nonlinear advection–diffusion–reaction equations. Numer Algorithms 43:355–383
16. Gottlieb D, Orszag SA (1997) Numerical analysis of spectral methods: theory and applications. SIAM-CMBS, Philadelphia
17. Gottlieb D, Xiu D (2008) Galerkin method for wave equations with uncertain coefficients. Commun Comput Phys 3(2):505–518
18. Goubet O, Shen J (2007) On the dual Petrov–Galerkin formulation of the KdV equation. Adv Differ Equ 12:221–239
19. Greengard L, Rokhlin V (1987) A fast algorithm for particle simulations. J Comput Phys 73:325–348
20. Gu Y, Liao W, Zhu J (2003) An efficient high-order algorithm for solving systems of 3-D reaction–diffusion equations. J Comput Appl Math 155:1–17
21. Guo B-Y, Shen J (2000) Laguerre–Galerkin method for nonlinear partial differential equations on a semi-infinite interval. Numer Math 86:635–654
22. Guo B-Y (1998) Spectral methods and their applications. World Scientific, River Edge
23. Guo B-Y, Shen J, Wang L-L (2006) Optimal spectral-Galerkin methods using generalized Jacobi polynomials. J Sci Comput 27(1–3):305–322
24. Hidalgo A, Dumbser M (2011) ADER schemes for nonlinear systems of stiff advection–diffusion–reaction equations. J Sci Comput 48:173–189
25. Hoff D (1978) Stability and convergence of finite difference methods for systems of nonlinear reaction–diffusion. SIAM J Numer Anal 15:1161–1177
26. Houston P, Schwab C, Süli E (2002) Discontinuous hp-finite element methods for advection–diffusion–reaction problems. SIAM J Numer Anal 39:2133–2163
27. Houzeaux G, Eguzkitza B, Vázquez M (2009) A variational multiscale model for the advection–diffusion–reaction equation. Commun Numer Methods Eng 25:787–809
28. Hrinca I (2002) An optimal control problem for the Lotka–Volterra system with diffusion. Panamer Math J 12(3):23–46
29. Huang W (2001) Uniqueness of the bistable traveling wave for mutualist species. J Dyn Differ Equ 13(1):147–183
30. Hundsdorfer W, Verwer JG (2003) Numerical solution of time-dependent advection–diffusion–reaction equations, vol 33. Springer series in computational mathematics. Springer, Berlin
31. Khan LA, Liu Philip L-F (1995) An operator splitting algorithm for coupled one-dimensional advection–diffusion–reaction equations. Comput Methods Appl Mech Eng 127:181–201
32. Ladyzenskaja OA, Solonnikov VA, Uralceva NN (1968) Linear and quasi-linear equations of parabolic type, vol 23. Translations of mathematical monographs. American Mathematical Society, Providence
33. Lagzi I, Kármán D, Turányi T, Tomlin A, Haszpra L (2004) Simulation of the dispersion of nuclear contamination using an adaptive Eulerian grid model. J Environ Radioact 75:59–82
34. Lagzi I, Mészáros R, Horváth L, Tomlin A et al (2004) Modelling ozone fluxes over Hungary. Atmos Environ 38:6211–6222
35. Li H, Wu H, Ma H (2003) The Legendre Galerkin–Chebyshev collocation method for Burgers-like equations. IMA J Numer Anal 23:109–124
36. Liao W, Zhu J, Khaliq Abdul QM (2002) An efficient high-order algorithm for solving systems of reaction–diffusion equations. Numer Methods Partial Differ Equ 18:340–354
37. Liu B (2009) An error analysis of a finite element method for a system of nonlinear advection–diffusion–reaction equations. Appl Numer Math 59:1947–1959
38. Liu B, Allen MB, Kojouharov H, Chen B (1996) Finite-element solution of reaction-diffusion equations with advection. In: Aldama AA et al (eds) Computational methods in water resources, vol 1. Computational methods in subsurface flow and transport problems. Computational Mechanics Publications, Southampton, pp 3–12

39. Lutscher F, McCauley E, Lewis MA (2007) Spatial patterns and coexistence mechanisms in systems with unidirectional flow. Theor Popul Biol 71:267–277
40. Ma HP (1998) Chebyshev–Legendre spectral viscosity method for nonlinear conservation laws. SIAM J Numer Anal 35:893–908
41. Ma HP (1998) Chebyshev–Legendre super spectral viscosity method for nonlinear conservation laws. SIAM J Numer Anal 35:869–892
42. Ma HP, Sun WW (2000) A Legendre–Petrov–Galerkin and Chebyshev collocation method for third-order differential equations. SIAM J Numer Anal 38:1425–1438
43. Ma HP, Sun WW (2001) Optimal error estimates of the Legendre–Petrov–Galerkin method for the Korteweg–de Vries equation. SIAM J Numer Anal 39:1380–1394
44. Matthies HG, Keese A (2005) Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. Comput Methods Appl Math Eng 194:1295–1331
45. McKibbin R, Lim LL, Smith TA, Sweatman WL (2005) A model for dispersal of eruption ejecta. In: Proceedings world geothermal congress, April 24–29, Antalya, Turkey
46. Mickens RE (2000) Nonstandard finite difference schemes for reaction–diffusion equations having linear advection. Numer Methods Partial Differ Equ 16:361–364
47. Mohebbi A, Dehghan M (2010) High-order compact solution of the one-dimension heat and advection–diffusion equation. Appl Math Model 34:3071–3084
48. Nakagaki T, Yamada H, Ito M (1999) Reaction–diffusion–advection model for pattern formation of rhythmic contraction in a giant amoeboid cell of the *Physarum plasmodium*. J Theor Biol 197:497–506
49. Nagumo J, Arimoto S, Yoshizawa S (1960) An active pulse transmission line simulating 1214-nerve axons. Proc IRL 50:2061
50. Naser G, Karney BW (2007) A 2-D transient multicomponent simulation model: application to pipe wall corrosion. J Hydro Environ Res 1:56–69
51. Nicolis G, Prigogine I (1977) Self-organization in nonequilibrium systems. Wiley, New York
52. Pao CV (1990) Numerical methods for coupled systems of nonlinear parabolic boundary value problems. J Math Anal Appl 15:581–608
53. Pao CV (1985) Monotone iterative methods for finite difference system of reaction diffusion equations. Numer Math 46:571–586
54. Pao CV (1995) Finite difference reaction–diffusion solutions with nonlinear boundary conditions. Numer Methods Partial Differ Equ 11:355–374
55. Pao CV (1999) Numerical analysis of coupled systems of nonlinear parabolic equations. SIAM J Numer Anal 36:393–416
56. Pao CV (2002) Finite difference reaction diffusion equations with coupled boundary conditions and time delays. J Math Anal Appl 272:407–434
57. Pan Z, Wang Y (1991) Numerical method for the system of reaction–diffusion equations with a small parameter. Appl Math Mech 12:813–819
58. Perthame B (2007) Transport equations in biology. Frontiers in mathematics. Birkhäuser, Basel
59. Perthame B, Ǵenieys S (2007) Concentration in the nonlocal Fisher equation: the Hamilton–Jacobi limit. Math Model Nat Phenom 4:135–151
60. Polyanin AD, Zaitsev VF (2004) Handbook of nonlinear partial differential equations. Chapman & Hall/CRC, Boca Raton
61. Pudykiewicz JA (2006) Numerical solution of the reaction–advection–diffusion equation on the sphere. J Comput Phys 213:358–390
62. Qiu Y, Sloan DM (1998) Numerical solution of Fisher's equation using a moving mesh method. J Comput Phys 146:726–746
63. Ritchie H (1985) Application of a semi-Lagrangian integration scheme to the moisture equation in a regional forecast model. Mon Weather Rev 113:424–435
64. Roberts LS, Janovy J, Schmidt GD (2008) Foundations of parasitology. McGraw Hill, Boston
65. Zhao TG, Liang YT, Ma HP (2011) Chebyshev–Legendre pseudo-spectral method for the modified Kawahara equation. Adv Mater Res 143–144:191–195
66. Shen J (1996) Efficient Chebyshev–Legendre Galerkin methods for elliptic problems. In: Ilin AV, Scott R (eds) Proceedings of ICOSA-HOM'95, Houston J Math,pp 233–240
67. Shen J (1994) Efficient spectral-Galerkin method I. Direct solvers for second- and fourth-order equations by using Legendre polynomials. SIAM J Sci Comput 15:1489–1505
68. Shen J (1995) Efficient spectral-Galerkin method II. Direct solvers for second- and fourth-order equations by using Chebyshev polynomials. SIAM J Sci Comput 16:74–87
69. Shen J, Wang L-L (2006) Laguerre and composite Legendre–Laguerre dual-Petrov–Galerkin methods for third-order equations. Discret Contin Dyn Syst Ser B 6(6):1381–1402
70. Shen J (2003) A new dual-Petrov–Galerkin method for third and higher odd-order differential equations: application to the KDV equation. SIAM J Numer Anal 41:1595–1619
71. Shen J (1997) Efficient spectral-Galerkin methods III. Polar and cylindrical geometries. SIAM J Sci Comput 18:1583–1604
72. Shen J (1999) Efficient spectral-Galerkin methods IV. Spherical geometries. SIAM J Sci Comput 20:1438–1455
73. Shen TT, Xing KZ, Ma HP (2011) A Legendre Petrov–Galerkin method for fourth-order differential equations. Comput Math Appl 61:8–16
74. Spee EJ, Verwer JG, de Zeeuw PM, Blom JG, Hundsdorfer W (1998) A numerical study for global atmospheric transport-chemistry problems. Math Comput Simul 48:177–204
75. Sun P (1996) A pseudo-non-time-splitting method in air quality modeling. J Comput Phys 127:152–157
76. Toro M, van Rijn L, Meijer K (1989) Three-dimensional modelling of sand and mud transport in current and waves. Technical Report No. H461, Delft Hydraulics, Delft, The Netherlands
77. Trefethen LN (2000) Spectral methods in MATLAB. SIAM, Philadelphia
78. Veldhuizen SV, Vuik C, Kleijn CR (2007) Inexact Newton methods for solving stiff systems of advection–diffusion–reaction equations. In: Giraud L et al (eds) Proceedings of the international conference on preconditioning techniques for large sparse matrix problems in scientific and industrial applications, France, Toulouse
79. Verwer JG, Sommeijer BP, Hundsdorfer W (2004) RKC time-stepping for advection–diffusion–reaction problems. J Comput Phys 201:61–79
80. Wang Y-M, Guo B-Y (2008) A monotone compact implicit scheme for nonlinear reaction–diffusion equations. J Comput Math 26:123–148
81. Wang Y-M, Pao CV (2006) Time-delayed finite difference reaction–diffusion systems with nonquasimonotone functions. Numer Math 103:485–513
82. Wang Y-M, Zhanga H-B (2009) Higher-order compact finite difference method for systems of reaction–diffusion equations. J Comput Appl Math 233:502–518
83. Wang Y-M (2001) Asymptotic behavior of the numerical solutions for a system of nonlinear integrodifferential reaction–diffusion equations. Appl Numer Math 39:205–223
84. Williamson DL, Rash PJ (1989) Two-dimensional semi-Lagrangian transport with shape-preserving interpolation. Mon Weather Rev 117:102–129
85. Williamson DL, Drake JB, Hack JJ, Jakob R, Shwartzrauber PN (1992) A standard test set for numerical approximations to the

shallow water equations in spherical geometry. J Comput Phys 102:211–224
86. Yang J, Vatsala AS (2005) Numerical investigation of generalized quasilinearization method for reaction diffusion systems. Comput Math Appl 50:587–598
87. Yuan JM, Shen J, Wu J (2008) A dual-Petrov–Galerkin method for the Kawahara-type equations. J Sci Comput 34:48–63
88. Zhao S, Ovadia J, Liu X, Zhang Y-T, Nie Q (2011) Operator splitting implicit integration factor methods for stiff reaction–diffusion–advection systems. J Comput Phys 230:5996–6009