Available online at www.sciencedirect.com

ScienceDirect

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

Journal of Computational and Applied Mathematics 214 (2008) 549 – 571

www.elsevier.com/locate/cam

Two-grid discretization schemes for nonlinear Schrödinger equations

C.-S. Chien[a,*,1], H.-T. Huang[b], B.-W. Jeng[c], Z.-C. Li[d]

[a] *Department of Applied Mathematics, National Chung-Hsing University, Taichung 402, Taiwan*
[b] *Department of Applied Mathematics, I-Show University, Kaohsiung County 840, Taiwan*
[c] *Department of Applied Mathematics, National Chiao-Tung University, Hsinchu 300, Taiwan*
[d] *Department of Applied Mathematics, National Sun Yat-sen University, Kaohsiung 804, Taiwan*

Received 8 August 2006; received in revised form 16 March 2007

Abstract

We study efficient two-grid discretization schemes with two-loop continuation algorithms for computing wave functions of two-coupled nonlinear Schrödinger equations defined on the unit square and the unit disk. Both linear and quadratic approximations of the operator equations are exploited to derive the schemes. The centered difference approximations, the six-node triangular elements and the Adini elements are used to discretize the PDEs defined on the unit square. The proposed schemes also can compute stationary solutions of parameter-dependent reaction–diffusion systems. Our numerical results show that it is unnecessary to perform quadratic approximations.

MSC: 65N99; 35Q55

Keywords: Schrödinger equations; Two-grid discretization schemes; Continuation; Adini’s elements

1. Introduction

In the past decade, Bose–Einstein condensation (BEC) of alkali atoms and hydrogen has been produced and studied in the laboratory [14], and has intrigued researchers in physics and mathematics. The macroscopic wave function of the BEC is governed by the nonlinear Schrödinger equation (NLS)

$$\begin{aligned}
&\mathbf{i}\Phi_t = -\Delta\Phi + V(x)\Phi + \mu|\Phi|^2\Phi, \quad x \in \mathbf{R}^2, \quad t > 0,\\
&\Phi(x,t) \in \mathbf{C},\\
&\Phi(x,t) \to 0 \quad \text{as } |x| \to +\infty, \quad t > 0. \qquad (1.1)
\end{aligned}$$

[*] Corresponding author. Tel.: +886 4 22860133/619; fax: +886 4 22873028.
E-mail address: cschien@amath.nchu.edu.tw (C.-S. Chien).

[1] Supported by the National Science Council of R.O.C. (TAIWAN) through Project NSC 94-2115-M-005-001.

0377-0427/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2007.03.017

Here $\Phi = \Phi(x,t)$ is the wave function of the BEC, $V(x) = \frac{1}{2}(\gamma_1^2 x_1^2 + \gamma_2^2 x_2^2)$ the trapping potential with γ_1 and γ_2 as the trap frequencies in x_1- and x_2-axis, respectively. The trapping potential is isotropic if $\gamma_1 = \gamma_2$, otherwise it is called nonisotropic. The coefficient μ can be positive or negative depending on the interaction is repulsive or attractive. Eq. (1.1) has been studied extensively for many years because of their importance in physical and mathematical problems. Experimental reports concerning the BEC can be found, e.g., in [1,14]. Mathematical and numerical study of (1.1) can be found in [3–5,18]. To compute wave functions of (1.1), in general one has to discretize or integrate the partial derivative of Φ with respect to t. For instance, Bao et al. [5] used time-splitting spectral discretizations to compute wave functions of (1.1).

In this paper, we will study two-grid discretization schemes for computing stationary state solutions of the two-coupled nonlinear Schrödinger equations (TCNLS)

$$\begin{aligned} &\mathbf{i}\frac{\partial}{\partial t}\Phi_j = -\Delta\Phi_j + V_j(x)\Phi_j + \mu_j|\Phi_j|^2\Phi_j + \sum_{i\neq j}\beta_{ij}|\Phi_i|^2\Phi_j \quad \text{for } x\in\mathbf{R}^2, \quad t>0,\\ &\Phi_j = \Phi_j(x,t)\in\mathbf{C}, \quad j=1,2,\\ &\Phi_j(x,t)\to 0 \quad \text{as } |x|\to+\infty, \quad t>0, \end{aligned} \tag{1.2}$$

where Φ_j and $V_j(x)$ have the same meaning as those of Φ and $V(x)$ in (1.1). The coefficients μ_j can be positive or negative. The coupling constant β_{ij} is the interaction between Φ_1 and Φ_2. We say the interaction is repulsive if $\beta_{ij} > 0$, and attractive if $\beta_{ij} < 0$. For simplicity we let $\beta_{12} = \beta_{21} = \beta$. Eq. (1.2) describes uniting two grown condensates to form one large single condensate. The first experiment involving the uniting of multicomponent BEC was reported in [20]. Bao [2] studied ground states and dynamics of multicomponent BEC using a continuous normalized gradient flow (CNGF), the backward Euler finite difference (BEFD), and a time-splitting sine-spectral method. Ghosh [12] studied the dynamics of the TCNLS. Recently, Chang et al. [7] proposed a time-independent approach for computing wave functions of (1.2). To be precise, let

$$\Phi_j(x,t) = \mathrm{e}^{-\mathbf{i}\lambda_j t}u_j(x), \tag{1.3}$$

where λ_j are the chemical potentials of the system. Then (1.2) can be transformed into

$$\begin{aligned} &-\Delta u_1 - \lambda_1 u_1 + V_1(x)u_1 + \mu_1 u_1^3 + \beta u_1 u_2^2 = 0 \quad \text{in } \Omega,\\ &-\Delta u_2 - \lambda_2 u_2 + V_2(x)u_2 + \mu_2 u_2^3 + \beta u_2 u_1^2 = 0 \quad \text{in } \Omega,\\ &u_1 = u_2 = 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{1.4}$$

where $\Omega \subset \mathbf{R}^2$ is a smooth domain with piecewise smooth boundary $\partial\Omega$. Note that in (1.1) and (1.2) the solutions decay to zero in the far-field. Thus, in practical computation, the domain Ω should be chosen as large as possible if we impose Dirichlet boundary conditions on it. The solutions Φ_j also represent the jth component of the beam in Kerr-like photorefractive media.

Eq. (1.4) is a nonlinear system of equations involving multiparameters. The solution manifolds of (1.4) can be traced numerically using continuation methods, where one of the chemical potentials, say λ_1, is used as the continuation parameter; see e.g., [6] and the further references cited therein. If the solution manifolds of (1.4) is numerically traced, then the wave functions of (1.2) can be easily obtained using (1.3). In [10] Chien and Jeng proposed two-grid discretization schemes for tracing solution manifolds of semilinear elliptic eigenvalue problems, where both centered difference approximations and the six-node triangular elements were used to discretize the PDEs. In this paper, we will derive a two-grid discretization scheme so that the wave functions of (1.2) can be computed efficiently. To derive two-grid discretization schemes for nonlinear eigenvalue problems, one of the main concerns is that which discretization method should be used so that singular points of the PDEs such as folds and bifurcation points can be accurately approximated. Besides the discretization schemes exploited in [10], the high order Adini elements will also be used to discretize the PDEs. Recently, Huang et al. [15,16] studied global superconvergence and error estimates of Adini's elements for the Poisson type equations. The advantage of implementing Adini's elements is that less elements are required to obtain approximate solutions with high accuracy especially when the domain is rectangular.

The two-grid discretization schemes we propose here also can be used to compute stationary solutions of parameter-dependent reaction–diffusion systems of the following form [21]:

$$\frac{\partial u}{\partial t} = d_1 \Delta u + f(u, v, \lambda) \quad \text{in } \Omega,$$

$$\frac{\partial v}{\partial t} = d_2 \Delta v + g(u, v, \lambda) \quad \text{in } \Omega \tag{1.5}$$

for all $t \geqslant 0$, subject to suitable boundary conditions. Here u and v represent the state variables, $\lambda \in \mathbf{R}^k$ is the parameter vector, and the domain Ω is defined as in (1.4). Eq. (1.5) is used as a mathematical model for various types of problems in sciences and engineering, e.g., the Brusselator [22] in chemistry, the Gierer–Meinhardt system [13], the Lotka–Volterra system [19] in mathematical biology, and so on. We wish to mention here that during the past years some two-grid mixed finite element discretization schemes have been developed for reaction–diffusion equations [8,24,25].

This paper is organized as follows. In Section 2 we derive linear and quadratic approximations for (1.5) in the rectangular coordinates. The two-grid centered difference discretization algorithms are described in Section 3. The two-grid finite element counterparts can be described in a similar way and is omitted here. We show in Section 4 how the two-grid centered difference discretization schemes can be adapted to the polar coordinates. In Section 5 we give centered difference approximations for (1.4) and the Brusselator. We also briefly discuss the Adini's elements therein. Sample numerical results are reported in Section 6. Finally, some concluding remarks are given in Section 7.

2. Linear and quadratic approximations

For convenience we consider (1.5) for all $t \geqslant 0$, subject to the homogenous Dirichlet boundary conditions

$$u(x, y, t) = u_0, \quad v(x, y, t) = v_0, \quad (x, y) \in \partial\Omega. \tag{2.1}$$

Here the unknowns u, v are state variables which represent concentrations of some intermediate chemicals in the reaction, d_1 and d_2 are diffusion rates, while $\lambda \in \mathbf{R}$ is one of the control parameters in the system, e.g., initial or final products, catalysts, temperature, etc., and (u_0, v_0) is a uniform steady state solution, i.e., u_0, v_0 are independent of the variables t, x, y which satisfy

$$f(u_0, v_0, \lambda) = g(u_0, v_0, \lambda) = 0.$$

We assume that the functions f and g are at least twice continuously differentiable.

We rewrite the stationary state of (1.5) as

$$\Phi(u, v, \lambda) = \begin{bmatrix} \phi_1(u, v, \lambda) \\ \phi_2(u, v, \lambda) \end{bmatrix} = \begin{bmatrix} d_1 \Delta u + f(u, v, \lambda) \\ d_2 \Delta v + g(u, v, \lambda) \end{bmatrix} = 0 \quad \text{in } \Omega, \tag{2.2}$$

where $\Phi : B_1 \times B_1 \times \mathbf{R}^k \to B_2$ is a smooth mapping with $u_1, u_2 \in B_1$, $\lambda \in \mathbf{R}^k$ with B_1 and B_2 two Banach spaces. Let $\widetilde{h}, h \in (0, 1)$ be any two fixed positive numbers. Since we will use the solution manifolds of (1.4) together with (1.3) to compute wave functions of (1.2), we discretize (2.2) by using centered differences or finite elements on the coarse grid with uniform meshsize $\widetilde{h}$. Let $(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})$ be an approximate stationary state solution of (1.5) on the coarse grid, and $(u, v, \lambda_{\tilde{h}})$ the counterpart on the fine grid with uniform meshsize h. The linear approximations of the mappings $\phi_1(u, v, \lambda_{\tilde{h}})$ and $\phi_2(u, v, \lambda_{\tilde{h}})$ at $(u_{\tilde{h}}, v_{\tilde{h}})$are given by

$$\phi_1(u, v, \lambda_{\tilde{h}}) \approx \phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) + D_u\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}}) + D_v\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(v - v_{\tilde{h}}),$$

$$\phi_2(u, v, \lambda_{\tilde{h}}) \approx \phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) + D_u\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}}) + D_v\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(v - v_{\tilde{h}}). \tag{2.3}$$

Setting $e_1 = u - u_{\tilde{h}}$ and $e_2 = v - v_{\tilde{h}}$. Eq. (2.3) becomes

$$D_u\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 + D_v\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 \approx -\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}),$$

$$D_u\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 + D_v\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 \approx -\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}). \tag{2.4}$$

Note that

$$D_u\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 = d_1\Delta e_1 + \frac{\partial f}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1,$$

$$D_v\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 = \frac{\partial f}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2,$$

$$D_u\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 = \frac{\partial g}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1,$$

$$D_v\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 = d_2\Delta e_2 + \frac{\partial g}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2.$$

Thus (2.4) can be expressed as

$$d_1\Delta e_1 + \frac{\partial f}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 + \frac{\partial f}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 \approx -d_1\Delta u_{\tilde{h}} - f(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}),$$

$$d_2\Delta e_2 + \frac{\partial g}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 + \frac{\partial g}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 \approx -d_2\Delta v_{\tilde{h}} - g(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}).$$

The state variables of the approximate solution $(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})$ can be corrected on the fine grid by solving the following equations:

$$\begin{aligned} &d_1\Delta e_1 + \frac{\partial f}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 + \frac{\partial f}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 = -d_1\Delta u_{\tilde{h}} - f(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) \quad \text{in } \Omega = [0, \ell] \times [0, 1] \\ &d_2\Delta e_2 + \frac{\partial g}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_1 + \frac{\partial g}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})e_2 = -d_2\Delta v_{\tilde{h}} - g(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) \quad \text{in } \Omega = [0, \ell] \times [0, 1] \end{aligned} \tag{2.5}$$

with Dirichlet boundary conditions

$$e_1(x, y) = 0, \quad e_2(x, y) = 0, \quad (x, y) \in \partial\Omega.$$

In other words, the correction (e_1, e_2) on the fine grid is obtained by solving the following system of PDEs with Dirichlet boundary conditions:

$$\begin{bmatrix} d_1\varDelta + \frac{\partial f}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) & \frac{\partial f}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) \\ \frac{\partial g}{\partial u}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) & d_2\varDelta + \frac{\partial g}{\partial v}(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} -d_1\Delta u_{\tilde{h}} - f(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) \\ -d_2\Delta v_{\tilde{h}} - g(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) \end{bmatrix}. \tag{2.6}$$

Since the functions f and g are twice continuously differentiable, the quadratic approximations of $\phi_1(u, v, \lambda_{\tilde{h}})$ and $\phi_2(u, v, \lambda_{\tilde{h}})$ at $(u_{\tilde{h}}, v_{\tilde{h}})$ for the further corrections on the coarse grid are

$$\begin{aligned} \phi_1(u, v, \lambda_{\tilde{h}}) \approx{}& \phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) + D_u\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}}) + D_v\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(v - v_{\tilde{h}}) \\ &+ \tfrac{1}{2}[D_{uu}\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}})^2 + 2D_{uv}\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}})(v - v_{\tilde{h}}) \\ &+ D_{vv}\phi_1(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(v - v_{\tilde{h}})^2], \\ \phi_2(u, v, \lambda_{\tilde{h}}) \approx{}& \phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}}) + D_u\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}}) + D_v\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(v - v_{\tilde{h}}) \\ &+ \tfrac{1}{2}[D_{uu}\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}})^2 + 2D_{uv}\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(u - u_{\tilde{h}})(v - v_{\tilde{h}}) \\ &+ D_{vv}\phi_2(u_{\tilde{h}}, v_{\tilde{h}}, \lambda_{\tilde{h}})(v - v_{\tilde{h}})^2]. \end{aligned} \tag{2.7}$$

Then

$$\begin{aligned}
&D_u\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1) + D_v\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)\\
&\quad\approx -\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) - \tfrac{1}{2}[D_{uu}\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2 + 2D_{uv}\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2)\\
&\qquad + D_{vv}\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2],\\
&D_u\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1) + D_v\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)\\
&\quad\approx -\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) - \tfrac{1}{2}[D_{uu}\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2 + 2D_{uv}\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2)\\
&\qquad + D_{vv}\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2],
\end{aligned} \tag{2.8}$$

where $\widetilde{e}_1 = u - u_{\tilde h} - e_1$ and $\widetilde{e}_2 = v - v_{\tilde h} - e_2$. Note that

$$D_{uu}\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2 = \frac{\partial^2 f}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2,$$

$$D_{uv}\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2) = \frac{\partial^2 f}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2),$$

$$D_{vv}\phi_1(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2 = \frac{\partial^2 f}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2,$$

$$D_{uu}\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2 = \frac{\partial^2 g}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2,$$

$$D_{uv}\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2) = \frac{\partial^2 g}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2),$$

$$D_{vv}\phi_2(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2 = \frac{\partial^2 g}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2.$$

Thus (2.8) can be expressed as

$$\begin{aligned}
&d_1\Delta(\widetilde{e}_1+e_1) + \frac{\partial f}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1) + \frac{\partial f}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)\\
&\quad\approx -d_1\Delta u_{\tilde h} - f(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) - \frac{1}{2}\left[\frac{\partial^2 f}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2 + 2\frac{\partial^2 f}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2)\right.\\
&\qquad\left. + \frac{\partial^2 f}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2\right],\\
&d_2\Delta(\widetilde{e}_2+e_2) + \frac{\partial g}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1) + \frac{\partial g}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)\\
&\quad\approx -d_2\Delta v_{\tilde h} - g(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) - \frac{1}{2}\left[\frac{\partial^2 g}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)^2 + 2\frac{\partial^2 g}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1+e_1)(\widetilde{e}_2+e_2)\right.\\
&\qquad\left. + \frac{\partial^2 g}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2+e_2)^2\right].
\end{aligned} \tag{2.9}$$

Since (e_1, e_2) is the solution of (2.5), the system (2.9) becomes

$$
\begin{aligned}
d_1\Delta\widetilde{e}_1 &+ \frac{\partial f}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_1 + \frac{\partial f}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_2 \\
&\approx -\frac{1}{2}\left[\frac{\partial^2 f}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1 + e_1)^2 + \frac{\partial^2 f}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1 + e_1)(\widetilde{e}_2 + e_2)\right. \\
&\qquad \left. + \frac{\partial^2 f}{\partial u\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2 + e_2)(\widetilde{e}_1 + e_1) + \frac{\partial^2 f}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2 + e_2)^2\right] \\
&\approx -\frac{1}{2}\left[\frac{\partial^2 f}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1^2 + 2\frac{\partial^2 f}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1 e_2 + \frac{\partial^2 f}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_2^2\right], \\
d_2\Delta\widetilde{e}_2 &+ \frac{\partial g}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_1 + \frac{\partial g}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_2 \\
&\approx -\frac{1}{2}\left[\frac{\partial^2 g}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1 + e_1)^2 + \frac{\partial^2 g}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_1 + e_1)(\widetilde{e}_2 + e_2)\right. \\
&\qquad \left. + \frac{\partial^2 g}{\partial u\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2 + e_2)(\widetilde{e}_1 + e_1) + \frac{\partial^2 g}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})(\widetilde{e}_2 + e_2)^2\right] \\
&\approx -\frac{1}{2}\left[\frac{\partial^2 g}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1^2 + 2\frac{\partial^2 g}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1 e_2 + \frac{\partial^2 g}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_2^2\right].
\end{aligned}
$$

Therefore the correction $(\widetilde{e}_1, \widetilde{e}_2)$ is obtained by solving

$$
\begin{aligned}
d_1\Delta\widetilde{e}_1 &+ \frac{\partial f}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_1 + \frac{\partial f}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_2 \\
&= -\frac{1}{2}\left[\frac{\partial^2 f}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1^2 + 2\frac{\partial^2 f}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1 e_2 + \frac{\partial^2 f}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_2^2\right], \\
d_2\Delta\widetilde{e}_2 &+ \frac{\partial g}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_1 + \frac{\partial g}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})\widetilde{e}_2 \\
&= -\frac{1}{2}\left[\frac{\partial^2 g}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1^2 + 2\frac{\partial^2 g}{\partial v\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1 e_2 + \frac{\partial^2 g}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_2^2\right],
\end{aligned}
\tag{2.10}
$$

in $\Omega = [0, \ell] \times [0, 1]$ with Dirichlet boundary conditions

$$\widetilde{e}_1(x, y) = 0, \quad \widetilde{e}_2(x, y) = 0, \quad (x, y) \in \partial\Omega.$$

From the viewpoint of numerical computations, we rewrite (2.10) as

$$
\begin{aligned}
&\begin{bmatrix} d_1\varDelta + \frac{\partial f}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) & \frac{\partial f}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) \\ \frac{\partial g}{\partial u}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) & d_2\varDelta + \frac{\partial g}{\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h}) \end{bmatrix} \begin{bmatrix} \widetilde{e}_1 \\ \widetilde{e}_2 \end{bmatrix} \\
&\quad = -\frac{1}{2}\begin{bmatrix} \frac{\partial^2 f}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1^2 + 2\frac{\partial^2 f}{\partial u\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1 e_2 + \frac{\partial^2 f}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_2^2 \\ \frac{\partial^2 g}{\partial u^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1^2 + 2\frac{\partial^2 g}{\partial u\partial v}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_1 e_2 + \frac{\partial^2 g}{\partial v^2}(u_{\tilde h}, v_{\tilde h}, \lambda_{\tilde h})e_2^2 \end{bmatrix}.
\end{aligned}
\tag{2.11}
$$

Note that the coefficient matrix in (2.11) are the same as that in (2.6). In (2.11) we need to compute four first-order and six second-order partial derivatives on the coarse grid. The computations are inexpensive because $h = \mathrm{O}(\widetilde{h}^2)$.

3. Two-grid centered difference discretization algorithms

We discretize (2.2), (2.5), and (2.10) by the centered difference approximations with uniform mesh sizes $\widetilde{h}$ and h defined as in Section 2. Then we adapt Algorithm 4.2 in [10] to trace solution curves of (2.2). Let $(U_{\tilde{h}}^{(i-1)}, V_{\tilde{h}}^{(i-1)}, \lambda_{\tilde{h}}^{(i-1)})$ be a point which has been accepted as an approximating point for the solution curve $\widetilde{c}$ on the coarse grid. The next approximating point $(U_{\tilde{h}}^{(i)}, V_{\tilde{h}}^{(i)}, \lambda_{\tilde{h}}^{(i)})$ for $\widetilde{c}$ can be obtained by solving (2.2) with the predictor–corrector continuation method, which is called the outer-loop continuation. In order to obtain the corresponding approximating point $(U_h^{(i)}, V_h^{(i)}, \lambda_h^{(i)})$ for the solution curve c on the fine grid, we use $(U_{\tilde{h}}^{(i)}, V_{\tilde{h}}^{(i)}, \lambda_{\tilde{h}}^{(i)})$ as a predicted point for the inner-loop continuation. In practice, we have to interpolate $U_{\tilde{h}}^{(i)}$ and$V_{\tilde{h}}^{(i)}$ on the fine space. In the corrector step, we consider the linear and quadratic approximations of (2.2) to obtain the corrections for the state variables $U_{\tilde{h}}^{(i)}$ and $V_{\tilde{h}}^{(i)}$, and thus obtain the state variables $U_h^{(i)}$ and $V_h^{(i)}$ on the fine grid. More precisely, we solve (2.5) on the fine grid to obtain a correction $((E_1)_h, (E_2)_h)$ for the state variable $(U_{\tilde{h}}^{(i)}, V_{\tilde{h}}^{(i)})$. The further coarse grid correction $((\widetilde{E}_1)_{\tilde{h}}, (\widetilde{E}_2)_{\tilde{h}})$ is obtained by solving (2.10) on the coarse grid. Thus the state variables $U_h^{(i)}$ and $V_h^{(i)}$ on the fine grid are given by

$$U_h^{(i)} = I_{\tilde{h}}^h U_{\tilde{h}}^{(i)} + (E_1)_h + I_{\tilde{h}}^h (\widetilde{E}_1)_{\tilde{h}} \quad \text{and} \quad V_h^{(i)} = I_{\tilde{h}}^h V_{\tilde{h}}^{(i)} + (E_2)_h + I_{\tilde{h}}^h (\widetilde{E}_2)_{\tilde{h}},$$

where the symbol $I_{\tilde{h}}^h$ denote the interpolation operator from the coarse grid to the fine grid. To guarantee the approximating point is accurate enough on the fine grid, we use $(U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)})$ as an initial guess and perform Newton's method until it converges to the desired solution. Then we finish the corrector step of the inner-loop continuation and go back to the coarse grid to find next approximating point for $\widetilde{c}$. We repeat this process until the solution curve c on the fine grid is traced. The two-grid discretization scheme for (2.2) is described as follows.

Algorithm 3.1. *A two-grid centered difference discretization algorithm with two-loop continuation algorithm for* (2.2).

```
Input:
$\widetilde{h}$ and $h$ := the uniform meshsizes on the x- and y-axis of the coarse and the fine grids, respectively.
$\varepsilon$ := accuracy tolerance of approximating points for both solution curves $\widetilde{c}$ and $c$ on the coarse and
fine grids, respectively.
$i_{max}$ := maximum number of continuation steps.
$(U_{\tilde{h}}^{(0)}, V_{\tilde{h}}^{(0)}, \lambda_{\tilde{h}}^{(0)})$ := starting approximating point for $\widetilde{c}$.
$i = 1$

1. Outer continuation.
   Use the predictor–corrector continuation method to find the centered difference approximate
   point $(U_{\tilde{h}}^{(i)}, V_{\tilde{h}}^{(i)}, \lambda_{\tilde{h}}^{(i)})$ for $\widetilde{c}$ on the coarse grid.
2. Inner continuation.
   (i) Predictor. Set $(I_{\tilde{h}}^h U_{\tilde{h}}^{(i)}, I_{\tilde{h}}^h V_{\tilde{h}}^{(i)}, \lambda_{\tilde{h}}^{(i)})$ as the predicted point.
   (ii) Corrector.
      (a) Make a correction on the fine-grid:
          Solve the linear system associated with (2.5) by using a precondition conjugate gra-
          dient type method to obtain the approximate solution $((E_1)_h, (E_2)_h)$.
      (b) Make a further correction on the coarse-grid:
          Solve the linear system associated with (2.10) to obtain the approximate solution $((\widetilde{E}_1)_{\tilde{h}}, (\widetilde{E}_2)_{\tilde{h}})$.
      (c) Set $U_h^{(i)} = I_{\tilde{h}}^h U_{\tilde{h}}^{(i)} + (E_1)_h + I_{\tilde{h}}^h (\widetilde{E}_1)_{\tilde{h}}$ and $V_h^{(i)} = I_{\tilde{h}}^h V_{\tilde{h}}^{(i)} + (E_2)_h + I_{\tilde{h}}^h (\widetilde{E}_2)_{\tilde{h}}$.
      (d) If $\|\Phi(U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)})\| > \varepsilon$, then use $(U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)})$ as an initial guess and perform Newton's method.
3. If $i = i_{max}$, exit.
   Else, set $i = i + 1$ and go to Step 1.
   End if
```

A few remarks should be given concerning the implementation of Algorithm 3.1.

(1) It is possible to use the approximating point $(I_{\tilde{h}}^{h} U_{\tilde{h}}^{(i)}, I_{\tilde{h}}^{h} V_{\tilde{h}}^{(i)}, \lambda_{\tilde{h}}^{(i)})$ on the coarse grid as an initial guess for Newton's method in the corrector step of the inner-loop continuation. Therefore, we do not have to use the Rayleigh quotient to update the parameter $\lambda_{\tilde{h}}$ in the corrector step of the inner continuation as we did before in [10].

(2) For convenience we denote $(I_{\tilde{h}}^{h} U_{\tilde{h}}^{(i)} + (E_1)_h, I_{\tilde{h}}^{h} V_{\tilde{h}}^{(i)} + (E_2)_h, \lambda_{\tilde{h}}^{(i)})$ by $(\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_{\tilde{h}}^{(i)})$. It could happen that after the fine grid correction, the approximating point $(\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_{\tilde{h}}^{(i)})$satisfies $\|\Phi(\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_{\tilde{h}}^{(i)})\| < \varepsilon$ for some positive constant ε which is sufficiently small. Then we only need to use the Rayleigh quotient to compute the parameter $\lambda_h^{(i)}$. Then the approximating point $(\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_h^{(i)})$ can be accepted as a solution on the fine grid. In this case we can skip the coarse grid correction and the Newton iteration.

(3) If Case (2) happens, then it follows from Case (1) that any point between the approximating point on the coarse grid and the approximating point on the fine grid can be used as an initial guess for Newton's method.

In summary, we have three variants of the corrector step for the inner-loop continuation.

(ii-1)

$$\left\{\begin{array}{l}
\text{(a) Compute the correction } ((E_1)_h, (E_2)_h) \text{ to the desired accuracy on the fine grid.}\\
\text{(b) If } \|\Phi(\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_{\tilde{h}}^{(i)})\| > \varepsilon, \text{ then use } (\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_{\tilde{h}}^{(i)}) \text{ as an initial guess and}\\
\text{perform Newton' s method. Otherwise, use the Rayleigh quotient to compute}\\
\lambda_h^{(i)}, \text{ accept } (\widehat{U}_h^{(i)}, \widehat{V}_h^{(i)}, \lambda_h^{(i)}) \text{ as an approximating point on the fine grid, and go to}\\
\text{Step 1.}
\end{array}\right.$$

(ii-2)

$$\left\{\begin{array}{l}
\text{(a) Do a few iterations for the correction } ((E_1)_h, (E_2)_h) \text{ and obtain } ((\bar{E}_1)_h, (\bar{E}_2)_h)\\
\text{on the fine grid.}\\
\text{(b) Set } U_h^{(i)} = I_{\tilde{h}}^{h} U_{\tilde{h}}^{(i)} + (\bar{E}_1)_h \text{ and } V_h^{(i)} = I_{\tilde{h}}^{h} V_{\tilde{h}}^{(i)} + (\bar{E}_2)_h.\\
\text{(c) If } \|\Phi(U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)})\| > \varepsilon, \text{ then use } (U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)}) \text{ as an initial guess and perform}\\
\text{Newton' s method.}
\end{array}\right.$$

(ii-3)

$$\left\{\begin{array}{l}
\text{(a) Set } U_h^{(i)} = I_{\tilde{h}}^{h} U_{\tilde{h}}^{(i)} \text{ and } V_h^{(i)} = I_{\tilde{h}}^{h} V_{\tilde{h}}^{(i)}.\\
\text{(b) If } \|\Phi(U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)})\| > \varepsilon, \text{ then use } (U_h^{(i)}, V_h^{(i)}, \lambda_{\tilde{h}}^{(i)}) \text{ as an initial guess and}\\
\text{perform Newton' s method.}
\end{array}\right.$$

The two-grid finite element discretization algorithms can be derived in a similar way and is omitted here.

4. Applications to polar coordinates

For simplicity we consider a single nonlinear Schrödinger equation

$$\begin{aligned}
&-\Delta u - \lambda u + Vu + \mu u^3 = 0 \quad \text{in } \Omega,\\
&u = 0 \quad \text{on } \partial\Omega, \qquad (4.1)
\end{aligned}$$

where $\Omega = \{(x_1, x_2) \in \mathbf{R}^2 : x_1^2 + x_2^2 < 1\}$ is the unit circle. Let $x_1 = r\cos\theta$, $x_2 = r\sin\theta$, and set $\widehat{u}(r, \theta) = u(r\cos\theta, r\sin\theta)$. Then (4.1) becomes

$$\begin{aligned}
&-\left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2}\right) - \lambda u + Vu + \mu u^3 = 0, \quad 0 < r < 1, \ \ 0 \leqslant \theta < 2\pi,\\
&u(1, \theta) = 0, \quad 0 \leqslant \theta < 2\pi, \qquad (4.2)
\end{aligned}$$

where we have dropped the "$\wedge$" sign in (4.2). We discretize (4.2) by the centered difference approximations described in [17] with uniform mesh width $\Delta r = 2/(2N+1)$ on the radial direction and $\Delta\theta = 2\pi/M$ on the azimuthal direction for some positive integers M and N. The locations of grid points are half integered in the radial direction and integered in the azimuthal direction, i.e.,

$$r_i = (i - \tfrac{1}{2})\Delta r, \quad \theta_j = (j-1)\Delta\theta, \ \ i = 1, 2, \ldots, N+1, \ \ j = 1, 2, \ldots, M+1.$$

Let $U_{ij} = u(r_i, \theta_j)$. Then the centered difference analogue of (4.2) is

$$\begin{aligned} &-\left(\frac{U_{i+1,j} - 2U_{ij} + U_{i-1,j}}{(\Delta r)^2} + \frac{1}{r_i}\frac{U_{i+1,j} - U_{i-1,j}}{2\Delta r} + \frac{1}{r_i^2}\frac{U_{i,j+1} - 2U_{ij} + U_{i,j-1}}{(\Delta\theta)^2}\right) \\ &\quad - \lambda U_{ij} + VU_{ij} + \mu U_{ij}^3 = 0 \quad \text{in } \Omega, \\ &U_{N+1,j} = 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{4.3}$$

Since $u(r,\theta)$ is 2π periodic in θ, we have $U_{i,0} = U_{i,M}$ and $U_{i,1} = U_{i,M+1}$. We can order the grid points either in the radial direction or in the azimuthal direction. In both cases we obtain two nonsymmetric but similar matrices. We refer to [7] for details.

The derivation of two-grid discretization schemes for (4.1) is the same as those given in Section 2 and is omitted here. In conclusion, suppose that $\Delta\tilde{r}$ is the radial meshsize on the coarse grid. To avoid the singularity at the origin, we have to choose $\Delta r = (1/3^n)\Delta\tilde{r}$ for some positive integer n as the radial meshsize on the fine grid.

5. Discrete systems

5.1. TCNLS

Eq. (1.4) can be expressed as

$$F(u_1, u_2, \lambda_1, \lambda_2) = \begin{bmatrix} -\Delta u_1 + f(u_1, u_2, \lambda_1) \\ -\Delta u_2 + g(u_1, u_2, \lambda_2) \end{bmatrix} = 0. \tag{5.1}$$

Let $\mathbf{u} = (u_1, u_2)$, $C_0^2(\Omega) := \{u \in C^2(\Omega) \mid u|_{\partial\Omega} = 0\}$, and $X = (C_0^2(\Omega))^2$, $Y = (C^0(\Omega))^2$. Then $F : X \times \mathbf{R}^2 \to Y$ is a smooth mapping. Note that $\mathbf{u}_0 = (0,0)$ is a trivial solution of (5.1) for all $\lambda \in \mathbf{R}^2$. Differentiating F with respect to $\mathbf{u}$ at the homogeneous equilibrium $\mathbf{u}_0 = (0,0)$, we obtain the linearization L of F, namely,

$$\begin{aligned} L := D_{\mathbf{u}}F(\mathbf{u}_0, \lambda_1, \lambda_2) &= \begin{bmatrix} -\varDelta + \dfrac{\partial f}{\partial u_1}(0, \lambda_1) & \dfrac{\partial f}{\partial u_2}(0, \lambda_1) \\ \dfrac{\partial g}{\partial u_1}(0, \lambda_2) & -\Delta + \dfrac{\partial g}{\partial u_2}(0, \lambda_2) \end{bmatrix} \\ &= \begin{bmatrix} -\varDelta - \lambda_1 I & 0 \\ 0 & -\varDelta - \lambda_2 I \end{bmatrix}, \end{aligned} \tag{5.2}$$

where $L : X \times \mathbf{R}^2 \to Y$. We discretize (5.1) by the centered difference approximations with uniform meshsize $h = 1/(N+1)$ on the x- and y-axis. The centered difference analogue of (5.1) can be expressed as

$$F(U, V, \lambda_1, \lambda_2) = \begin{cases} A_1 U - \lambda_1 U + \mu_1 U^3 + \beta V^2 \circ U = 0, \\ A_1 V - \lambda_2 V + \mu_2 U^3 + \beta U^2 \circ V = 0. \end{cases} \tag{5.3}$$

Here $A_1 \in \mathbf{R}^{N^2 \times N^2}$ is the coefficient matrix associated with the discretization of the Laplacian $-\varDelta$, $U = [U_1, U_2, \ldots, U_{N^2}]^{\mathrm{T}}$, $V = [V_1, V_2, \ldots, V_{N^2}]^{\mathrm{T}}$ with $h = 1/(N+1)$ on the x- and y-axis, $U \circ V = [U_1V_1, U_2V_2, \ldots, U_{N^2}V_{N^2}]$ denotes the Hadamard product of U and V, and $U^r = U \circ \cdots \circ U$, the r-times Hadamard products of U. We denote $X = [U, V]^{\mathrm{T}}$

and treat λ_1 as the continuation parameter by varying the values of λ_2 and β. Note that $F : \mathbf{R}^M \times \mathbf{R} \to \mathbf{R}^M$ is a smooth mapping with $M = 2N^2$. For convenience we fix λ_2. Let the Jacobian matrix of F be denoted by $DF = [D_X F, D_{\lambda_1} F] \in \mathbf{R}^{(M+1)\times M}$. We have

$$D_X F = \begin{bmatrix} A_1 + \mathrm{diag}(-\lambda_1 + 3\mu_1 U^2 + \beta V^2) & \mathrm{diag}(2\beta V \circ U) \\ \mathrm{diag}(2\beta U \circ V) & A_1 + \mathrm{diag}(-\lambda_2 + 3\mu_2 V^2 + \beta U^2) \end{bmatrix}$$

and

$$D_{\lambda_1} F = \begin{bmatrix} -U \\ 0 \end{bmatrix}. \tag{5.4}$$

Note that A_1 is symmetric and positive definite, and $D_X F$ is symmetric. The discrete operator corresponding to the linear operator L in (5.2) is denoted by

$$A = \begin{bmatrix} A_1 - \lambda_1 I & 0 \\ 0 & A_1 - \lambda_2 I \end{bmatrix} \in \mathbf{R}^{M\times M}. \tag{5.5}$$

The eigenvalues and corresponding eigenvectors of A_1 are

$$\lambda_{p,q} = 4(N+1)^2 \left(\sin^2 \frac{p\pi}{2(N+1)} + \sin^2 \frac{q\pi}{2(N+1)} \right), \quad 1 \leqslant p,\ q \leqslant N,$$

$$U_{p,q}(x_j, y_k) = \sin \frac{jp\pi}{N+1} \sin \frac{kq\pi}{N+1}. \tag{5.6}$$

We can fix λ_2 so that the matrix A in (5.5) becomes singular if $\lambda_1 = \lambda_{p,q}$. Thus the bifurcation point of the discrete two-coupled nonlinear Schrödinger equations occur on the trivial solution curve at $\{(U, V, \lambda_1) = (0, 0, \lambda_{p,q}) \mid p, q = 1, 2, \ldots, N\}$.

5.2. The Brusselator

The Brusselator is governed by

$$\frac{\partial u}{\partial t} = d_1 \Delta u + f(u, v, \lambda) = d_1 \Delta u - (\lambda + 1)u + u^2 v + \alpha \quad \text{in } \Omega = [0, \ell] \times [0, 1],$$

$$\frac{\partial v}{\partial t} = d_2 \Delta v + g(u, v, \lambda) = d_2 \Delta v + \lambda u - u^2 v \quad \text{in } \Omega = [0, \ell] \times [0, 1],$$

$$u(x, y, t) = \alpha, \quad v(x, y, t) = \frac{\lambda}{\alpha} \quad \text{on } \partial\Omega. \tag{5.7}$$

Eq. (5.7) has a uniform steady state solution $(u_0, v_0) = (\alpha, \lambda/\alpha)$. We treat λ as the bifurcation parameter and fix α. Let $\mathbf{u} = (u, v)$. Shifting $\mathbf{u}_0 = (\alpha, \lambda/\alpha)$ to $\mathbf{u}_0 = (0, 0)$, we obtain

$$f(\mathbf{u}, \lambda) = (\lambda - 1)u + \alpha^2 v + ((\lambda/\alpha)u^2 + 2\alpha uv + u^2 v),$$

$$g(\mathbf{u}, \lambda) = -\lambda u - \alpha^2 v - ((\lambda/\alpha)u^2 + 2\alpha uv + u^2 v). \tag{5.8}$$

Thus the steady state equations of (5.7) can be expressed as

$$F(\mathbf{u}, \lambda) = \begin{bmatrix} d_1 \Delta u + f(\mathbf{u}, \lambda) \\ d_2 \Delta v + g(\mathbf{u}, \lambda) \end{bmatrix} = 0, \tag{5.9}$$

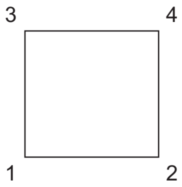

Fig. 1. An Adini's element $\square_{ij}$.

Table 1
The total execution time (in seconds) and total number of iterations on the fine gird for tracing the solution branch of (6.1) bifurcating at $(0, \lambda^*_{1,1})$ by implementing Methods 1, 2, 4 with six-node triangular elements

	Method 1	Method 2	Method 4
(a) $\mu = 30$			
Time(s)	2489.22	2105.08	3799.11
Total Newton iterations	13	0	99
Total linear systems	126	100	198
Total Lanczos iterations	94581	79704	144723
(b) $\mu = -30$			
Time(s)	2495.63	2139.39	3867.81
Total Newton iterations	13	0	99
Total linear systems	126	100	198
Total Lanczos iterations	94765	81033	147127

where $f(\mathbf{u}, \lambda)$ and $g(\mathbf{u}, \lambda)$ are defined as in (5.8). Differentiating F with respect to $\mathbf{u}$ at the homogeneous equilibrium $\mathbf{u}_0 = (0, 0)$, we obtain the linearization L of F with

$$L := D_{\mathbf{u}}F(0,\lambda) = \begin{bmatrix} d_1\Delta + \dfrac{\partial f}{\partial u}(0,\lambda) & \dfrac{\partial f}{\partial v}(0,\lambda) \\ \dfrac{\partial g}{\partial u}(0,\lambda) & d_2\Delta + \dfrac{\partial g}{\partial v}(0,\lambda) \end{bmatrix} = \begin{bmatrix} d_1\Delta + (\lambda-1)I & \alpha^2 I \\ -\lambda I & d_2\Delta - \alpha^2 I \end{bmatrix}. \tag{5.10}$$

To simplify our computations, we choose $\ell = 1$. The centered difference analogue of (5.10) is given by

$$A = \begin{bmatrix} -d_1A_1 + (\lambda-1)I & \alpha^2 I \\ -\lambda I & -d_2A_1 - \alpha^2 I \end{bmatrix}, \tag{5.11}$$

where A_1 is defined as in (5.4). Let $\lambda_{p,q}$ be defined as in (5.6). Similar to the discussion given in [9,11], the eigenvalues $\lambda^*_{p,q}$ of A are determined by those of the 2×2 matrices

$$B_{p,q} = \begin{bmatrix} -d_1\lambda_{p,q} + (\lambda^*_{p,q} - 1) & \alpha^2 \\ -\lambda^*_{p,q} & -d_2\lambda_{p,q} - \alpha^2 \end{bmatrix}, \quad p, q = 1, \ldots, N.$$

Since $\det B_{p,q} = d_1d_2\lambda^2_{p,q} - (d_2\lambda^*_{p,q} - d_2 - \alpha^2 d_1)\lambda_{p,q} + \alpha^2 = 0$, therefore

$$\lambda^*_{p,q} = d_1\lambda_{p,q} + \left(\frac{1}{\lambda_{p,q}} + d_1\right)\frac{\alpha^2}{d_2} + 1, \quad p, q = 1, \ldots, N. \tag{5.12}$$

Thus the bifurcation points of the Brusselator occur on the trivial solution curve at $\{(0, 0, \lambda^*_{p,q}) \mid p, q = 1, \ldots, N\}$, where $\lambda^*_{p,q}$ are defined in (5.12).

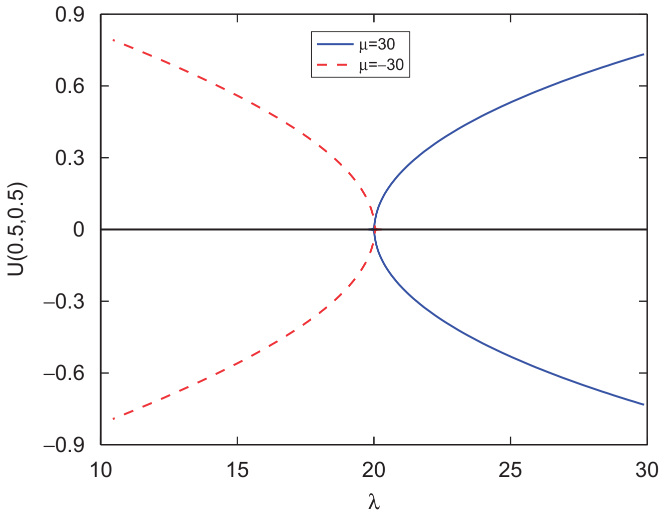

Fig. 2. The first solution branches of (6.1) with $\mu = 30$ and $\mu = -30$, respectively.

Table 2
Locations of the first bifurcation points of (6.1)

Method	Meshsize	Matrix order	Bifurcation $(0, \lambda_h)$	$\lvert\lambda_h - \lambda_h^*\rvert$	Time(s)
Triangular elements	$h = \frac{1}{64}$	4225 × 4225	(0, 20.02088404)	4.40×10^{-4}	129.95
Triangular elements	$h = \frac{1}{128}$	16641 × 16641	(0, 20.02090534)	4.19×10^{-4}	924.77
Triangular elements	$h = \frac{1}{256}$	66049 × 66049	(0, 20.02093030)	3.94×10^{-4}	6637.56
Adini's elements	$h = \frac{1}{32}$	3267 × 3267	(0, 20.02091925)	4.05×10^{-4}	198.13
Two-grid solution $(0, \lambda_h^*) = (0, 20.02132439)$					

Table 3
The total execution time (in seconds) and total number of iterations on the fine gird for tracing the solution branch of (1.4) bifurcating at $(0, 0, \lambda_{1,1})$ by implementing Methods 1–4 and the single-grid continuation method, $V_1 = V_2 = 0$, $\mu_1 = 10$, $\mu_2 = 5$, $\lambda_2 = 30$, $\beta = 300$, and $\Omega = (0, 1)^2$

	Method 1	Method 2	Method 3	Method 4	Single-grid
Time(s)	552.48	350.73	630.69	542.95	769.33
Total Newton iterations	53	12	99	99	118
Total linear systems	206	124	298	198	336
Total preconditioned Lanczos iterations	18111	11438	20695	17983	25248

5.3. The Adini elements

Let the domain S be split into small rectangles $\Box_{ij}$, i.e., $S = \cup_{ij}\Box_{ij}$. Denote by h_i and k_j the boundary lengths of $\Box_{ij}$, and $h = \max_{i,j}\{h_i, k_j\}$. The rectangles $\Box_{ij}$ are said to be quasiuniform if

$$\frac{h}{\min_{i,j}\{h_i, k_j\}} \leqslant c$$

for some constant c. The quasiuniform elements are said to be uniform if $h_i = h$ and $k_j = k$. Without loss of generality we may assume $h \geqslant k$. Let $V_A \subset H_0^1(\Omega)$ be the finite dimensional subspace spanned by the Adini elements,

$$V_A = \{v \in H_0^1(S) : v|\Box_{ij} \in \widehat{P}_3,\ v_x \text{ and } v_y \text{ are continuous at all vertices of } \Box_{ij}\},$$

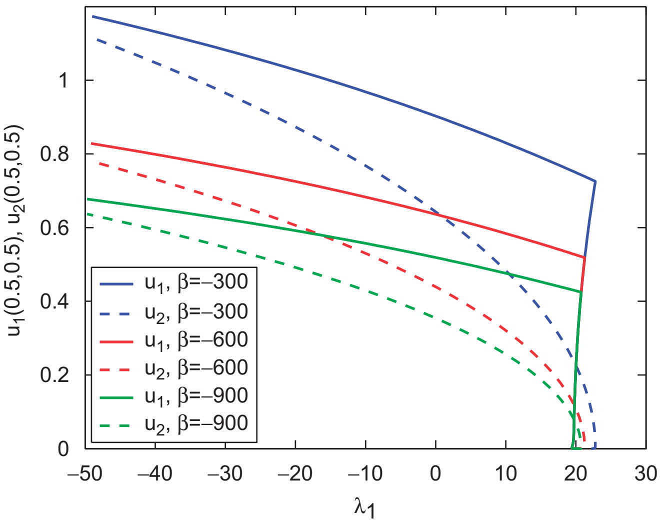

Fig. 3. The solution curves of u_1 and u_2 of (1.4) with $V_1 = V_2 = 0$, $\mu_1 = 10$, $\mu_2 = 5$, $\lambda_2 = 30$, and $\beta = -300, -600, -900$.

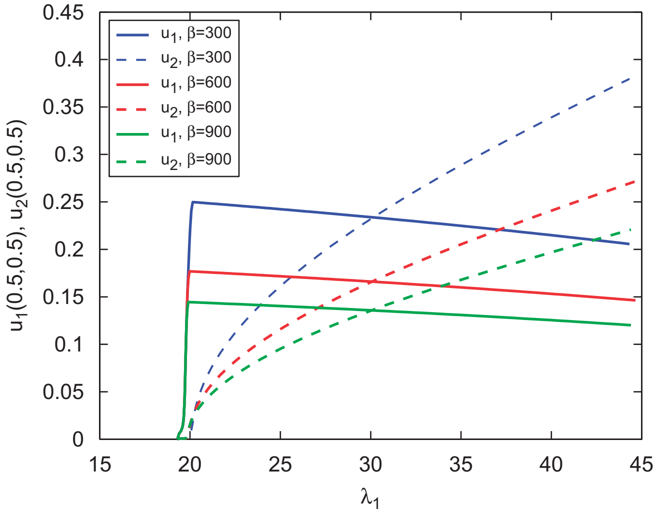

Fig. 4. The solution curves of u_1 and u_2 of (1.4) with $V_1 = V_2 = 0$, $\mu_1 = 10$, $\mu_2 = 5$, $\lambda_2 = 30$, and $\beta = 300, 600, 900$.

where

$$\widehat{P}_3 = \text{span}\{1, x, y, x^2, y^2, xy, x^3, y^3, x^2y, xy^2, x^3y, xy^3\}. \tag{5.13}$$

Note that the Adini elements are conforming for Poisson's equation but not for biharmonic equations.

Denote $\square_{ij} = \{(x, y) : x_i \leqslant x \leqslant x_{i+1}, y_j \leqslant y \leqslant y_{j+1}\}$. We choose the affine transformations $\xi = (x - x_i)/h_i$ and $\eta = (y - y_j)/k_j$, where $h_i = x_{i+1} - x_i$ and $k_j = y_{j+1} - y_j$. The admissible functions for the Adini elements are given by

$$v(x, y) = \sum_{l=1}^{4} v_l \phi_l(\xi, \eta) + h_i \sum_{l=1}^{4} (v_x)_l \psi_l(\xi, \eta) + k_j \sum_{l=1}^{4} (v_y)_l \theta_l(\xi, \eta). \tag{5.14}$$

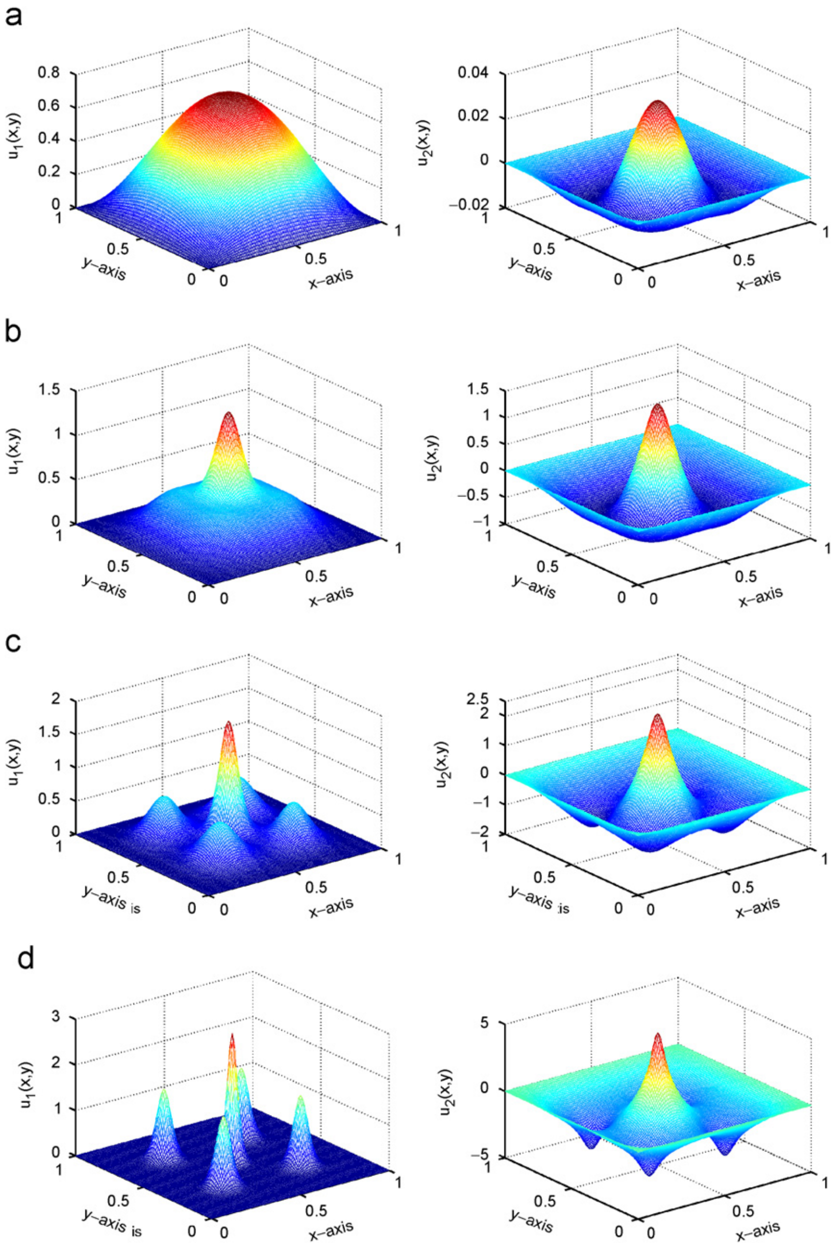

Fig. 5. The contours of the solutions u_1 and u_2 of (1.4) with $V_1=V_2=0$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=30$, and $\beta=-300$ at $\lambda_1=22.7315546$, -94.0162850, -396.206293, -2947.69410, respectively: (a) $\lambda_1=22.7315546$, (b) $\lambda_1=-94.0162850$, (c) $\lambda_1=-396.206293$, (d) $\lambda_1=-2947.69410$.

where the nodal points 1, 2, 3, 4 denote the coordinates (i, j), $(i+1, j)$, $(i, j+1)$, and $(i+1, j+1)$, respectively. See Fig. 1

.

The 12 basis functions on $S=[0,1]^2$ are given explicitly by

$$\phi_1(x, y) = (1-x)(1-y)(1+x+y-2x^2-2y^2),$$

$$\phi_2(x, y) = x(1-y)(3x+y-2x^2-2y^2),$$

$$\phi_3(x, y) = (1-x)y(x+3y-2x^2-2y^2),$$

Table 4
The total execution time (in seconds) and total number of iterations on the fine gird for tracing the first solution branch of (1.4) defined on the unit circle by implementing Methods 1, 2, 4, and the single-grid continuation method, $V_1 = (x_1^2 + x_2^2)/2$, $V_2 = (x_1^2 + x_2^2)/5$, $\mu_1 = 10$, $\mu_2 = 5$, $\lambda_2 = 10$, and $\beta = 300$.

	Method 1	Method 2	Method 4	Single-grid
Time(s)	178.92	114.92	182.18	249.15
Total Newton iterations	39	14	110	201
Total linear systems	178	128	220	402
Total Bi-CGSTAB iterations	65975	43864	70560	95803

$$\begin{aligned}
&\phi_4(x, y) = xy(-1 + 3x + 3y - 2x^2 - 2y^2),\\
&\phi_5(x, y) = \psi_1(x, y) = (1 - y)\widehat{\psi}_0(x),\\
&\phi_6(x, y) = \psi_2(x, y) = (1 - y)\widehat{\psi}_1(x),\\
&\phi_7(x, y) = \psi_3(x, y) = y\widehat{\psi}_0(x),\\
&\phi_8(x, y) = \psi_4(x, y) = y\widehat{\psi}_1(y),\\
&\phi_9(x, y) = \theta_1(x, y) = (1 - x)\widehat{\psi}_0(y),\\
&\phi_{10}(x, y) = \theta_2(x, y) = x\widehat{\psi}_0(y),\\
&\phi_{11}(x, y) = \theta_3(x, y) = (1 - x)\widehat{\psi}_1(y),\\
&\phi_{12}(x, y) = \theta_4(x, y) = x\widehat{\psi}_1(y),
\end{aligned} \tag{5.15}$$

where

$$\widehat{\psi}_0(s) = s^3 - 2s^2 + s, \quad \widehat{\psi}_1(s) = s^3 - s^2 \tag{5.16}$$

are the cubic Hermite basis functions on [0, 1].

6. Numerical results

The numerical algorithms described in Section 3 were implemented to traced solution branches of the TCNLS, where $\widetilde{h} = \frac{1}{16}$ and $h = \frac{1}{256}$. For convenience we denote Algorithm 3.1 and the three variants of Algorithm 3.1, i.e., replacing Step 2(ii) of Algorithm 3.1 by the corrector procedures (ii-1)–(ii-3) as Methods 1–4, respectively. The TCNLS were discretized by the centered difference approximation with uniform meshsize on the x- and y-axis for the rectangular domains $\Omega = (0, 1)^2$ and $(-6, 6) \times (-6, 6)$, and on the radial- and azimuthal-direction for the unit disk. We also compare the performance of six-node triangular elements and the Adini elements on a single grid. As a side application Algorithm 3.1 was also implemented to trace solution branches of the Brusselator. The accuracy tolerances for the linear solvers and the Newton corrector are 5×10^{-10} and 5×10^{-8}, respectively. All computations were executed on a Pentium 4 computer using FORTRAN 95 Language with double precision arithmetic. In these examples we computed 100 approximating points on the solution curves branching from the first bifurcation points.

Example 1 (*A single NLS*). We consider a single NLS

$$\begin{aligned}
&-\Delta u(x) + V(x)u(x) + \mu|u(x)|^2u(x) = \lambda u(x), \quad x \in \Omega = (0, 1)^2,\\
&u(x) = 0, \quad x \in \partial\Omega,
\end{aligned} \tag{6.1}$$

where $V(x) = \frac{1}{2}(x_1^2 + x_2^2)$. To start with, we used a two-grid finite element discretization scheme with $\widetilde{h} = \frac{1}{16}$ and $h = \frac{1}{256}$ to compute the first eigenpair of the linear Schrödinger eigenvalue problem

$$-\Delta u(x) + V(x)u(x) = \lambda u(x), \quad x \in \Omega,$$

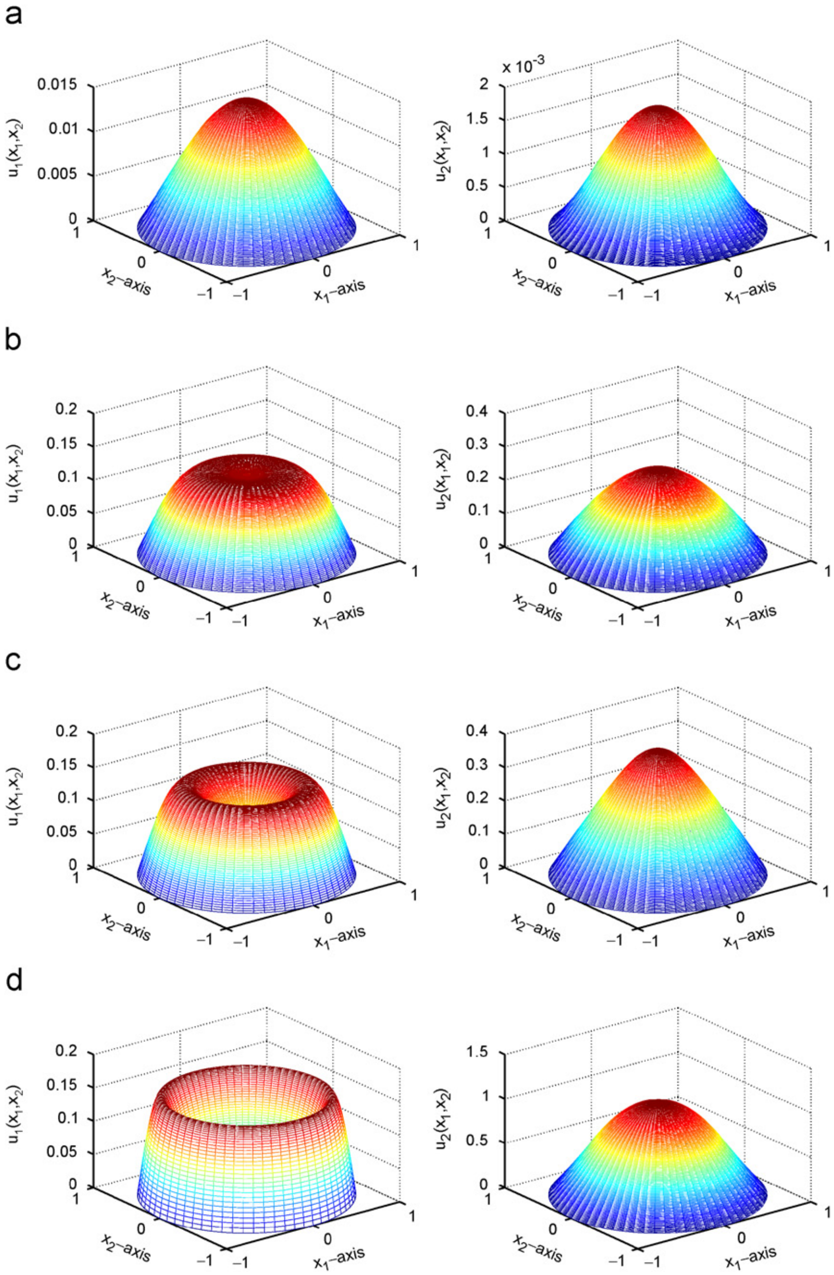

Fig. 6. The contours of the solutions u_1 and u_2 of (1.4) with $V_1=(x_1^2+x_2^2)/2$, $V_2=(x_1^2+x_2^2)/5$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=10$, and $\beta=300$ at $\lambda_1=5.75096348$, 16.4065407, 23.0804876, 73.0623656, respectively: (a) $\lambda_1=5.75096348$, (b) $\lambda_1=16.4065407$, (c) $\lambda_1=f$, (d) $\lambda_1=73.0623656$.

$$u(x)=0, \quad x\in\partial\Omega. \tag{6.2}$$

The first discrete eigenvalue of (6.2) on the fine grid is $\lambda_h^*=20.02132439$, where the residual norm is 10^{-9}. Next, we used the two-grid six-node triangular discretization scheme to trace the solution curve of (6.1) branching from the first bifurcation point $(0,\lambda_h^*)\approx(0, 20.02132439)$, where Methods 1, 2, and 4 were implemented with the Lanczos method as the linear solver. Table 1 lists the total execution time as well as the total Lanczos and Newton iterations. Fig. 2 displays the first solution branches of (6.1) with $\mu=30$ and $\mu=-30$, respectively. Both the six-node triangular elements and the Adini elements were also implemented on a single grid to trace the first solution branch of (6.1). Table 2 lists the locations of the detected bifurcation points $(0,\lambda_h)$ with various grid sizes.

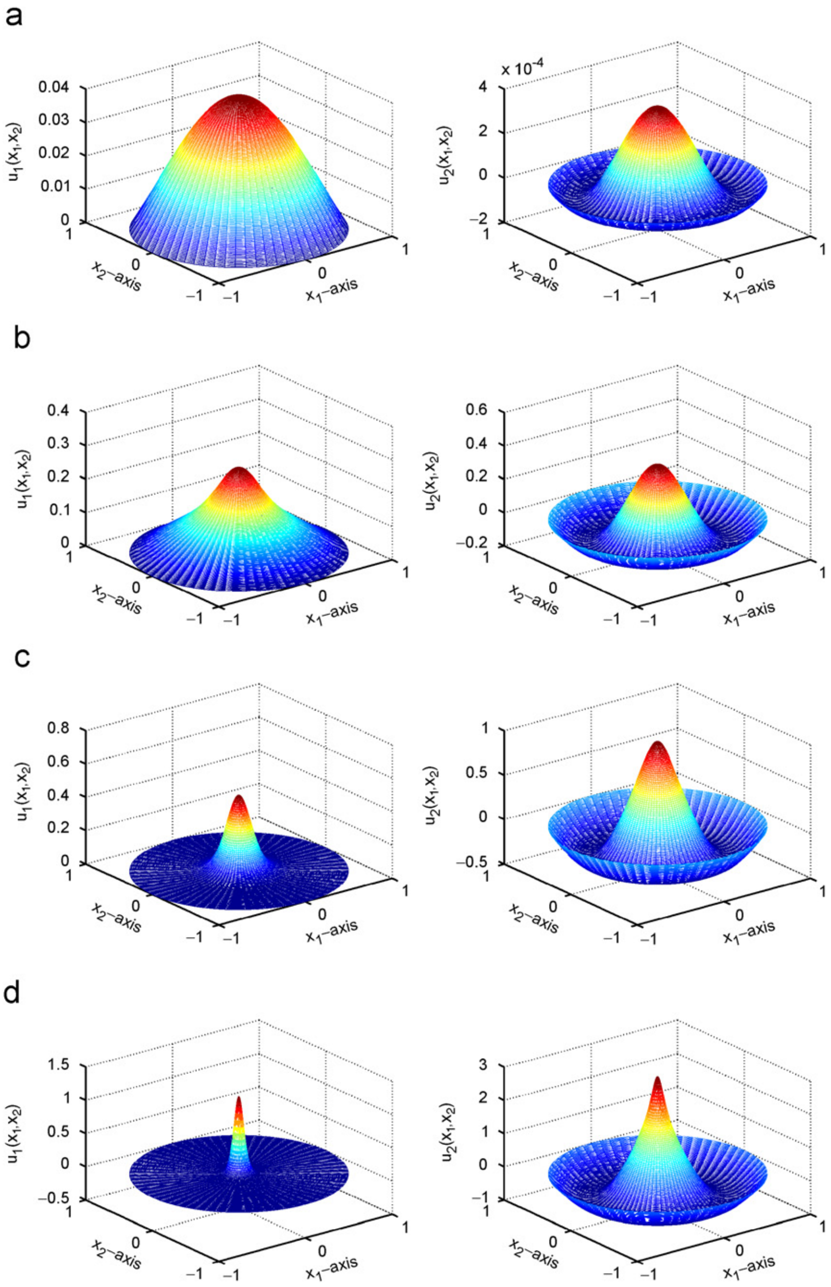

Fig. 7. The contours of the solutions u_1 and u_2 of (1.4) with $V_1=(x_1^2+x_2^2)/2$, $V_2=(x_1^2+x_2^2)/5$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=25$, and $\beta=-300$ at $\lambda_1=5.84931608, -1.01785437, -112.638902, -1490.83931$, respectively: (a) $\lambda_1=5.84931608$, (b) $\lambda_1=-1.01785437$, (c) $\lambda_1=-112.638902$, (d) $\lambda_1=-1490.83931$.

Example 2 (*TCNLS on the unit square*). We chose $V_1=V_2=0$, $\mu_1=10$, $\mu_2=5$ and $\lambda_2=30$ in (1.4), and treated λ_1 as the continuation parameter. Table 3 lists the total execution time as well as the total preconditioned Lanczos and Newton iterations, where Methods 1–4 and the single-grid continuation method were implemented to trace the first solution branch. Fig. 3 shows that the solution curves of u_1 and u_2 are pitchfork and subcritical, where $\beta=-300$, -600 and -900. That is, the solution curves turn to the left of the bifurcation point. Fig. 4 shows that the solution curves of u_1 and u_2 are pitchfork and supercritical, where $\beta=300$, 600 and 900. The contours of the solution curves of u_1 and u_2 with $\beta=-300$ at $\lambda_1=22.7315546$, -94.0162850, -396.206293, and -2947.69410 are displayed in Fig. 5.

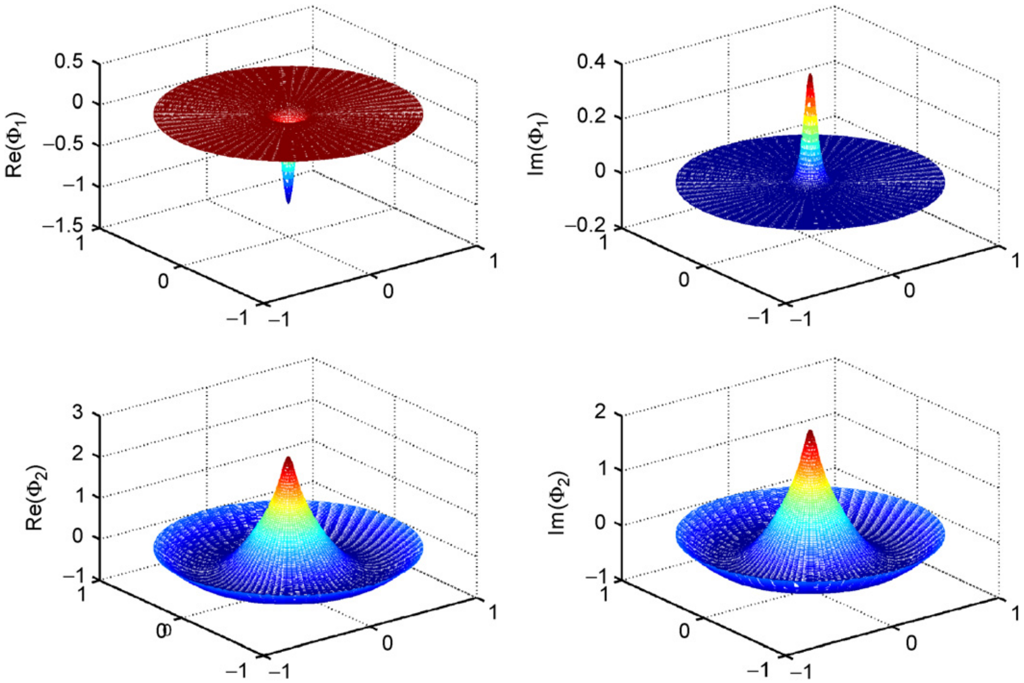

Fig. 8. The contours of the real and imaginary parts of the wave solutions Φ_j, $j=1,2$ with $\lambda_1=-1490.83931$, $\lambda_2=25.0$, respectively, at $t=100$.

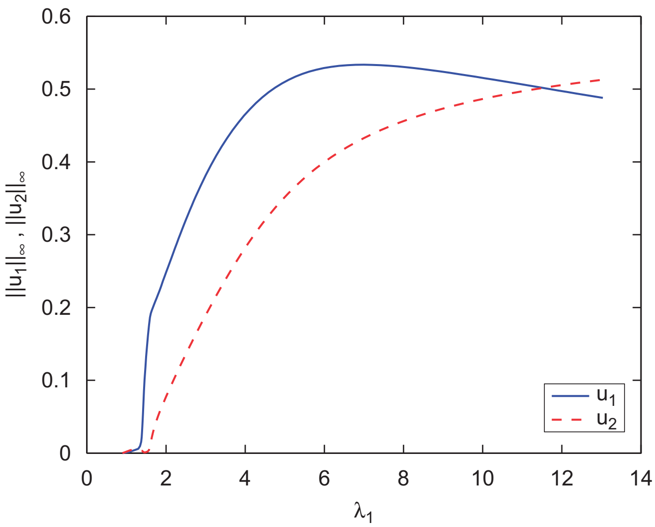

Fig. 9. The solution curves of u_1 and u_2 of (1.4) with $\Omega=(-6,6)^2$, $V_1=(x_1^2+x_2^2)/2$, $V_2=(x_1^2+x_2^2)/5$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=2.5$, and $\beta=300$.

Example 3 (*TCNLS on the unit circle*). We study the two-coupled NLS as in (1.4), where $V_1=(x_1^2+x_2^2)/2$, $V_2=(x_1^2+x_2^2)/5$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=10$ or 25, $\beta=300$ or -300, and treated λ_1 as the continuation parameter. In Table 4 we list the total execution time as well as the total Bi-CGSTAB and Newton iterations, where Methods 1, 2, 4, and the single-grid continuation method were implemented to trace the first solution branch. We observe that the two-grid methods are better than the single-grid method and Method 2 is superior to the other two two-grid methods. The contours of the solution curves of u_1 and u_2 with $\lambda_2=10$ and $\beta=300$ at $\lambda_1=5.75096348$, 16.4065407, 23.0804876, and 73.0623656 are displayed in Fig. 6. The contours of the solution curves of u_1 and u_2 with $\lambda_2=25$ and $\beta=-300$

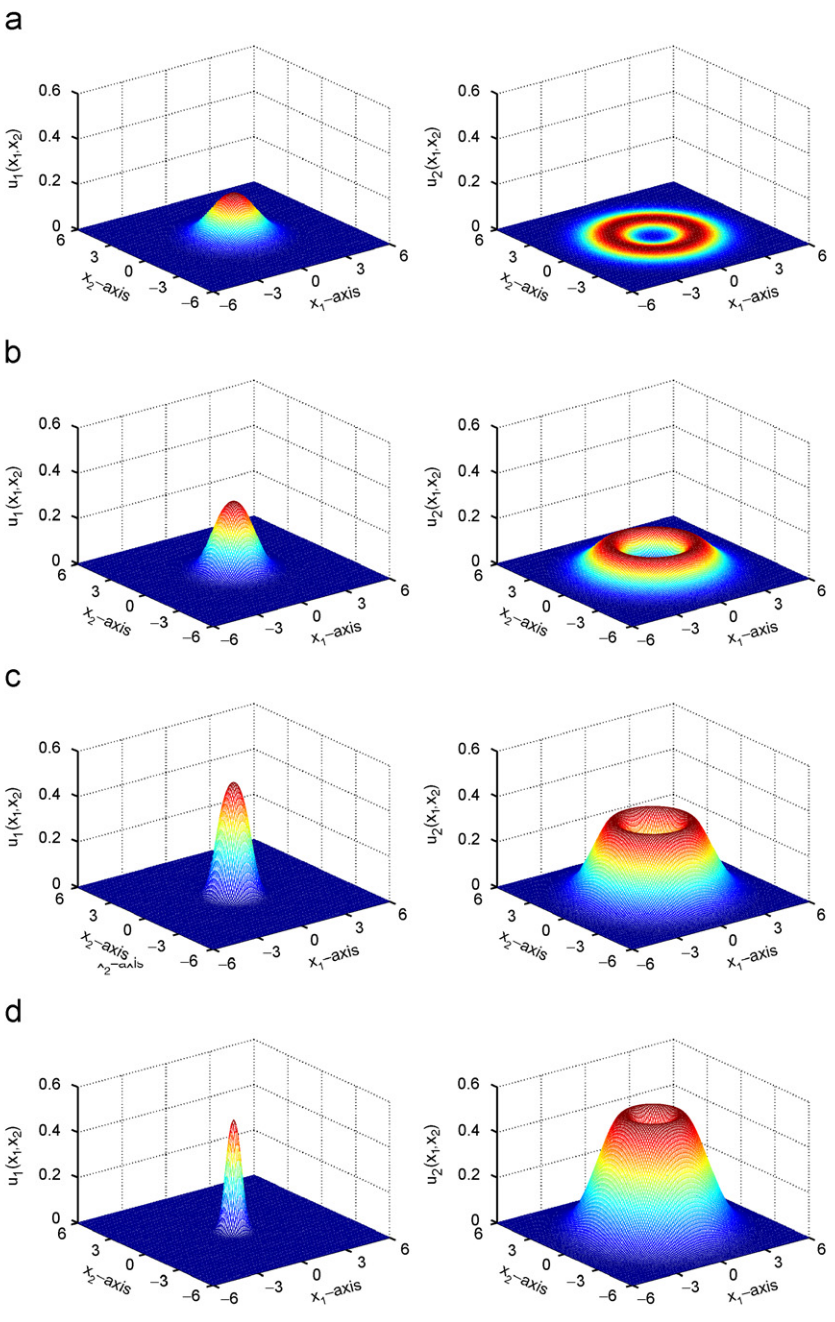

Fig. 10. The contours of the solutions u_1 and u_2 of (1.4) with $\Omega=(-6,6)^2$, $V_1=(x_1^2+x_2^2)/2$, $V_2=(x_1^2+x_2^2)/5$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=2.5$, and $\beta=300$ at $\lambda_1=1.61545937, 2.41149804, 4.59511622, 13.0373400$, respectively: (a) $\lambda_1=1.61545937$, (b) $\lambda_1=2.41149804$, (c) $\lambda_1=4.59511622$, (d) $\lambda_1=13.0373400$.

at $\lambda_1=5.84931608$, -1.01785437, -112.638902, and -1490.83931 are displayed in Fig. 7. Fig. 8 displays the wave functions Φ_1 and Φ_2 at $\lambda_1=-1490.83931$, $\lambda_2=25$, and $t=100$.

Example 4 (*TCNLS on the square $\Omega=(-\ell,\ell)^2$*). We study the two-coupled NLS (1.4) defined on $\Omega=(-\ell,\ell)^2$, where $\ell=6$, $V_1=(x_1^2+x_2^2)/2$, $V_2=(x_1^2+x_2^2)/5$, $\mu_1=10$, $\mu_2=5$, $\lambda_2=2.5$, $\beta=300$, and treated λ_1 as the continuation parameter. Methods 1–4 were implemented to traced solution branches of this example, where we chose

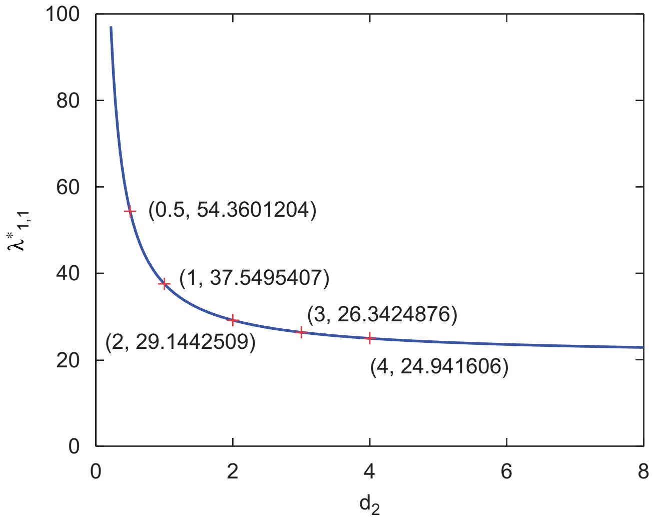

Fig. 11. The bifurcation curve of (5.9) with $d_1 = 1$, $\alpha = 4$, $h = \frac{1}{256}$.

Table 5
The total execution time (in seconds) and total number of iterations on the fine gird for tracing the solution branch of (5.9) bifurcating at $(0, 0, \lambda^*_{1,1})$

	Method 1	Method 2	Method 3	Method 4	Single-grid
(a) *Right solution branch of u and v*					
Time(s)	3565.23	1249.34	2328.91	2494.53	3732.38
Total Newton iterations	96	0	90	99	104
Total linear systems	292	100	280	198	308
Total Bi-CGSTAB iterations	152226	53377	99149	106447	158853
(b) *Left solution branch of u and v*					
Time(s)	3739.50	3046.36	2529.66	2567.72	3838.73
Total Newton iterations	98	89	95	99	102
Total linear systems	296	278	290	198	304
Total Bi-CGSTAB iterations	160010	130093	107694	109710	163867

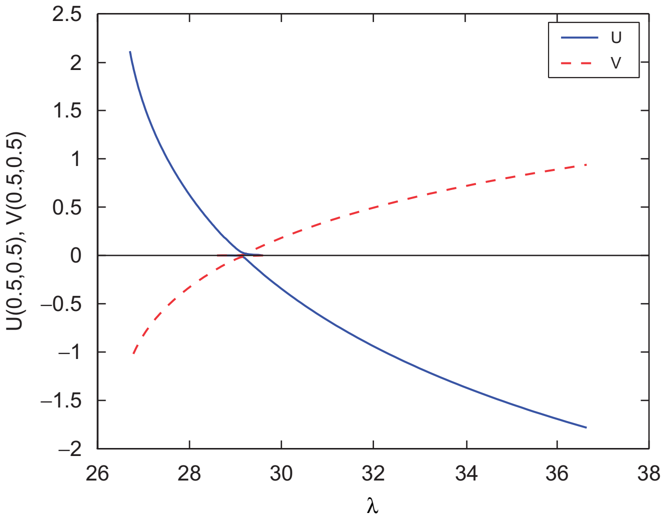

Fig. 12. The solution curves of u and v branching from the first bifurcation point $(0, 0, \lambda^*_{1,1})$ of (5.9) with $d_1 = 1$, $d_2 = 2$, $\alpha_2 = 4$, and $\ell = 1$.

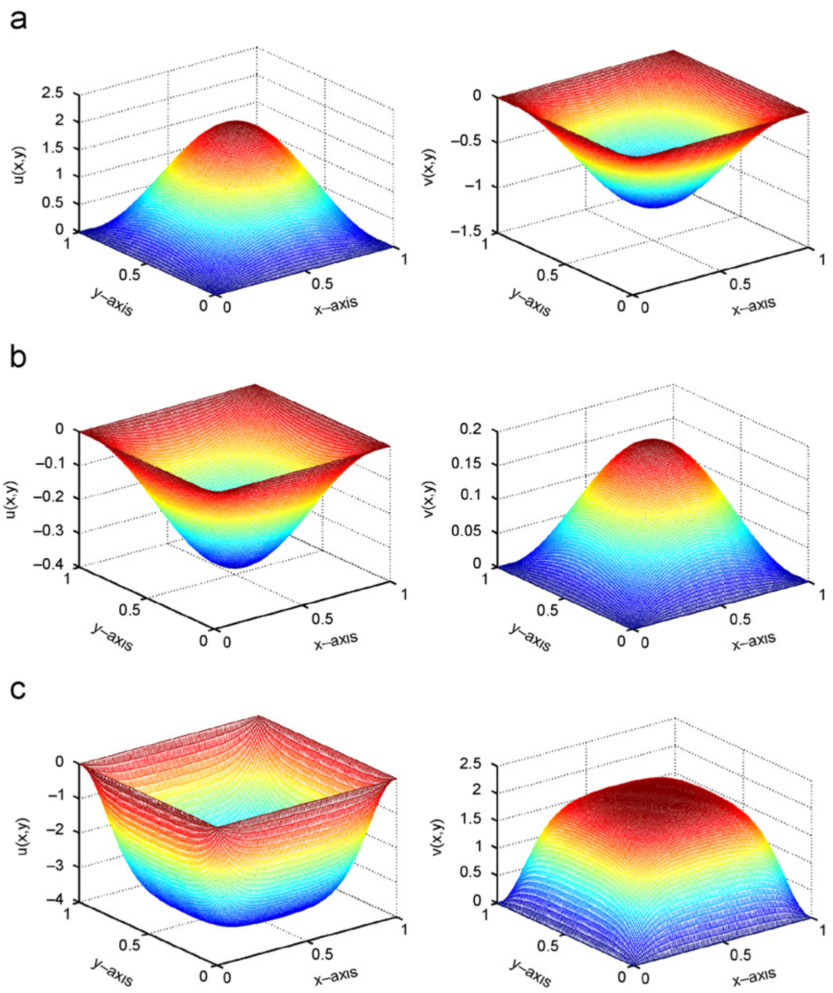

Fig. 13. The contours of the solutions u and v of (5.9) with $d_1 = 1$, $d_2 = 2$, $\alpha = 4$, and $\ell = 1$ at: (a) $\lambda = 26.7036738$, (b) $\lambda = 30.0809857$, (c) $\lambda = 228.491101$, respectively.

$\widetilde{h} = 2\ell/32$ and $h = 2\ell/256$. The first discrete bifurcation points on the coarse and fine grids are $(u_1, u_2, \lambda_1) \approx (0, 0, 1.40536883)$ and $(0, 0, 1.41407622)$, respectively. Fig. 9 shows the first solution branches of u_1 and u_2. The contours of the solution curves of u_1 and u_2 at $\lambda_1 = 1.61545937$, 2.41149804, 4.59511622, and 13.0373400 are displayed in Fig. 10.

Example 5 (*The Brusselator*). In this example, we chose $\ell = 1$, $d_1 = 1$, $d_2 = 2$ and $\alpha = 4$ in (5.9).The first bifurcation point on the trivial solution curve is $(u, v, \lambda) \approx (0, 0, 29.1444935367)$. For $h = \frac{1}{256}$, the first discrete bifurcation point is $(U, V, \lambda^*_{1,1}) = (0, 0, 29.1442509002)$. Fig. 11 displays the bifurcation curve for various values of d_2 with $d_1 = 1$, $\alpha = 4$, $h = \frac{1}{256}$. In our numerical experiments, Methods 1–4 and the single-grid continuation method were implemented to trace the solution curve branching from the first bifurcation point $(0, 0, \lambda^*_{1,1})$, where the Bi-CGSTAB method [23] was used as the linear solver. Table 5 lists the total execution time, the total Bi-CGSTAB iterations, and the Newton iterations on the fine grid. Fig. 12 shows that the solution curves of u and v branching from $(0, 0, \lambda^*_{1,1})$ are transcritical. The contours of the solution curves of u and v at $\lambda = 26.7036738$, 30.0809857, 228.491101 are displayed in Fig. 13.

7. Conclusions

We have presented efficient two-grid discretization schemes with two-loop continuation algorithms for computing wave functions of nonlinear Schrödinger equations. Some variants of the correct procedure in the inner continuation are also proposed. Both rectangular and polar coordinates are considered in our numerical experiments. As a comparison the six-node triangular elements and the higher order Adini elements are also exploited to discretize a single NLS. Our numerical result shows that it is very promising to develop two-grid discretization schemes using the Adini elements when the domain is rectangular. The two-grid discretization schemes we discussed in this paper also can be used to compute stationary solutions of reaction–diffusion systems.

Based on the numerical results reported in Section 6, we wish to give some conclusions concerning the performance of the algorithms we proposed in Section 3.

1. The main costs of the proposed algorithms depend on: (i) how many linear systems are required to be solved on the fine grid and (ii) how close are the initial guesses to the exact solution curves. In case (ii) we can see, e.g., from Tables 1,3,4, and 5a that most initial guesses for Newton's method in Method 2 can be accepted as approximating points for the solution curve c on the fine grid.
2. Implementing Algorithm 3.1 for Examples 2–5 is only slightly cheaper than implementing the continuation algorithm on a single fine grid.
3. Finally, we conclude that it is unnecessary to perform quadratic approximations for computing stationary solutions of reaction–diffusion systems.

References

[1] J.R. Anglin, W. Ketterle, Bose–Einstein condensation of atomic gases, Nature 416 (2002) 211–218.

[2] W. Bao, Ground states and dynamics of multicomponent Bose–Einstein condensates, Multiscale Model. Simul. 2 (2004) 210–236.

[3] W. Bao, Q. Du, Computing the ground state solution of Bose–Einstein condensates by a normalized gradient flow, SIAM J. Sci. Comput. 25 (2004) 1674–1697.

[4] W. Bao, D. Jacksch, P.A. Markowich, Numerical solution of the Gross-Pitaevskii equation for Bose–Einstein condensation, J. Comput. Phys. 187 (2003) 318–342.

[5] W. Bao, S. Jin, P.A. Markowich, Numerical study of time-splitting spectral discretizations of nonlinear Schrödinger equations in the semiclassical regimes, SIAM J. Sci. Comput. 25 (2003) 27–64.

[6] S.-L. Chang, C.-S. Chien, B.-W. Jeng, Liapunov–Schmidt reduction and continuation for nonlinear Schrödinger equations, SIAM J. Sci. Comput. 29 (2007) 729–755.

[7] S.-L. Chang, C.-S. Chien, B.-W. Jeng, Computing wave functions of nonlinear Schrödinger equations: a time-independent approach, J. Comput. Phys. (2007), in press, doi:10.1016/j.jcp.2007.03.028.

[8] Y. Chen, Y. Huang, D. Yu, A two-grid method for expanded mixed finite-element solution of semilinear reaction–diffusion equations, Internat. J. Numer. Methods Eng. 57 (2003) 193–209.

[9] C.-S. Chien, M.-H. Chen, Multiple bifurcations in a reaction–diffusion problem, Comput. Math. Appl. 35 (1998) 15–39.

[10] C.-S. Chien, B.-W. Jeng, A two-grid discretization scheme for semilinear elliptic eigenvalue problems, SIAM J. Sci. Comput. 27 (2006) 1287–1304.

[11] C.-S. Chien, Z. Mei, C.-L. Shen, Numerical continuation at double bifurcation points of a reaction–diffusion problem, Internat. J. Bifurcation Chaos 8 (1998) 117–139.

[12] P. Ghosh, Exact results on the dynamics of two-component Bose–Einstein condensate, Phys. Rev. Lett. 85 (2000) 5030–5033.

[13] A. Gierer, H. Meinhardt, A theory of biological pattern formation, Kybernetik 12 (1972) 30–39.

[14] D.S. Hall, M.R. Matthews, J.R. Ensher, C.E. Wieman, E.A. Cornell, Dynamics of component separation in a binary mixture of Bose–Einstein condensates, Phys. Rev. Lett. 81 (1998) 1539–1542.

[15] H.T. Huang, Z.C. Li, Global superconvergence of Adini's elements coupled with the Trefftz method for singular problems, Eng. Anal. Boundary Elem. 27 (2003) 227–240.

[16] H.T. Huang, Z.C. Li, N. Yan, New error estimates of Adini's elements for Poisson's equation, Appl. Numer. Math. 50 (2004) 49–74.

[17] M.-C. Lai, A note on finite difference discretizations for Poisson equation on a disk, Numer. Methods Partial Differential Equations 17 (2001) 199–203.

[18] T.-C. Lin, J. Wei, Ground state of N coupled nonlinear Schrödinger equations in $\mathbf{R}^n$, $n \leqslant 3$, Comm. Math. Phys. 255 (2005) 629–653.

[19] J.D. Murray, Mathematical Biology, Springer, Berlin, 1989.

[20] C.J. Myatt, E.A. Burt, R.W. Ghrist, E.A. Cornell, C.E. Wieman, Production of two overlapping Bose–Einstein condensates by sympathetic cooling, Phys. Rev. Lett. 78 (1997) 586–589.

[21] W.-M. Ni, Qualitative properties of solutions to elliptic problems, in: M. Chipot, P. Quittner (Eds.), Handbook of Differential Equations, vol. I, Stationary Partial Differential Equations, North-Holland, Amsterdam, 2004, pp. 157–233.

[22] I. Prigogine, R. Lefever, Stability and self-organization in open systems, in: G. Nicolis, R. Lefever (Eds.), Membranes, Dissipative Structures, and Evolution, Benjamin, New York, 1974, pp. 1–16.

[23] H.A. van der Vorst, Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. 13 (1992) 631–644.

[24] L. Wu, M.B. Allen, A two-grid method for mixed finite-element solution of reaction–diffusion equations, Numer. Methods Partial Differential Equations 15 (1999) 317–332.

[25] L. Wu, M.B. Allen, Two-grid methods for mixed finite-element solution of coupled reaction–diffusion systems, Numer. Methods Partial Differential Equations 15 (1999) 589–604.