Journal of Computational Mathematics
Vol.42, No.1, 2024, 111–133.

http://www.global-sci.org/jcm
doi:10.4208/jcm.2202-m2021-0302

SEMI-IMPLICIT SPECTRAL DEFERRED CORRECTION METHODS BASED ON SECOND-ORDER TIME INTEGRATION SCHEMES FOR NONLINEAR PDES*

Ruihan Guo
School of Mathematics and Statistics, Zhengzhou University, Zhengzhou 450001, China
Email: rguo@zzu.edu.cn
Yan Xu[1)]
School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, China
Email: yxu@ustc.edu.cn

Abstract

In [20], a semi-implicit spectral deferred correction (SDC) method was proposed, which is efficient for highly nonlinear partial differential equations (PDEs). The semi-implicit SDC method in [20] is based on first-order time integration methods, which are corrected iteratively, with the order of accuracy increased by one for each additional iteration. In this paper, we will develop a class of semi-implicit SDC methods, which are based on second-order time integration methods and the order of accuracy are increased by two for each additional iteration. For spatial discretization, we employ the local discontinuous Galerkin (LDG) method to arrive at fully-discrete schemes, which are high-order accurate in both space and time. Numerical experiments are presented to demonstrate the accuracy, efficiency and robustness of the proposed semi-implicit SDC methods for solving complex nonlinear PDEs.

Mathematics subject classification: 65M60, 35L75, 35G25.
Key words: Spectral deferred correction method, Nonlinear PDEs, Local discontinuous Galerkin method, Second-order scheme.

1. Introduction

Following the method-of-lines approach, the application of the local discontinuous Galerkin (LDG) method for spatial discretization of PDE will generate a system of ODEs. In some cases, the right hand side can be written as the sum of two terms, a stiff one (F_S) and a non-stiff one (F_N)

$$\begin{cases} u_t = F_S(t, u(t)) + F_N(t, u(t)), \quad t \in [0, T], \\ u(0) = u_0. \end{cases} \tag{1.1}$$

An efficient time discretization technique to solve the above ODEs is semi-implicit methods [10, 26], which treating the non-stiff terms explicitly and the stiff terms implicitly.

However, not all ODEs containing stiff and non-stiff components appear in partitioned form (1.1), and therefore the use of standard semi-implicit schemes is not straightforward. Boscarino *et al.* [2] developed a new semi-implicit Runge-Kutta method to solve a large class of PDEs

* Received October 28, 2021 / Revised version received January 18, 2022 / Accepted February 14, 2022 / Published online January 26, 2023 /
[1)] Corresponding author

and obtained high-order accuracy. However, the Runge-Kutta method has some limitations, for example, it is much more difficult to construct for higher order accuracy. Motivated by these, we developed a semi-implicit SDC method [20] to solve ODEs without easily separating of stiff and non-stiff components.

Dutt, Greengard and Rokhlin first developed a variation on the classical defect or deferred correction methods [1,21], called the SDC method [11]. It is based on first-order time integration methods, which are corrected iteratively, with the order of accuracy increased by one for each additional iteration. Then, a semi-implicit SDC method was introduced by Minion [29] to solve equations containing both stiff and non-stiff components. Recently, the semi-implicit SDC method was generalized for solving a series of nonlinear problems [14, 19, 28, 34, 35], which all have easily separating of stiff and non-stiff components. In the semi-implicit SDC scheme, one treats the stiff components implicitly and the non-stiff components explicitly. Various numerical simulations demonstrate that the semi-implicit SDC method is effective and robust.

The original SDC methods are based on first-order time integration methods, which are corrected iteratively, with the order of accuracy increased by one for each additional iteration. In [3,4,7–9], a variant of SDC, integral deferred correction (IDC), constructed using uniform nodes and high-order Runge-Kutta integrators in both the prediction and corrections was introduced. Using a Runge-Kutta method of order r in the correction results in r more degrees of accuracy with each successive correction. It was demonstrated that the IDC methods are more efficient than SDC methods based on first-order time integration methods. Motivated by the idea, we will develop a class of semi-implicit SDC methods, which are based on second-order time integration methods and the order of accuracy are increased by two for each additional iteration. In addition, the SDC methods are efficient for a large class of PDEs, including those without easily separating stiff and non-stiff components.

To use the SDC methods, the main difficulty is to construct an efficient basic second-order scheme that is unconditionally stable. One idea is based on the second-order Crank-Nicolson/Adams-Bashforth (CN/AB) method, the other one is based on the second-order invariant energy quadratization (IEQ) approach which will result in high-order linear schemes. Many application problems, such as the convection diffusion equation, the surface diffusion of graphs, the nonlinear Schrödinger equation and the gradient flow models, can be solved by using the proposed SDC scheme coupled with high-order LDG methods.

The rest of this paper is organized as follows. In Section 2, we develop two SDC schemes, which are based on the second-order CN/AB method, and the IEQ approach respectively. In Section 3, we present some applications of the proposed SDC methods, including the convection diffusion equation, the surface diffusion of graphs, the nonlinear Schrödinger equation, the Allen-Cahn equation, the Cahn-Hilliard equation and the Cahn-Hilliard phase field model of the binary fluid-surfactant system. Numerical examples are also given to validate the proposed SDC methods. Finally, we give the concluding remarks in Section 4. In Appendix A, we give the analysis of the accuracy for the SDC method. In Appendix B, we take the nonlinear Schrödinger equation as an example to illustrate the LDG method and to prove the energy stability in the fully-discrete level.

2. SDC Schemes Based on Second-Order Time Integration Methods

In this section, we will develop a class of SDC methods, which are based on second-order time integration methods and the order of accuracy are increased by two for each additional iteration.

2.1. The SDC method based on the second-order CN/AB method

For highly nonlinear PDEs, the stiff and non-stiff components cannot be well separated. After the LDG spatial discretization for these PDEs, we can get an ODEs of the form

$$\begin{cases} u_t = F(t, u(t), u(t)), \quad t \in [0, T], \\ u(0) = u_0, \end{cases} \tag{2.1}$$

where u_0, $u(t) \in \mathbb{C}^l$, $l \in \mathbb{N}$ and $F : \mathbb{R} \times \mathbb{C}^l \times \mathbb{C}^l \to \mathbb{C}^l$. Requiring that $F \in C^1(\mathbb{R} \times \mathbb{C}^l \times \mathbb{C}^l)$ is sufficient smooth to guarantee local existence and uniqueness of the solution to (2.1). The dependence on the second argument of F is non-stiff, while the dependence on the last argument of F is stiff. Then, we briefly describe the SDC method to solve (2.1) in the following.

Integrating (2.1) from a to t, we obtain the corresponding Picard integral equation

$$u(t) = u_a + \int_a^t F(\tau, u(\tau), u(\tau))d\tau. \tag{2.2}$$

Given an initial approximation solution $u_1(t)$ to (2.1), we denote the error function by $\delta(t) = u(t) - u_1(t)$. Another measure for the quality of the approximation is defined as

$$\epsilon(t, u_1) = u_a + \int_a^t F(\tau, u_1(\tau), u_1(\tau))d\tau - u_1(t). \tag{2.3}$$

Substituting $u(t) = u_1(t) + \delta(t)$ into (2.2), we obtain

$$\delta(t) = u_a + \int_a^t F(\tau, u_1(\tau) + \delta(\tau), u_1(\tau) + \delta(\tau))d\tau - u_1(t). \tag{2.4}$$

Combining (2.3) with (2.4), we achieve

$$\delta(t) = \int_a^t F(\tau, u_1(\tau) + \delta(\tau), u_1(\tau) + \delta(\tau)) - F(\tau, u_1(\tau), u_1(\tau))d\tau + \epsilon(t, u_1), \tag{2.5}$$

which is referred to as the correction equation.

Suppose now the time interval $[0, T]$ is divided into M non-overlapping intervals by the partition $0 = t^0 < t^1 < \cdots < t^n < \cdots < t^M = T$. Let $\Delta t^n = t^{n+1} - t^n$ and u^n denotes the numerical approximation of $u(t^n)$. We do the SDC procedure in every interval $[t^n, t^{n+1}]$. Divide the time interval $[t^n, t^{n+1}]$ into P subintervals by choosing the points $t^{n,m}$ for $m = 0, \ldots, P$ such that $t^n = t^{n,0} < t^{n,1} < \cdots < t^{n,m} < \cdots < t^{n,P} = t^{n+1}$. Let $\Delta t^{n,m} = t^{n,m+1} - t^{n,m}$. The points $\{t^{n,m}\}_{m=0}^P$ can be chosen to be the Gauss-Lobatto nodes on $[t^n, t^{n+1}]$. To simplify the presentation, we take $u_1^{n,m}$ to represent the initial approximation at $t^{n,m}$ and $u_k^{n,m}$ to represent the k-th corrected approximation at $t^{n,m}$.

The initial approximate solution $u_1^{n,m}$ can be obtained by using the second-order CN/AB method, which is given by

$$u_1^{n,m+1} = u_1^{n,m} + \Delta t^{n,m} F\left(\frac{3}{2}t^{n,m} - \frac{1}{2}t^{n,m-1}, \frac{3}{2}u_1^{n,m} - \frac{1}{2}u_1^{n,m-1}, \frac{1}{2}(u_1^{n,m+1} + u_1^{n,m})\right). \tag{2.6}$$

Then we compute the corrections $\delta_k^{n,m}$ to increase the accuracy for $u_1^{n,m}$ by $u_{k+1}^{n,m} = u_k^{n,m} + \delta_k^{n,m}$.

By using the second-order CN/AB scheme to the correction Eq. (2.5), we obtain

$$\begin{aligned}\delta_k^{n,m+1} = \delta_k^{n,m} &+ \Delta t^{n,m} F\left(\frac{3}{2}t^{n,m} - \frac{1}{2}t^{n,m-1}, \frac{3}{2}(u_k^{n,m} + \delta_k^{n,m}) - \frac{1}{2}(u_k^{n,m-1} + \delta_k^{n,m-1}),\right.\\ &\left.\frac{1}{2}(u_k^{n,m+1} + \delta_k^{n,m+1} + u_k^{n,m} + \delta_k^{n,m})\right)\\ &- \Delta t^{n,m} F\left(\frac{3}{2}t^{n,m} - \frac{1}{2}t^{n,m-1}, \frac{3}{2}u_k^{n,m} - \frac{1}{2}u_k^{n,m-1}, \frac{1}{2}(u_k^{n,m+1} + u_k^{n,m})\right)\\ &+ \epsilon^{n,m+1}(u_k) - \epsilon^{n,m}(u_k). \end{aligned} \tag{2.7}$$

Denote

$$I_m^{m+1}(u_k) = \int_{t^{n,m}}^{t^{n,m+1}} F(t, u_k(t), u_k(t))dt. \tag{2.8}$$

Then by (2.3), we can derive

$$I_m^{m+1}(u_k) = \epsilon^{n,m+1}(u_k) - \epsilon^{n,m}(u_k) + u_k^{n,m+1} - u_k^{n,m}. \tag{2.9}$$

Note $u_k^{n,m+1} + \delta_k^{n,m+1} = u_{k+1}^{n,m+1}$, substituting (2.9) into (2.7) gives

$$\begin{aligned}u_{k+1}^{n,m+1} = u_{k+1}^{n,m} &+ \Delta t^{n,m} F\left(\frac{3}{2}t^{n,m} - \frac{1}{2}t^{n,m-1}, \frac{3}{2}u_{k+1}^{n,m} - \frac{1}{2}u_{k+1}^{n,m-1}, \frac{1}{2}(u_{k+1}^{n,m+1} + u_{k+1}^{n,m})\right)\\ &- \Delta t^{n,m} F\left(\frac{3}{2}t^{n,m} - \frac{1}{2}t^{n,m-1}, \frac{3}{2}u_k^{n,m} - \frac{1}{2}u_k^{n,m-1}, \frac{1}{2}(u_k^{n,m+1} + u_k^{n,m})\right)\\ &+ I_m^{m+1}(u_k). \end{aligned} \tag{2.10}$$

The integral in Eq. (2.10) is computed as follows. Approximating $F(t, u_k, u_k)$ by its Lagrange interpolation polynomials

$$F_p(t, u_k(t), u_k(t)) = \sum_{j=0}^{P} F(t^{n,j}, u_k^{n,j}, u_k^{n,j})\mathcal{L}_j^p(t), \tag{2.11}$$

and approximating $I_m^{m+1}(u_k)$ by

$$I_m^{m+1}(u_k) \approx \int_{t^{n,m}}^{t^{n,m+1}} F_p(t, u_k(t), u_k(t)) = \sum_{j=0}^{P} F(t^{n,j}, u_k^{n,j}, u_k^{n,j})c_p^{mj}, \tag{2.12}$$

where

$$c_p^{mj} = \int_{t^{n,m}}^{t^{n,m+1}} \mathcal{L}_j^p(t)dt.$$

The details related to the SDC scheme implementation are summarized in the following algorithm:

- **Step 1.** For $m = 0, \ldots, P-1$, compute $u_1^{n,m+1}$ from (2.6).
- **Step 2.** For $k = 1, \ldots, K$, $m = 0, \ldots, P-1$, compute $u_{k+1}^{n,m+1}$ by (2.10).
- **Step 3.** Update u^{n+1} by $u^{n+1} = u_{K+1}^{n,P}$, then goto the next step.

Proposition 2.1 (Local truncation error). *The local truncation error obtained with the above SDC scheme is* $\mathcal{O}(h^{\min[2K+2,2P]})$*, where* $h = \max_{n,m} \Delta t^{n,m}$.

Proof. We provide the proof in Appendix A. □

Remark 2.1. In the SDC scheme, we use the Gauss-Lobatto quadrature nodes on $[t^n, t^{n+1}]$. It is different from the IDC methods, where uniform quadrature nodes are necessary. Numerical experiments are then presented to show the high-order accuracy of our proposed SDC scheme.

Remark 2.2. The SDC scheme proposed above is based on a second-order CN/AB method. It is easy to verify the region of absolute stability for the second-order method contains the left-half plan and the second-order method is A-stable scheme [23]. While for the SDC method, the successive corrections are necessary to increase the accuracy, which will alter the the region of absolute stability. We take the fourth-order SDC scheme ($K = 1, P = 2$) as an example to study the region of absolute stability and present it in Fig. 2.1. It is observed that the fourth-order SDC scheme is not A-stable, but the region of absolute stability is still large.

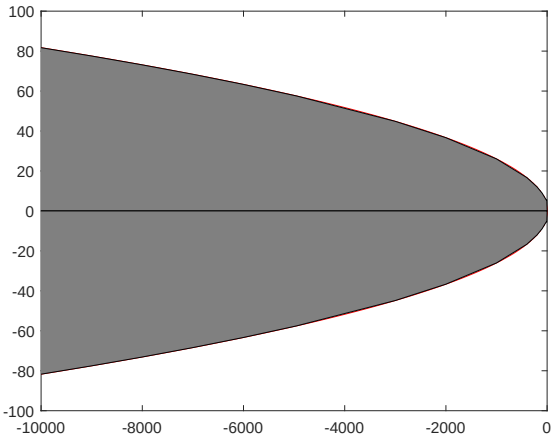


Fig. 2.1. The region of absolute stability for the SDC scheme with $K = 1$ and $P = 2$.

2.2. The SDC scheme based on the second-order IEQ approach

In this subsection, we will develop another SDC scheme, which is based on second-order linear time integration methods constructed by the IEQ approach. The IEQ approach was proposed by Yang *et al.* [15, 16, 22] to deal with general gradient flow models, obtaining linear energy stable schemes. Here, we take the gradient flow models as an example and present the details of the SDC scheme based on the IEQ approach. In addition, the SDC scheme can be applied to solve other PDEs beyond the gradient flow models, which can be found in Section 3.

We first briefly describe the IEQ approach for gradient flow models. Consider a free energy on a bounded domain Ω with dimension $d \leq 3$

$$E(\phi) = \int_\Omega \left(\frac{1}{2} \phi L \phi + N(\phi) \right) d\boldsymbol{x}, \tag{2.13}$$

where L is a linear self-adjoint operator, $N(\phi)$ is a nonlinear free energy and $\boldsymbol{x} \in \mathbb{R}^d$. A general gradient flow is given by

$$\phi_t = G\mu, \quad \mu := \frac{\delta E}{\delta \phi} = L\phi + N'(\phi), \tag{2.14}$$

where G is a negative semi-definite operator. $G = -I$ is the L^2 gradient flow, e.g., the Allen-Cahn equation; $G = \Delta$ is the H^{-1} gradient flow, e.g., the Cahn-Hilliard equation.

Introduce auxiliary variable $q = \sqrt{N(\phi) + C}$, where C is a positive constant to ensure $N(\phi) + C > 0$. Then the original gradient flow (2.14) can be rewritten as

$$\begin{cases} \phi_t = G\left(L\phi + \dfrac{q}{\sqrt{N(\phi)+C}} N'(\phi)\right), \\ q_t = \dfrac{N'(\phi)}{2\sqrt{N(\phi)+C}} \phi_t, \end{cases} \tag{2.15}$$

the reformulated free energy is given by

$$\widetilde{E} = \int_\Omega \left(\frac{1}{2}\phi L\phi + q^2\right) d\boldsymbol{x}. \tag{2.16}$$

A second-order linear scheme for the system (2.15) is

$$\begin{cases} \dfrac{\phi^{n+1} - \phi^n}{\Delta t} = G\left(L\phi^{n+\frac{1}{2}} + \dfrac{q^{n+\frac{1}{2}}}{\sqrt{N(\widetilde{\phi}^{n+\frac{1}{2}}) + C}} N'(\widetilde{\phi}^{n+\frac{1}{2}})\right), \\ \dfrac{q^{n+1} - q^n}{\Delta t} = \dfrac{N'(\widetilde{\phi}^{n+\frac{1}{2}})}{2\sqrt{N(\widetilde{\phi}^{n+\frac{1}{2}}) + C}} \dfrac{\phi^{n+1} - \phi^n}{\Delta t}, \end{cases} \tag{2.17}$$

where

$$\phi^{n+\frac{1}{2}} = \frac{1}{2}(\phi^{n+1} + \phi^n), \quad \widetilde{\phi}^{n+\frac{1}{2}} = \frac{3}{2}\phi^n - \frac{1}{2}\phi^{n-1}.$$

Based on the second-order scheme (2.17), we can achieve the corresponding SDC method. Here, we omit the details of the derivation and only present the SDC algorithm. To simplify the presentation, we introduce the following notations:

$$\widetilde{(\cdot)}^{n,m+\frac{1}{2}} = \frac{3}{2}(\cdot)^{n,m} - \frac{1}{2}(\cdot)^{n,m-1}, \quad (\cdot)^{n,m+\frac{1}{2}} = \frac{1}{2}((\cdot)^{n,m+1} + (\cdot)^{n,m})$$

in the SDC algorithm.

Compute the initial approximation: $\phi_1^{n,0} = \phi^n, \quad q_1^{n,0} = q^n.$
Use the second-order linear scheme (2.17) to compute approximate solution ϕ_1, q_1 at the nodes $\{t^{n,m}\}_{m=1}^P$, i.e.
For $m = 0, \ldots, P-1$,

$$\phi_1^{n,m+1} = \phi_1^{n,m} + \Delta t^{n,m} G\left(L\phi_1^{n,m+\frac{1}{2}} + \frac{q_1^{n,m+\frac{1}{2}}}{\sqrt{N(\widetilde{\phi}_1^{n,m+\frac{1}{2}}) + C}} N'(\widetilde{\phi}_1^{n,m+\frac{1}{2}})\right), \tag{2.18}$$

$$q_1^{n,m+1} = q_1^{n,m} + \frac{N'(\widetilde{\phi}_1^{n,m+\frac{1}{2}})}{2\sqrt{N(\widetilde{\phi}_1^{n,m+\frac{1}{2}}) + C}} (\phi_1^{n,m+1} - \phi_1^{n,m}). \tag{2.19}$$

Compute successive corrections: For $k = 1, \ldots, K$,

$$\phi_{k+1}^{n,0} = \phi^n, \quad q_{k+1}^{n,0} = q^n.$$

For $m = 0, \ldots, P-1$,

$$\phi_{k+1}^{n,m+1} = \phi_{k+1}^{n,m} + \Delta t^{n,m} G\left(L\phi_{k+1}^{n,m+\frac{1}{2}} + \frac{\widetilde{q}_{k+1}^{n,m+\frac{1}{2}}}{\sqrt{N(\widetilde{\phi}_{k+1}^{n,m+\frac{1}{2}}) + C}} N'(\widetilde{\phi}_{k+1}^{n,m+\frac{1}{2}})\right) \tag{2.20}$$

$$- \Delta t^{n,m} G\left(L\phi_{k}^{n,m+\frac{1}{2}} + \frac{\widetilde{q}_{k}^{n,m+\frac{1}{2}}}{\sqrt{N(\widetilde{\phi}_{k}^{n,m+\frac{1}{2}}) + C}} N'(\widetilde{\phi}_{k}^{n,m+\frac{1}{2}})\right) + I_m^{m+1}(F(t, \phi_k, q_k)),$$

$$q_{k+1}^{n,m+1} = \sqrt{N(\phi_{k+1}^{n,m+1}) + C}, \tag{2.21}$$

where

$$F(t, \phi_k, q_k) = G\left(L\phi_k + \frac{q_k}{\sqrt{N(\phi_k) + C}} N'(\phi_k)\right),$$

and $I_m^{m+1}(F(t, \phi_k, q_k))$ is the integral of the P-th degree interpolating polynomial on the $P+1$ points $(t^{n,m}, F(t^{n,m}, \phi_k^{n,m}, q_k^{n,m}))_{m=0}^{P}$ over the subinterval $[t^{n,m}, t^{n,m+1}]$.

Finally we have $\phi^{n+1} = \phi_{K+1}^{n,P}$ and $q^{n+1} = q_{K+1}^{n,P}$.

Remark 2.3. Motivated by the idea in [11], the SDC methods proposed above can be extended using high-order time integration schemes in the initial approximation and successive correction. However, it is not easy to construct higher order stable time integration schemes as the building block. Thus, in this paper, we only present SDC methods based on second-order schemes.

3. Applications and Numerical Validation

In this section, we apply the proposed SDC schemes to several PDEs, including the convection diffusion equation, the surface diffusion of graphs, the nonlinear Schrödinger equation and the phase field models, to demonstrate the efficiency and accuracy of the SDC methods. In all examples, we assume periodic boundary conditions and employ the LDG method with piecewise $\mathcal{P}^k$ polynomial basis for spatial discretization. To show that the SDC method is high-order accurate, we choose $K = 1$ and $P = 2$ in the SDC method, which has fourth-order of accuracy. At each time step, we solve the linear algebraic equations by multigrid solver [18, 33].

3.1. The convection diffusion equations

We consider the convection diffusion equation in 1D

$$u_t = u_{xx} - u_x, \quad x \in [0, 6\pi]. \tag{3.1}$$

A second-order CN/AB scheme to solve the equation is

$$\frac{u^{n+1} - u^n}{\Delta t} = \frac{1}{2}(u_{xx}^{n+1} + u_{xx}^n) - \left(\frac{3}{2}u_x^n - \frac{1}{2}u_x^{n-1}\right). \tag{3.2}$$

Example 3.1. In this example, we test the accuracy of the proposed SDC method (2.6) and (2.10), and the performance of various SDC methods and IDC methods. The exact solution of Eq. (3.1) is given by

$$u(x, t) = e^{-t}\sin(x - t).$$

The L^2 errors and the numerical orders of accuracy are presented in Table 3.1, which shows the expected accuracy of $\min(2K+2, 2P)$.

In Table 3.2, SDC-CN/AB denotes the SDC method (2.6) and (2.10) constructed using three Gauss-Lobatto nodes and the second-order CN/AB scheme (3.2) for the initial and correction steps. SDC-IMEX denotes the SDC method constructed using three Gauss-Lobatto nodes and the first order IMEX scheme ($u^{n+1} = u^n + \Delta t(u_{xx}^{n+1} - u_x^n)$) for the initial and correction steps. IDC-ARK2 (IDC-IMEX) denote the IDC method constructed using four uniformly distributed nodes and the ARK2 [9] (IMEX) integrators for the prediction and correction loops. Fourth-order of accuracy is observed for all cases as expected.

For the sake of comparison of efficiency, we present the CPU time of various SDC and IDC methods in Table 3.3. We notice that the SDC-CN/AB method takes least CPU time, because it is based on second-order schemes, and the order of accuracy increased by two for each additional iteration.

Table 3.1: Accuracy test for the one-dimensional convection diffusion equation at time $T = 1$. $\Delta t = 0.05\Delta x$ and $\Delta x = 6\pi/N$.

-	$K=0,\ P=2$		$K=1,\ P=2$		$K=2,\ P=4$	
N	L^2 error	order	L^2 error	order	L^2 error	order
8	3.65E-01	–	8.58E-03	–	9.63E-05	–
16	8.37E-02	2.12	5.93E-04	3.85	1.40E-06	6.10
32	2.16E-02	1.95	3.88E-05	3.93	2.08E-08	6.06
64	5.64E-03	1.94	2.48E-06	3.97	3.17E-010	6.04

Table 3.2: Accuracy test of various SDC methods and IDC methods for the one-dimensional convection diffusion equation at time $T = 0.5$. $\Delta t = 0.05\Delta x$ and $\Delta x = 6\pi/N$.

-	SDC-CN/AB		SDC-IMEX		IDC-ARK2		IDC-IMEX	
N	L^2 error	order	L^2 error	order	L^2 error	order	L^2 error	order
8	1.41E-02	–	1.41E-02	–	1.41E-02	–	4.10E-06	3.97
16	9.77E-04	3.85	9.77E-04	3.85	9.77E-04	3.85	9.77E-04	3.85
32	6.41E-05	3.93	6.41E-05	3.93	6.41E-05	3.93	6.41E-05	3.93
64	4.10E-06	3.97	4.10E-06	3.97	3.99E-06	4.01	4.10E-06	3.97

Table 3.3: Total CPU time using various SDC methods and IDC methods for the one-dimensional convection diffusion equation at time $T = 0.5$, $N = 64$.

-	SDC-CN/AB	SDC-IMEX	IDC-ARK2	IDC-IMEX
CPU time	18.89	32.69	28.25	39.68

3.2. The surface diffusion of graphs

We consider the surface diffusion of graphs

$$u_t + \nabla \cdot \left(Q \left(I - \frac{\nabla u \otimes \nabla u}{Q^2} \right) \nabla \left(\nabla \cdot \left(\frac{\nabla u}{Q} \right) \right) \right) = 0, \tag{3.3}$$

where Q is the area element $Q = \sqrt{1+|\nabla u|^2}$, which is a highly nonlinear PDE, and the stiff and non-stiff components cannot be separated. Thus, it is difficult to construct high-order semi-implicit schemes.

Treating the nonlinear part explicitly and the linear part implicitly and a second-order scheme to solve Eq. (3.3) is

$$\frac{u^{n+1}-u^n}{\Delta t} = -\nabla\cdot\left(\widetilde{Q}^{n+\frac{1}{2}}\left(I-\frac{\nabla\widetilde{u}^{n+\frac{1}{2}}\otimes\nabla\widetilde{u}^{n+\frac{1}{2}}}{(\widetilde{Q}^{n+\frac{1}{2}})^2}\right)\nabla\left(\nabla\cdot\left(\frac{\nabla u^{n+\frac{1}{2}}}{\widetilde{Q}^{n+\frac{1}{2}}}\right)\right)\right), \tag{3.4}$$

where

$$\widetilde{Q}^{n+\frac{1}{2}} = \sqrt{1+|\nabla\widetilde{u}^{n+\frac{1}{2}}|^2}.$$

Based on the second-order scheme (3.4), we can employ the SDC scheme (2.6) and (2.10) for the surface diffusion of graphs.

Example 3.2. In this example, we consider the accuracy test for surface diffusion of graphs (3.3). We test our method taking the exact solution

$$u(x,y,t) = 0.05e^{-2t}\sin(x+y)$$

for Eq. (3.3) with a source term.

The L^2 and L^∞ errors and the numerical orders of accuracy at time $T=0.5$ are obtained in Table 3.4, which shows up to fourth-order accuracy in both time and space.

Table 3.4: Accuracy test for the surface diffusion of graphs when using $\mathcal{P}^k$ approximation on a uniform mesh with N cells at time $T=0.5$. $\Delta t = 0.1\Delta x$ and $\Delta x = 2\pi/N$.

	N	L^2 error	order	L^∞ error	order
$\mathcal{P}^1$	8	5.84E-03	–	5.16E-03	–
	16	1.49E-03	1.97	1.33E-03	1.95
	32	3.76E-04	1.99	3.37E-04	1.98
	64	9.42E-05	2.00	8.47E-05	2.00
$\mathcal{P}^2$	8	8.43E-04	–	9.57E-04	–
	16	1.06E-04	2.98	1.21E-04	2.98
	32	1.34E-05	2.99	1.52E-05	2.99
	64	1.67E-06	3.00	1.91E-06	3.00
$\mathcal{P}^3$	8	8.49E-05	–	1.08E-04	–
	16	5.41E-06	3.97	7.03E-06	3.95
	32	3.42E-07	3.98	4.49E-07	3.97
	64	2.25E-08	3.92	3.17E-08	3.83

3.3. The Schrödinger equations

In this subsection, we present an application of the SDC scheme proposed in Subsection 2.2 to solve the nonlinear Schrödinger equation

$$\mathrm{i}u_t + \Delta u + (v(\boldsymbol{x}) + \beta|u|^2)u = 0, \tag{3.5}$$

where $u(\boldsymbol{x},t)$ is a complex function, $v(\boldsymbol{x})$ ia an arbitrary real function, $\beta>0$ is a real constant. The Schrödinger equation preserve the following energy conservation law:

$$\frac{d}{dt}E=0, \tag{3.6}$$

where

$$E=\int_\Omega\left(-\frac{1}{2}|\nabla u|^2+\frac{1}{2}v(\boldsymbol{x})|u|^2+\frac{\beta}{4}|u|^4\right)d\boldsymbol{x}. \tag{3.7}$$

To develop a second-order linear scheme for Eq. (3.5), we introduce an auxiliary function

$$W=\sqrt{\frac{1}{2}v(\boldsymbol{x})|u|^2+\frac{\beta}{4}|u|^4+C}.$$

Then the Schrödinger equation (3.5) can be rewritten as

$$\begin{cases}\mathrm{i}u_t+\Delta u+HW=0,\\ W_t=\dfrac{1}{2}Hu_t,\end{cases} \tag{3.8}$$

where

$$H=\frac{(v(\boldsymbol{x})+\beta|u|^2)u}{\sqrt{\frac{1}{2}v(\boldsymbol{x})|u|^2+\frac{\beta}{4}|u|^4+C}}.$$

In reformulated Eq. (3.8), the reformulated free energy is given by

$$\widetilde{E}=\int_\Omega\left(-\frac{1}{2}|\nabla u|^2+|W|^2\right)d\boldsymbol{x}. \tag{3.9}$$

Scheme for the Schrödinger equation: A second-order linear scheme to solve the Schrödinger equation (3.8) is

$$\mathrm{i}\frac{u^{n+1}-u^n}{\Delta t}+\Delta u^{n+\frac{1}{2}}+\widetilde{H}^{n+\frac{1}{2}}W^{n+\frac{1}{2}}=0, \tag{3.10}$$

$$\frac{W^{n+1}-W^n}{\Delta t}=\frac{1}{2}\widetilde{H}^{n+\frac{1}{2}}\frac{u^{n+1}-u^n}{\Delta t}, \tag{3.11}$$

where

$$\widetilde{H}^{n+\frac{1}{2}}=\frac{(v(\boldsymbol{x})+\beta|\widetilde{u}^{n+\frac{1}{2}}|^2)\widetilde{u}^{n+\frac{1}{2}}}{\sqrt{\frac{1}{2}v(\boldsymbol{x})|\widetilde{u}^{n+\frac{1}{2}}|^2+\frac{\beta}{4}|\widetilde{u}^{n+\frac{1}{2}}|^4+C}}.$$

Proposition 3.1. *For the Schrödinger equation* (3.8)*, the second-order linear scheme* (3.10)*,* (3.11) *is unconditionally energy stable, i.e.*

$$\widetilde{E}(u^{n+1},W^{n+1})=\widetilde{E}(u^n,W^n), \tag{3.12}$$

where $\widetilde{E}$ *is defined by* (3.9)*.*

Remark 3.3. To make the order of accuracy in space compatible with high-order accuracy in time, we employ the local discontinuous Galerkin (LDG) method [17, 36, 37] for spatial discretization. In Appendix B, we take the Schrödinger equation as an example to illustrate the LDG method and the energy stability in the fully-discrete level.

Based on the second-order scheme (3.10)-(3.11), we can employ the SDC scheme (2.18)-(2.21) proposed in subsection 2.2 for the Schrödinger equation.

Example 3.4. Here, we consider the Schrödinger equation (3.5) in the domain $[0, 2\pi]^2$. The parameters are chosen as $v(\boldsymbol{x}) = 0$, $\beta = 1$, $C = 0.1$. The exact solution is

$$u(x, y, t) = e^{i(x+y-t)}. \tag{3.13}$$

We test the convergence rate of the proposed SDC scheme at the final time $T = 0.5$. The L^2 and L^∞ errors of the real part and the imaginary part are summarized in Table 3.5, which all shows up to fourth-order accuracy in both time and space.

The energy evolution using the second order linear scheme (3.10)-(3.11) is presented in Fig. 3.1. We can see the energy is conservative, which is consistent with the theoretical result presented in Proposition B.1.

Table 3.5: Accuracy test for the Schrödinger equation when using $\mathcal{P}^k$ approximation on a uniform mesh with N cells at time $T = 0.5$. $\Delta t = 0.1\Delta x$ and $\Delta x = 2\pi/N$.

		Real part				Imaginary part			
	N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
$\mathcal{P}^1$	8	4.22E-01	–	3.20E-01	–	4.22E-01	–	3.20E-01	–
	16	1.35E-01	1.65	8.13E-02	1.98	1.35E-01	1.65	8.13E-02	1.98
	32	3.85E-02	1.81	2.68E-02	1.60	3.85E-02	1.81	2.68E-02	1.60
	64	8.79E-03	2.13	6.98E-03	1.94	8.79E-03	2.13	6.98E-03	1.94
$\mathcal{P}^2$	8	3.75E-02	–	3.95E-02	–	3.75E-02	–	3.95E-02	–
	16	1.10E-02	1.77	1.08E-02	1.86	1.10E-02	1.77	1.08E-02	1.86
	32	1.31E-03	3.06	1.45E-03	2.90	1.31E-03	3.06	1.45E-03	2.90
	64	1.21E-04	3.44	1.10E-04	3.71	1.21E-04	3.44	1.10E-04	3.71
$\mathcal{P}^3$	8	7.08E-03	–	9.27E-03	–	7.08E-03	–	9.27E-03	–
	16	5.84E-04	3.60	5.60E-04	4.04	5.84E-04	3.60	5.60E-04	4.04
	32	3.73E-05	3.97	3.90E-05	3.84	3.73E-05	3.97	3.90E-05	3.84
	64	2.84E-06	3.72	2.41E-06	4.01	2.84E-06	3.72	2.41E-06	4.01

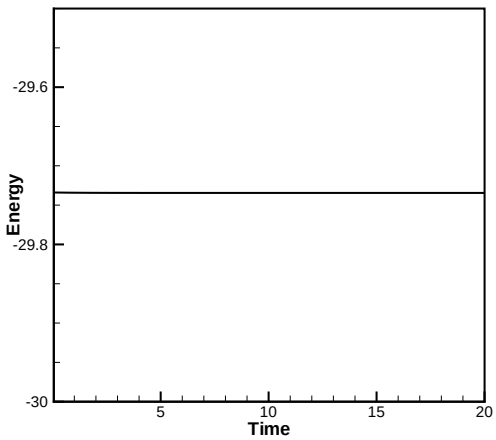


Fig. 3.1. Energy evolution using the second order scheme (3.10)-(3.11) for the Schrödinger equation.

3.4. The phase field models

For gradient flows, numerous efficient and energy stable numerical schemes have been developed recently, including the convex splitting method [6, 12], the stabilized semi-implicit method [5, 13, 24], the exponential time differencing (ETD) method [25, 27], the IEQ approach [38–40] and the scalar auxiliary variable (SAV) approach [30, 31]. For a detailed description about efficient numerical methods for phase field models, we refer readers to the review paper [32].

In this subsection, we will consider three typical phase field models as application examples of SDC scheme based on IEQ approach, including the Allen-Cahn equation, the Cahn-Hilliard equation and the highly nonlinear coupled Cahn-Hilliard phase field model of the binary fluid-surfactant system.

3.4.1. The Allen-Cahn equation

The Allen-Cahn equation

$$\phi_t = b(\phi)(\varepsilon^2\Delta\phi - F'(\phi)), \tag{3.14}$$

can be viewed as the gradient flow of the energy function

$$E_{AC} = \int_\Omega \left(\frac{\varepsilon^2}{2}|\nabla\phi|^2 + F(\phi)\right) d\boldsymbol{x}, \quad F(\phi) = \frac{1}{4}(\phi^2-1)^2. \tag{3.15}$$

Introduce auxiliary variable $U = \phi^2 - 1$, then the Allen-Cahn equation (3.14) can be rewritten as

$$\begin{cases} \phi_t = b(\phi)(\varepsilon^2\Delta\phi - \phi U), \\ U_t = 2\phi\phi_t. \end{cases} \tag{3.16}$$

In reformulated Eq. (3.16), the reformulated free energy are given by

$$\widetilde{E}_{AC} = \int_\Omega \left(\frac{\varepsilon^2}{2}|\nabla\phi|^2 + \frac{1}{4}U^2\right) d\boldsymbol{x}, \tag{3.17}$$

which is equal to the original free energy E_{AC} in the continuous level.

Scheme for the Allen-Cahn equation: A second-order linear scheme to solve the Allen-Cahn equation (3.16) is

$$\begin{cases} \dfrac{\phi^{n+1}-\phi^n}{\Delta t} = b(\widetilde{\phi}^{n+\frac{1}{2}})(\varepsilon^2\Delta\phi^{n+\frac{1}{2}} - \widetilde{\phi}^{n+\frac{1}{2}}U^{n+\frac{1}{2}}), & (3.18) \\ \dfrac{U^{n+1}-U^n}{\Delta t} = 2\widetilde{\phi}^{n+\frac{1}{2}}\dfrac{\phi^{n+1}-\phi^n}{\Delta t}. & (3.19) \end{cases}$$

Proposition 3.2. *For the Allen-Cahn equation* (3.16)*, the second-order linear scheme* (3.18)–(3.19) *is unconditionally energy stable, i.e.*

$$\widetilde{E}_{AC}(\phi^{n+1}, U^{n+1}) \le \widetilde{E}_{AC}(\phi^n, U^n), \tag{3.20}$$

where $\widetilde{E}_{AC}$ *is defined by* (3.17).

Based on the energy stable second-order scheme, we can employ the SDC method (2.18)-(2.21) to solve the Allen-Cahn equation.

Example 3.5. We test the proposed numerical schemes for solving the Allen-Cahn equation (3.14). To verify the convergence rate, we choose the suitable forcing function so that the exact solution is given by

$$\phi(x, y, t) = e^{-2t} \sin(x) \sin(y).$$

The computational domain is set as $\Omega = [0, 2\pi]^2$. The parameters are set as $b(\phi) = 1 - \phi^2$ and $\varepsilon = 1$.

The L^2 and L^∞ errors and the numerical orders of accuracy at $T = 0.5$ are summarized in Table 3.6, which shows the expected accuracy.

Table 3.6: Accuracy test for the Allen-Cahn equation when using $\mathcal{P}^k$ approximation on a uniform mesh with N cells at time $T = 0.5$. $\Delta t = 0.1\Delta x$ and $\Delta x = 2\pi/N$.

	N	L^2 error	order	L^∞ error	order
$\mathcal{P}^1$	8	8.45E-02	–	5.80E-02	–
	16	2.12E-02	1.99	1.52E-02	1.93
	32	5.33E-03	2.00	3.86E-03	1.98
	64	1.33E-03	2.00	9.71E-04	1.99
$\mathcal{P}^2$	8	1.09E-02	–	1.01E-02	–
	16	1.35E-03	3.01	1.25E-03	3.02
	32	1.69E-04	3.00	1.54E-04	3.01
	64	2.11E-05	3.00	1.92E-05	3.01
$\mathcal{P}^3$	8	1.07E-03	–	1.27E-03	–
	16	6.73E-05	4.00	8.71E-05	3.87
	32	4.21E-06	4.00	5.57E-06	3.96
	64	2.63E-07	4.00	3.50E-07	3.99

3.4.2. The Cahn-Hilliard equation

The Cahn-Hilliard equation

$$\phi_t = \nabla \cdot (b(\phi)\nabla(-\varepsilon^2 \Delta\phi + F'(\phi))) \tag{3.21}$$

can be viewed as the gradient flow of the energy function

$$E_{CH} = \int_\Omega \left(\frac{\varepsilon^2}{2} |\nabla\phi|^2 + F(\phi) \right) d\boldsymbol{x}, \quad F(\phi) = \frac{1}{4}(\phi^2 - 1)^2. \tag{3.22}$$

To develop the IEQ scheme, we introduce the auxiliary variable $U = \phi^2 - 1$, then the Cahn-Hilliard equation (3.21) can be rewritten as

$$\begin{cases} \phi_t = \nabla \cdot (b(\phi)\nabla(-\varepsilon^2 \Delta\phi + \phi U)), \\ U_t = 2\phi\phi_t. \end{cases} \tag{3.23}$$

Scheme for the Cahn-Hilliard equation: A second-order linear scheme to solve the Cahn-Hilliard equation (3.23) is

$$\begin{cases} \dfrac{\phi^{n+1} - \phi^n}{\Delta t} = \nabla \cdot (b(\widetilde{\phi}^{n+\frac{1}{2}})\nabla(-\varepsilon^2 \Delta\phi^{n+\frac{1}{2}} + \widetilde{\phi}^{n+\frac{1}{2}} U^{n+\frac{1}{2}})), & (3.24) \\ \dfrac{U^{n+1} - U^n}{\Delta t} = 2\widetilde{\phi}^{n+\frac{1}{2}} \dfrac{\phi^{n+1} - \phi^n}{\Delta t}. & (3.25) \end{cases}$$

Proposition 3.3. *For the Cahn-Hilliard equation* (3.23)*, the second-order linear scheme* (3.24)-(3.25) *is unconditionally energy stable, i.e.*

$$\widetilde{E}_{CH}(\phi^{n+1}, U^{n+1}) \leq \widetilde{E}_{CH}(\phi^n, U^n), \tag{3.26}$$

where $\widetilde{E}_{CH}$ *is defined by*

$$\widetilde{E}_{CH} = \int_\Omega \left(\frac{\varepsilon^2}{2} |\nabla \phi|^2 + \frac{1}{4} U^2 \right) d\boldsymbol{x}. \tag{3.27}$$

Based on the energy stable second-order scheme, we then employ the SDC method (2.18)-(2.21) to solve the Cahn-Hilliard equation.

Example 3.6. We consider the Cahn-Hilliard equation (3.21) with $b(\phi) = 1 - \phi^2$ and $\varepsilon = 1$. The initial condition is

$$\phi_0(x, y) = \sin(x)\sin(y), \quad 0 \leq x, y \leq 2\pi.$$

We first test the accuracy, and the L^2 and L^∞ errors and the numerical orders of accuracy at $T = 0.5$ are presented in Table 3.7.

Table 3.7: Accuracy test for the Cahn-Hilliard equation when using $\mathcal{P}^k$ approximation on a uniform mesh with N cells at time $T = 0.5$. $\Delta t = 0.1\Delta x$ and $\Delta x = 2\pi/N$.

	N	L^2 error	order	L^∞ error	order
$\mathcal{P}^1$	8	8.42E-02	–	5.77E-02	–
	16	2.12E-02	1.99	1.51E-02	1.93
	32	5.33E-03	2.00	3.85E-03	1.98
	64	1.33E-03	2.00	9.66E-04	1.99
$\mathcal{P}^2$	8	1.19E-02	–	9.97E-03	–
	16	1.40E-03	3.09	1.24E-03	3.00
	32	1.71E-04	3.03	1.54E-04	3.01
	64	2.12E-05	3.00	1.92E-05	3.00
$\mathcal{P}^3$	8	1.12E-03	–	1.44E-03	–
	16	7.10E-05	3.98	1.12E-04	3.68
	32	4.32E-06	4.03	7.35E-06	3.94
	64	2.92E-07	3.89	4.74E-07	3.95

Next, we compare the second-order scheme (3.24)-(3.25) and the fourth-order SDC scheme for solving the Cahn-Hilliard equation. The parameters are $b(\phi) = 1$ and $\varepsilon = 0.1$. We use the following initial condition:

$$\phi_0(x, y) = \text{rand}(x, y),$$

where $\text{rand}(x, y)$ is a random number satisfying $-1 \leq \text{rand}(x, y) \leq 1$.

The energy evolution using different temporal discretization methods are presented in Fig. 3.2. We take the numerical solution using the second-order scheme (3.24)-(3.25) with $\Delta t = 10^{-6}$ as the reference solution. We observe that for the second-order scheme, it is coincident with the reference energy evolution with time step $\Delta t = 10^{-4}$. For the fourth-order SDC scheme, it is coincident with the reference energy evolution with time step $\Delta t = 10^{-2}$, which is 10^2 of the one with the second-order scheme.

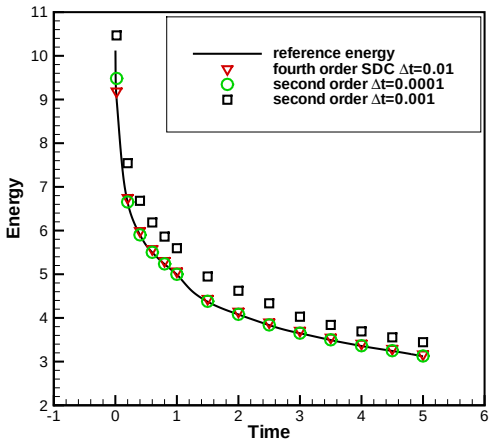


Fig. 3.2. Energy evolution using the second-order scheme (3.24)-(3.25) and the fourth-order SDC scheme with various time steps for the Cahn-Hilliard equation.

Thus, we use the fourth-order SDC scheme to show the long time simulation of spinodal decomposition. The parameters are $b(\phi) = 0.01$ and $\varepsilon = 0.1$. We use the following initial condition:

$$\phi_0(x, y) = 0.1 + 0.01\text{rand}(x, y).$$

The computational parameters are $\Delta x = 2\pi/N$ with $N = 64$ and the piecewise $\mathcal{P}^2$ approximation; the time step $\Delta t = 0.1\Delta x$. Fig. 3.3 presents the numerical results at different times, we can see the fourth-order SDC methods can capture the coarsening dynamics accurately with a larger time step of $\Delta t = 0.1\Delta x$.

3.4.3. The binary fluid-surfactant system

We consider in this subsection the Cahn-Hilliard phase field model of the binary fluid-surfactant system [41]

$$\begin{cases} \phi_t = M_\phi \Delta \mu_\phi, & (3.28) \\ \mu_\phi = -\Delta\phi + \alpha\Delta^2\phi + \dfrac{1}{\epsilon^2}\phi(\phi^2 - 1) + 2\theta\nabla\cdot(\rho\nabla\phi), & (3.29) \\ \rho_t = M_\rho \Delta \mu_\rho, & (3.30) \\ \mu_\rho = -\beta\Delta\rho + \dfrac{1}{\eta^2}\rho(\rho - \rho_s)\left(\rho - \dfrac{\rho_s}{2}\right) - \theta|\nabla\phi|^2, & (3.31) \end{cases}$$

where $\alpha, \beta, \epsilon, \eta, \rho_s, \theta$ are all positive parameters. The system (3.28)-(3.31) preserves the following PDE energy law:

$$\frac{d}{dt}E(\phi, \rho) \le 0, \tag{3.32}$$

where the free energy is given as

$$E(\phi, \rho) = \int_\Omega \left(\frac{1}{2}|\nabla\phi|^2 + \frac{\alpha}{2}(\Delta\phi)^2 + \frac{1}{4\epsilon^2}(\phi^2 - 1)^2 + \frac{\beta}{2}|\nabla\rho|^2 + \frac{1}{4\eta^2}\rho^2(\rho - \rho_s)^2 - \theta\rho|\nabla\phi|^2 \right) d\boldsymbol{x}. \tag{3.33}$$

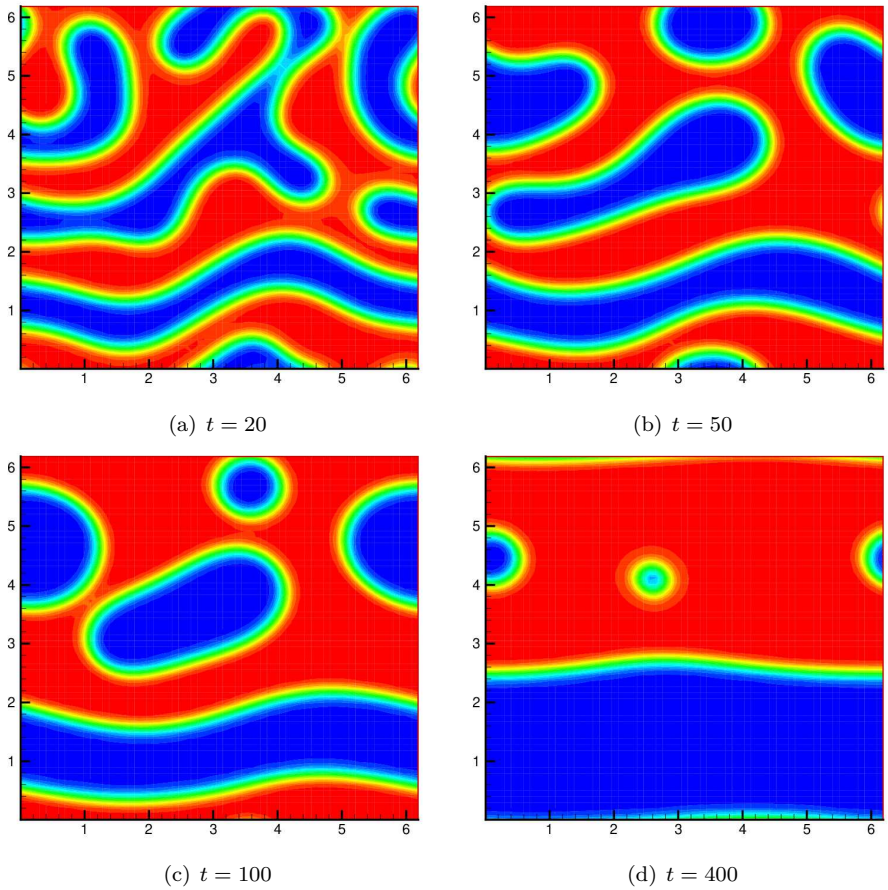
Yang [41] developed a second-order linear scheme by introducing two auxiliary functions

$$U = \phi^2 - 1, \quad V = \rho(\rho - \rho_s),$$

and rewriting the binary fluid-surfactant system as

$$\begin{cases} \phi_t = M_\phi \Delta \mu_\phi, & (3.34) \\ \mu_\phi = -\Delta\phi + \alpha\Delta^2\phi + \dfrac{1}{\epsilon^2} HU + 2\theta\nabla\cdot(\rho\nabla\phi), & (3.35) \\ \rho_t = M_\rho \Delta \mu_\rho, & (3.36) \\ \mu_\rho = -\beta\Delta\rho + \dfrac{1}{\eta^2} GV - \theta|\nabla\phi|^2, & (3.37) \\ U_t = 2H\phi_t, & (3.38) \\ V_t = 2G\rho_t, & (3.39) \end{cases}$$

where $H(\phi) = \phi$, $G(\rho) = \rho - \dfrac{\rho_s}{2}$.



(a) $t = 20$ (b) $t = 50$

(c) $t = 100$ (d) $t = 400$

Fig. 3.3. Numerical results of the Cahn-Hilliard equation using the fourth-order SDC scheme with time step $\Delta t = 0.1\Delta x$.

Scheme for the binary fluid-surfactant system: A second-order, linear and decoupled scheme to solve the binary fluid-surfactant system (3.34)-(3.39) is given as follows:

Step 1. Update ρ^{n+1} and V^{n+1} as follows:

$$\begin{cases} \dfrac{\rho^{n+1}-\rho^n}{\Delta t} = M_\rho \Delta \left(-\beta \Delta \rho^{n+\frac{1}{2}} + \dfrac{1}{\eta^2} G(\widetilde{\rho}^{n+\frac{1}{2}}) V^{n+\frac{1}{2}} - \theta |\nabla \widetilde{\phi}^{n+\frac{1}{2}}|^2 \right), & (3.40) \\ \dfrac{V^{n+1}-V^n}{\Delta t} = 2G(\widetilde{\rho}^{n+\frac{1}{2}}) \dfrac{\rho^{n+1}-\rho^n}{\Delta t}. & (3.41) \end{cases}$$

Step 2. Update ϕ^{n+1} and U^{n+1} as follows:

$$\begin{cases} \dfrac{\phi^{n+1}-\phi^n}{\Delta t} = M_\phi \Delta \Big(-\Delta \phi^{n+\frac{1}{2}} + \alpha \Delta^2 \phi^{n+\frac{1}{2}} + \dfrac{1}{\epsilon^2} H(\widetilde{\phi}^{n+\frac{1}{2}}) U^{n+\frac{1}{2}} \\ \qquad\qquad +2\theta \nabla \cdot (\rho^{n+\frac{1}{2}} \nabla \phi^{n+\frac{1}{2}}) \Big), & (3.42) \\ \dfrac{U^{n+1}-U^n}{\Delta t} = 2H(\widetilde{\phi}^{n+\frac{1}{2}}) \dfrac{\phi^{n+1}-\phi^n}{\Delta t}. & (3.43) \end{cases}$$

Based on the above second-order scheme, we then employ the SDC scheme to obtain high order temporal accuracy.

Example 3.7. In this example, we consider the Cahn-Hilliard phase field model of the binary fluid-surfactant system (3.28)-(3.31). For the accuracy test, we choose a suitable source term such that the exact solution is taken as

$$\begin{cases} u(x,y,t) = 0.5 + 0.1e^{-2t}\sin(x+y), \\ \rho(x,y,t) = 0.3 + 0.1e^{-2t}\sin(x+y). \end{cases} \tag{3.44}$$

The parameters are set below

$$\begin{aligned} &M_\rho = 2.5e-3, \quad \alpha = 0.01, \quad \epsilon = 0.05, \quad \theta = 0.03, \\ &M_\phi = 2.5e-3, \quad \beta = 1, \qquad \eta = 0.08, \quad \rho_s = 1. \end{aligned}$$

The L^2 and L^∞ errors and the numerical orders of accuracy using the fourth-order SDC scheme at $T = 0.2$ are presented in Table 3.8. The error is expected to be at the order of $\min\{\mathcal{O}(\Delta t^4), \mathcal{O}(\Delta x^{k+1})\}$ for $\mathcal{P}^k$ approximation.

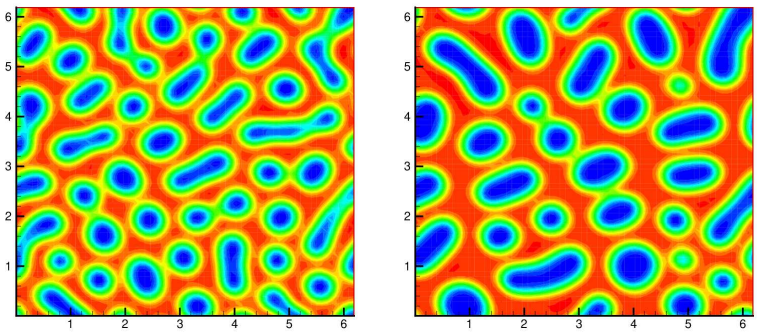
Table 3.8: Accuracy test for the Cahn-Hilliard phase field model of the binary fluid-surfactant system when using $\mathcal{P}^k$ approximation on a uniform mesh with N cells at time $T = 0.2$. $\Delta t = 0.1\Delta x$ and $\Delta x = 2\pi/N$.

		ϕ				ρ			
	N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
$\mathcal{P}^1$	8	1.83E-02	–	1.59E-02	–	1.77E-02	–	1.37E-02	–
	16	5.39E-03	1.77	4.84E-03	1.72	5.52E-03	1.68	4.92E-03	1.48
	32	1.37E-03	1.97	1.23E-03	1.97	1.37E-03	2.00	1.23E-03	2.00
	64	3.43E-04	2.00	3.08E-04	2.00	3.43E-04	2.00	3.08E-04	2.00
$\mathcal{P}^2$	8	3.26E-03	–	3.47E-03	–	2.44E-03	–	2.46E-03	–
	16	4.18E-04	2.96	5.11E-04	2.76	3.97E-04	2.62	4.44E-04	2.47
	32	4.90E-05	3.09	5.62E-05	3.18	4.90E-05	3.02	5.58E-05	2.99
	64	6.13E-06	3.00	7.01E-06	3.00	6.14E-06	3.00	6.99E-06	3.00
$\mathcal{P}^3$	8	4.69E-04	–	4.57E-04	–	2.52E-04	–	3.12E-04	–
	16	2.71E-05	4.10	4.00E-05	3.51	2.03E-05	3.62	2.84E-05	3.46
	32	1.25E-06	4.44	1.77E-06	4.49	1.24E-06	4.03	1.63E-06	4.11
	64	8.13E-08	3.94	1.19E-07	3.89	7.75E-08	4.00	1.00E-07	4.01

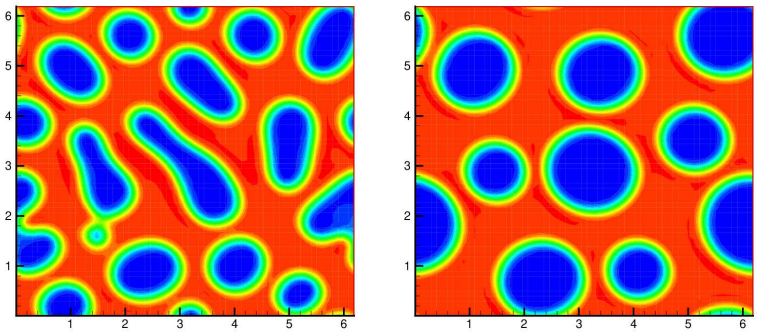
Next, we use the fourth-order SDC scheme to show the long time simulation of spinodal decomposition. We use the following initial condition:

$$\phi_0(x,y) = 0.2 + 0.001\text{rand}(x,y),$$
$$\rho_0(x,y) = 0.2 + 0.001\text{rand}(x,y).$$

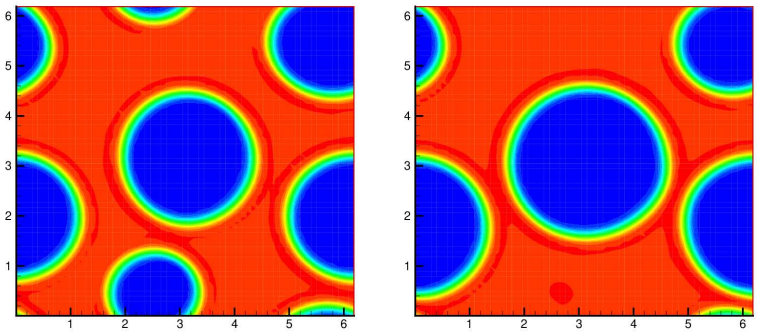
The computational parameters are $\Delta x = 2\pi/N$ with $N = 64$ and the piecewise $\mathcal{P}^2$ approximation; the time step $\Delta t = 0.1\Delta x$. Fig. 3.4 presents the numerical results at different times, which shows the spinodal decomposition phenomenon.



(a) $t = 0.5$ (b) $t = 1$



(c) $t = 2$ (d) $t = 10$



(e) $t = 50$ (f) $t = 100$

Fig. 3.4. Numerical results of the Cahn-Hilliard phase field model of the binary fluid-surfactant system using the fourth-order SDC scheme with time step $\Delta t = 0.1\Delta x$.

4. Concluding Remarks

In this paper, we proposed a class of semi-implicit SDC methods, which were based on second-order time integration methods, and the order of accuracy were increased by two for each additional iteration. We adopted the SDC methods to solve the convection diffusion equation, the surface diffusion of graphs, the nonlinear Schrödinger equation and the phase field models. Numerical experiments showed the efficiency, robustness and accuracy of the proposed SDC methods. We claim that the methods can be used to solve a large class of PDEs beyond those addressed in this paper.

Appendix A Proof of Proposition 2.1

Proof. Assume $u_k^{n,0} = u(t^n)$, $k = 1, \ldots, K+1$. Taking the difference of

$$u(t^{n,m+1}) = u(t^{n,m}) + \int_{t^{n,m}}^{t^{n,m+1}} F(\tau, u(\tau), u(\tau))d\tau \tag{A.1}$$

and (2.10), we have

$$\begin{aligned}
&u(t^{n,m+1}) - u_{k+1}^{n,m+1} \\
&= u(t^{n,m}) - u_{k+1}^{n,m} - \Delta t^{n,m}\left(F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k+1}^{n,m+\frac{1}{2}}, u_{k+1}^{n,m+\frac{1}{2}}\big) - F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, u_{k}^{n,m+\frac{1}{2}}\big)\right) \\
&\quad + \int_{t^{n,m}}^{t^{n,m+1}} F(\tau, u(\tau), u(\tau))d\tau - I_m^{m+1}(u_k).
\end{aligned} \tag{A.2}$$

By induction on both m and k, we assume, when $k \le P$,

$$u(t^{n,l}) - u_k^{n,l} = \mathcal{O}(h^{2k+2}), \quad \forall l \quad \text{in level} \quad k, \tag{A.3}$$

$$u(t^{n,l}) - u_{k+1}^{n,l} = \mathcal{O}(h^{2k+4}), \quad \text{for} \quad l \le m \quad \text{in level} \quad k+1. \tag{A.4}$$

Then we have

$$\begin{aligned}
&u(t^{n,m}) - u_{k+1}^{n,m} = \mathcal{O}(h^{2k+4}), \\
&F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k+1}^{n,m+\frac{1}{2}}, u_{k+1}^{n,m+\frac{1}{2}}\big) - F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, u_{k}^{n,m+\frac{1}{2}}\big) \\
&= F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k+1}^{n,m+\frac{1}{2}}, u_{k+1}^{n,m+\frac{1}{2}}\big) - F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, u_{k+1}^{n,m+\frac{1}{2}}\big) \\
&\quad + F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, u_{k+1}^{n,m+\frac{1}{2}}\big) - F\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, u_{k}^{n,m+\frac{1}{2}}\big) \\
&= F_2'\big(\widetilde{t}^{n,m+\frac{1}{2}}, \xi_1, u_{k+1}^{n,m+\frac{1}{2}}\big)\big(\widetilde{u}_{k+1}^{n,m+\frac{1}{2}} - \widetilde{u}_{k}^{n,m+\frac{1}{2}}\big) \\
&\quad + F_3'\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, \xi_2\big)\big(u_{k+1}^{n,m+\frac{1}{2}} - u_{k}^{n,m+\frac{1}{2}}\big) \\
&= F_2'\big(\widetilde{t}^{n,m+\frac{1}{2}}, \xi_1, u_{k+1}^{n,m+\frac{1}{2}}\big)\mathcal{O}(h^{2k+2}) \\
&\quad + F_3'\big(\widetilde{t}^{n,m+\frac{1}{2}}, \widetilde{u}_{k}^{n,m+\frac{1}{2}}, \xi_2\big)\left(\frac{1}{2}(u_{k+1}^{n,m+1} - u(t^{n,m+1})) + \mathcal{O}(h^{2k+2})\right), \\
&\int_{t^{n,m}}^{t^{n,m+1}} F(\tau, u(\tau), u(\tau))d\tau - I_m^{m+1}(u_k) \\
&= \int_{t^{n,m}}^{t^{n,m+1}} F(\tau, u(\tau), u(\tau))d\tau - I_m^{m+1}(u) + I_m^{m+1}(u) - I_m^{m+1}(u_k) \\
&= \mathcal{O}(h^{2P+1}) + \mathcal{O}(h^{2k+3}).
\end{aligned}$$

Based on all the analysis above, we have

$$u(t^{n,m+1}) - u_{k+1}^{n,m+1} = \mathcal{O}\big(h^{\min[2k+3,2P+1]}\big). \tag{A.5}$$

This completes the proof. □

Appendix B LDG Method and Energy Stability for the Schrödinger Equation

In order to describe the LDG method, we first briefly introduce the following notations. Let $\mathcal{T}_h = \{K\}$ be a shape-regular subdivision of Ω. $\mathcal{E}_h$ denotes the union of the boundary faces of elements $K \in \mathcal{T}_h$, and $\mathcal{E}_0 = \mathcal{E}_h \setminus \partial\Omega$. The discontinuous Galerkin finite element spaces are denoted by

$$\begin{aligned} {}_cV_h &= \Big\{v \in L^2(\Omega) :\ v|_K \in \mathcal{P}^k(K),\ \forall K \in \mathcal{T}_h\Big\},\\ {}_c\boldsymbol{W}_h &= \Big\{\boldsymbol{w} \in (L^2(\Omega))^d :\ \boldsymbol{w}|_K \in (\mathcal{P}^k(K))^d,\ \forall K \in \mathcal{T}_h\Big\}, \end{aligned}$$

where $\mathcal{P}^k(K)$ is the space of complex polynomials of degree at most k on K. Note that functions in ${}_cV_h$ and ${}_c\boldsymbol{W}_h$ are allowed to have discontinuities across element interfaces.

In order to develop the LDG scheme to solve the linear scheme (3.10)-(3.11), we rewrite it into a system of the first order equations

$$\begin{aligned} &\mathrm{i}\frac{u^{n+1}-u^n}{\Delta t} = -\nabla\cdot\boldsymbol{q}^{n+\frac12} - \widetilde{H}^{n+\frac12}W^{n+\frac12},\\ &\boldsymbol{q}^{n+1} = \nabla u^{n+1},\\ &\frac{W^{n+1}-W^n}{\Delta t} = \frac12\widetilde{H}^{n+\frac12}\frac{u^{n+1}-u^n}{\Delta t}. \end{aligned}$$

Then the fully-discrete LDG scheme becomes the following: Find $u^{n+1}, W^{n+1} \in {}_cV_h$ and $\boldsymbol{q}^{n+1} \in {}_c\boldsymbol{W}_h$ such that for all test functions $\varphi, \xi \in {}_cV_h$ and $\boldsymbol{\eta} \in {}_c\boldsymbol{W}_h$, we have

$$\mathrm{i}\int_K \frac{u^{n+1}-u^n}{\Delta t}\varphi dK = \int_K \boldsymbol{q}^{n+\frac12}\cdot\nabla\varphi dK - \int_{\partial K}\widehat{\boldsymbol{q}}^{n+\frac12}\cdot\boldsymbol{\nu}\varphi ds - \int_K \widetilde{H}^{n+\frac12}W^{n+\frac12}\varphi dK, \tag{B.1a}$$

$$\int_K \boldsymbol{q}^{n+1}\cdot\boldsymbol{\eta}dK = -\int_K u^{n+1}\nabla\cdot\boldsymbol{\eta}dK + \int_{\partial K}\widehat{u}^{n+1}\boldsymbol{\eta}\cdot\boldsymbol{\nu}ds, \tag{B.1b}$$

$$\int_K \frac{W^{n+1}-W^n}{\Delta t}\xi dK = \frac12\int_K \widetilde{H}^{n+\frac12}\frac{u^{n+1}-u^n}{\Delta t}\xi dK. \tag{B.1c}$$

Here $\widehat{\boldsymbol{q}}^{n+1}$ and $\widehat{u}^{n+1}$ are the numerical fluxes. To complete the definition of the LDG method, we need to define these numerical fluxes.

Let e be an interior face shared by the "left" and "right" elements K_L and K_R, and $\boldsymbol{\nu}_L$, $\boldsymbol{\nu}_R$ are the normal vectors on e pointing exterior to K_L and K_R, respectively. We aim to uniquely define "left" and "right" for each face according to any fixed rule. For example, we choose $\boldsymbol{\nu}_0$ as a constant vector and $\boldsymbol{\nu}_L\cdot\boldsymbol{\nu}_0 \neq 0$. The left element K_L to the face e requires that $\boldsymbol{\nu}_L\cdot\boldsymbol{\nu}_0 < 0$, and the right one K_R requires $\boldsymbol{\nu}_L\cdot\boldsymbol{\nu}_0 > 0$. If u is a function on K_L and K_R, but possibly discontinuous across e, let u_L denote $(u|_{K_L})|_e$ and u_R denote $(u|_{K_R})|_e$, the left and right trace, respectively. Here we use the simple alternating numerical fluxes

$$\widehat{\boldsymbol{q}}^{n+1} = \boldsymbol{q}_L^{n+1}, \quad \widehat{u}^{n+1} = u_R^{n+1}. \tag{B.2}$$

Next, we will prove the energy stability for fully-discrete LDG scheme (B.1) with the alternating numerical fluxes (B.2) and the periodic boundary condition.

Proposition B.1. *The solution to the LDG scheme* (B.1) *with numerical fluxes* (B.2) *and the periodic boundary condition satisfies the energy stability*

$$\widetilde{E}(\boldsymbol{q}^{n+1}, W^{n+1}) = \widetilde{E}(\boldsymbol{q}^{n}, W^{n}), \tag{B.3}$$

where

$$\widetilde{E}(\boldsymbol{q}, W) = \int_{\Omega} \left(-\frac{1}{2}|\boldsymbol{q}|^2 + |W|^2 \right) d\boldsymbol{x}.$$

Proof. Let $\mathcal{D}u$ denote $u^{n+1} - u^n$. For Eq. (B.1a), choosing the test function $\varphi = \mathcal{D}u^*$, we have

$$\begin{aligned} \frac{\mathrm{i}}{\Delta t} \int_K |\mathcal{D}u|^2 dK = & \int_K \boldsymbol{q}^{n+\frac{1}{2}} \cdot \nabla(\mathcal{D}u^*) dK - \int_{\partial K} \widehat{\boldsymbol{q}}^{n+\frac{1}{2}} \cdot \boldsymbol{\nu} (\mathcal{D}u^*) ds \\ & - \int_K \widetilde{H}^{n+\frac{1}{2}} W^{n+\frac{1}{2}} (\mathcal{D}u^*) dK. \end{aligned} \tag{B.4}$$

Taking the conjugate of (B.1a) and choosing the test function $\varphi = \mathcal{D}u$, we get

$$\begin{aligned} \frac{-\mathrm{i}}{\Delta t} \int_K |\mathcal{D}u|^2 dK = & \int_K \boldsymbol{q}^{n+\frac{1}{2},*} \cdot \nabla(\mathcal{D}u) dK - \int_{\partial K} \widehat{\boldsymbol{q}}^{n+\frac{1}{2},*} \cdot \boldsymbol{\nu} (\mathcal{D}u) ds \\ & - \int_K \widetilde{H}^{n+\frac{1}{2},*} W^{n+\frac{1}{2},*} (\mathcal{D}u) dK. \end{aligned} \tag{B.5}$$

For Eq. (B.1b), taking the difference between two time levels and choosing test function $\boldsymbol{\eta} = -\boldsymbol{q}^{n+\frac{1}{2},*}$ give

$$-\frac{1}{2} \int_K (|\boldsymbol{q}^{n+1}|^2 - |\boldsymbol{q}^n|^2) dK = \int_K (\mathcal{D}u) \nabla \cdot \boldsymbol{q}^{n+\frac{1}{2},*} dK - \int_{\partial K} (\mathcal{D}\widehat{u}) \boldsymbol{q}^{n+\frac{1}{2},*} \cdot \boldsymbol{\nu} ds. \tag{B.6}$$

Taking the conjugate of (B.1b) and the difference between two time levels, choosing $\boldsymbol{\eta} = -\boldsymbol{q}^{n+\frac{1}{2}}$ give

$$-\frac{1}{2} \int_K (|\boldsymbol{q}^{n+1}|^2 - |\boldsymbol{q}^n|^2) dK = \int_K (\mathcal{D}u^*) \nabla \cdot \boldsymbol{q}^{n+\frac{1}{2}} dK - \int_{\partial K} (\mathcal{D}\widehat{u}^*) \boldsymbol{q}^{n+\frac{1}{2}} \cdot \boldsymbol{\nu} ds. \tag{B.7}$$

In Eq. (B.1c), we choose the test function $\xi = 2W^{n+\frac{1}{2},*}$ to obtain

$$\int_K (|W^{n+1}|^2 - |W^n|^2) dK = \int_K \widetilde{H}^{n+\frac{1}{2}} (\mathcal{D}u) W^{n+\frac{1}{2},*} dK, \tag{B.8}$$

and also take the conjugate of (B.1c), choose the test function $\xi = 2W^{n+\frac{1}{2}}$, one can obtain

$$\int_K (|W^{n+1}|^2 - |W^n|^2) dK = \int_K \widetilde{H}^{n+\frac{1}{2},*} (\mathcal{D}u^*) W^{n+\frac{1}{2}} dK. \tag{B.9}$$

Let (B.4)+(B.5)+(B.6)+(B.7)+(B.8)+(B.9), and with the help of the alternating numerical fluxes (B.2) on interior faces and the periodic boundary condition on boundary faces, we obtain

$$-\frac{1}{2} \int_{\Omega} (|\boldsymbol{q}^{n+1}|^2 - |\boldsymbol{q}^n|^2) d\boldsymbol{x} + \int_K (|W^{n+1}|^2 - |W^n|^2) d\boldsymbol{x} = 0, \tag{B.10}$$

which implies the energy stability (B.3). □

Acknowledgments. Research of R. Guo is supported by NSFC (Grant No. 11601490). Research of Y. Xu is supported by NSFC (Grant No.12071455).

References

[1] K. Böhmer, H. Stetter, Defect Correction Methods, Theory and Applications, Springer-Verlag, Wien-New York, 1984.

[2] S. Boscarino, F. Filbet and G. Russo, High order semi-implicit schemes for time dependent partial differential equations, *J. Sci. Comput.*, **66** (2016), 1–27.

[3] S. Boscarino and J.M. Qiu, Error estimates of the integral deferred correction method for stiff problems, *ESIAM Math. Model. Numer. Anal.*, **50** (2016), 1137–1166.

[4] S. Boscarino, J.M. Qiu and G. Russo, Implicit-explicit integral deferred correction methods for stiff problems, *SIAM J. Sci. Comput.*, **40** (2018), A787–A816.

[5] L. Chen and J. Shen, Applications of semi-implicit Fourier-spectral method to phase field equations, *Comput. Phys. Commun.*, **108** (1998), 147–158.

[6] W. Chen, S. Conde, C. Wang, X. Wang and S.M. Wise, A linear energy stable scheme for a thin film model without slope selection, *J. Sci. Comput.*, **52** (2012), 546–562.

[7] A. Christlieb, B. Ong and J.M. Qiu, Comments on high-order integrators embedded within integral deferred correction methods, *Commun. Appl. Math. Comput. Sci.*, **4** (2009), 27–56.

[8] A. Christlieb, B. Ong and J.M. Qiu, Integral deferred correction methods constructed with high order Runge-Kutta integrators, *Math. Comput.*, **79** (2010), 761–783.

[9] A. Christlieb, M. Morton, B. Ong and J.M. Qiu, Semi-implicit integral deferred correction constructed with additive Runge-Kutta methods, *Commun. Math. Sci.*, **9** (2011), 879–902.

[10] S.M. Cox and P.C. Matthews, Exponential time differencing for stiff systems, *J. Comput. Phys.*, **176** (2002), 430–455.

[11] A. Dutt, L. Greengard and V. Rokhlin, Spectral deferred correction methods for ordinary differential equations, *BIT*, **40** (2000), 241–266.

[12] D.J. Eyre, Unconditionally gradient stable time marching the Cahn-Hilliard equation, *Mrs Online Proceedings Library Archive*, **529** (1998), 39–46.

[13] X. Feng, T. Tang and J. Yang, Stabilized Crank-Nicolson/Adams-Bashforth schemes for phase field models, *East Asian J. Appl. Math.*, **3** (2013), 59–80.

[14] X. Feng, T. Tang and J. Yang, Long time numerical simulation for phase-field problems using p-adaptive spectral deferred correction methods, *SIAM J. Sci. Comput.*, **37** (2015), A271–A294.

[15] Y. Gong, J. Zhao and Q. Wang, Second order fully discrete energy stable methods on staggered grids for hydrodynamics phase field models of binary viscous fluids, *SIAM J. Sci. Comput.*, **40** (2018), B528–B553.

[16] Y. Gong, J. Zhao, X. Yang and Q. Wang, Second-order linear schemes for hydrodynamic phase field models of viscous fluid flows with variable densities, *SIAM J. Sci. Comput.*, **4** (2018), B138–B167.

[17] L. Guo and Y. Xu, Energy conserving local discontinuous Galerkin methods for the nonlinear Schrödinger equation with wave operator, *J. Sci. Comput.*, **65** (2015), 622–647.

[18] R. Guo and Y. Xu, Efficient solvers of discontinuous Galerkin discretization for the Cahn-Hilliard equations, *J. Sci. Comput.*, **58** (2014), 380–408.

[19] R. Guo and Y. Xu, Local discontinuous Galerkin method and high order semi-implicit scheme for the phase field crystal equation, *SIAM J. Sci. Comput.*, **38** (2016), A105–A127.

[20] R. Guo, Y. Xia and Y. Xu, Semi-implicit spectral deferred correction methods for highly nonlinear partial differential equations, *J. Comput. Phys.*, **338** (2017), 269–284.

[21] B. Gustafsson and W. Kress, Deferred correction methods for initial value problems, *BIT*, **41** (2001), 986–995.

[22] D. Han, A. Brylev, X. Yang and Z. Tan, Numerical analysis of second order, fully discrete energy stable schemes for phase field models of two phase incompressible flows, *J. Sci. Comput.*, **70** (2017), 965–989.

[23] E. Hairer and G. Wanner, Solving Ordinary Differential Equations II: Stiff and Differential-

Algebraic Problems. Second revised edition, Springer Series in Computational Mathematics, 14. Springer-Verlag, Berlin, 2010.

[24] Y. He, Y. Liu and T. Tang, On large time-stepping methods for the Cahn-Hilliard equation, *Appl. Numer. Math.*, **57** (2007), 616–628.

[25] L. Ju, J. Zhang and Q. Du, Fast and accurate algorithms for simulating coarsening dynamics of Cahn-Hilliard equations, *Comput. Mater. Sci.*, **108** (2015), 272–282.

[26] C.A. Kennedy and M.H. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations, *Appl. Num. Math.*, **44** (2003), 139–181.

[27] X. Li, L. Ju and X. Meng, Convergence analysis of exponential time differencing schemes for the Cahn-Hilliard equation, *Commun. Comput. Phys.*, **26** (2019), 1510–1529.

[28] F. Liu and J. Shen, Stabilized semi-implicit spectral deferred correction methods for Allen-Cahn and Cahn-Hilliard equations, *Math. Methods Appl. Sci.*, **38** (2016), 4564–4575.

[29] M.L. Minion, Semi-implicit spectral deferred correction methods for ordinary differential equations, *Commun. Math. Sci.*, **1** (2003), 471–500.

[30] J. Shen, J. Xu and J. Yang, The scalar auxiliary variable (SAV) approach for gradient flows, *J. Comput. Phys.*, **353** (2017), 407–416.

[31] J. Shen, J. Xu and J. Yang, A new class of efficient and robust energy stable schemes for gradient flows, *SIAM Rev.*, **61** (2019), 474–506.

[32] T. Tang and Z. Qiao, Efficient numerical methods for phase-field equations, *SCIENTIA SINICA Mathematica*, **48** (2020), 1–20.

[33] U. Trottenberg, C. Oosterlee and A. Schüller, Multigrid, Academic Press, New York, 2005.

[34] Y. Xia, Y. Xu and C.W. Shu, Efficient time discretization for local discontinuous Galerkin methods, *Discret. Contin. Dyn. Syst. Ser. B*, **8** (2007), 677–693.

[35] Y. Xia, A fully discrete stable discontinuous Galerkin method for the thin film epitaxy problem without slope selection, *J. Comput. Phys.*, **280** (2015), 248–260.

[36] Y. Xu and C.W. Shu, Local discontinuous Galerkin methods for nonlinear Schrödinger equations, *J. Comput. Phys.*, **205** (2005), 72–97.

[37] Y. Xu and C.W. Shu, Local discontinuous Galerkin methods for high-order time-dependent partial differential equations, *Commun. Comput. Phys.*, **7** (2010), 1–46.

[38] X. Yang, Linear, first and second-order, unconditionally energy stable numerical schemes for the phase field model of homopolymer blends, *J. Comput. Phys.*, **327** (2016), 294–316.

[39] X. Yang and L. Ju, Efficient linear schemes with unconditional energy stability for the phase field elastic bending energy model, *Comput. Methods Appl. Mech. Engrg.*, **315** (2017), 691–712.

[40] X. Yang and L. Ju, Linear and unconditionally energy stable schemes for the binary fluid-surfactant phase field model, *Comput. Methods Appl. Mech. Engrg.*, **318** (2017), 1005–1029.

[41] X. Yang, Numerical approximations for the Cahn-Hilliard phase field model of the binary fluid-surfactant system, *J. Sci. Comput.*, **74** (2018), 1533–1553.