J Sci Comput (2014) 58:380–408
DOI 10.1007/s10915-013-9738-4

Efficient Solvers of Discontinuous Galerkin Discretization for the Cahn–Hilliard Equations

Ruihan Guo · Yan Xu

Received: 24 October 2012 / Revised: 14 May 2013 / Accepted: 20 May 2013 /
Published online: 2 June 2013

Abstract In this paper, we develop and analyze a fast solver for the system of algebraic equations arising from the local discontinuous Galerkin (LDG) discretization and implicit time marching methods to the Cahn–Hilliard (CH) equations with constant and degenerate mobility. Explicit time marching methods for the CH equation will require severe time step restriction ($\Delta t \sim O(\Delta x^4)$), so implicit methods are used to remove time step restriction. Implicit methods will result in large system of algebraic equations and a fast solver is essential. The multigrid (MG) method is used to solve the algebraic equations efficiently. The Local Mode Analysis method is used to analyze the convergence behavior of the linear MG method. The discrete energy stability for the CH equations with a special homogeneous free energy density $\Psi(u) = \frac{1}{4}(1-u^2)^2$ is proved based on the convex splitting method. We show that the number of iterations is independent of the problem size. Numerical results for one-dimensional, two-dimensional and three-dimensional cases are given to illustrate the efficiency of the methods. We numerically show the optimal complexity of the MG solver for $\mathcal{P}^1$ element. For $\mathcal{P}^2$ approximation, the optimal or sub-optimal complexity of the MG solver are numerically shown.

Keywords Cahn–Hilliard equation · Local discontinuous Galerkin method · Convex splitting method · Multigrid algorithm · FAS multigrid · Additive Runge–Kutta · Diagonally implicit Runge–Kutta · Local mode analysis

Mathematics Subject Classification (2010) 65M60 · 35K55

Research supported by NSFC Grant No. 10971211, No. 11031007, FANEDD No. 200916, NCET No. 09-0922, Fok Ying Tung Education Foundation No. 131003.

R. Guo · Y. Xu (✉)
School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, People's Republic of China
e-mail: yxu@ustc.edu.cn

R. Guo
e-mail: guoguo88@mail.ustc.edu.cn

1 Introduction

The Cahn–Hilliard (CH) equation in $\Omega \in \mathbb{R}^d (d \leq 3)$ is

$$\begin{cases} u_t = \nabla \cdot (b(u)\nabla w), \\ w = -\gamma \Delta u + \Psi'(u), \end{cases} \tag{1.1}$$

with the boundary condition

$$\frac{\partial u}{\partial \nu} = b(u)\frac{\partial w}{\partial \nu} = 0, \quad \text{on} \quad \partial\Omega, \tag{1.2}$$

where γ is a positive constant. The mobility $b(u)$ is non-negative and the function $\Psi(u)$ represents the homogeneous free energy in the energy functional

$$\varepsilon(u) = \int_{\Omega} (\frac{\gamma}{2}|\nabla u|^2 + \Psi(u))dx.$$

For the fourth order nonlinear CH equation, explicit time discretization methods will require severe time step restriction ($\Delta t \sim O(\Delta x^4)$), so we explore implicit methods. Being implicit in time, it requires to solve the system of algebraic equations at each time step, arising by the local discontinuous Galerkin (LDG) spatial discretization and high order time discretization methods. The efficiency of the implicit methods highly depends on the efficiency of the solver.

Our main interest is to explore an efficient high order implicit time discretization method for the CH equation coupled with the LDG spatial discretization. In this paper, we will apply the implicit additive Runge–Kutta (ARK) method to the CH equation and the numerical results show that it is an efficient time discretization method for the CH equation coupled with the LDG method. The third order ARK method requires to solve three linear systems of equations at each time step. Traditionally iterative methods such as Gauss–Seidel method suffer from slow convergence rates. We will apply the MG method to accelerate the convergence rates when solving the system of equations, which derived by the LDG spatial discretization and high order time marching method for the CH equation. In order to predict the MG behavior, a two-level Local Mode Analysis is used to study the convergence of the MG method.

The mobility $b(u)$ in the CH equation can be constant or degenerate. For the degenerate mobility, the ARK method with the linear MG solver is not efficient. We will apply the diagonally implicit Runge–Kutta (DIRK) [1] time discretization method to treat the nonlinear CH equation. Then it requires to solve nonlinear systems of algebraic equations at each time step. The nonlinear Gauss–Seidel method and the Newton method can be used to solve the nonlinear equations, but they are not effective for large system. We will apply the nonlinear Full Approximation Scheme (FAS) MG method coupled with the LDG spatial discretization for the CH equation and the numerical results show that the convergence rates of the method is $O(N)$. We also prove the unconditional energy stability for the first order scheme in time based on the convex splitting method. We numerically show the optimal complexity of the MG solver for $\mathcal{P}^1$ element. For $\mathcal{P}^2$ approximation, the optimal or sub-optimal complexity of the MG solver are numerically shown.

The discontinuous Galerkin (DG) method is a class of finite element methods using completely discontinuous piecewise polynomials as the solution and the test spaces. Reed and Hill [26] first introduced the DG method in 1973, in the framework of neutron linear transport. For partial differential equations (PDEs) containing higher than first order spatial derivatives, the DG method can not be applied directly, so the LDG method was introduced. The first LDG

method was introduced by Cockburn and Shu [12] for time-dependent convection-diffusion systems. The idea of the LDG method is to rewrite the equations with higher order derivatives as a first order system, then apply the DG method to the system. In [35], the ARK method was explored to solve the stiff ordinary differential equations (ODEs) resulting from a LDG spatial discretization to PDEs with higher order spatial derivatives. For a detailed description about the LDG methods for high-order time-dependent PDEs, we refer the readers to [36]. Recently, the MG method coupled with the DG spatial discretization for the compressible Naiver–Stokes equation [25,27] and the Euler equation [5,6] have been studied. In [30,31], the MG method was introduced to solve the system of algebraic equations arising from the higher order DG discretization of advection dominated flows.

Many numerical methods have been developed to treat the CH equation, using finite elements [2–4,7,8,13–15,17], discontinuous Galerkin methods [11,18,32], multigrid method [22–24] and finite difference methods [16,19,28]. Xia et al. [34] developed the LDG methods for the CH equation, which was high order accurate, nonlinear stable and flexible for arbitrary h and p adaptivity. The explicit time discretization method was used in this paper and it lead to strict restrict on time step, so implicit methods should be used to improve computational efficiency.

The rest of the paper is organized as following. In Sect. 2, we describe a full-discrete LDG scheme for CH equation and prove the unconditional energy stability based on the convex splitting method. In Sect. 3, we introduce two different high order time discretization methods, the ARK method and the DIRK method. In Sect. 4, we study the convergence of the linear bi-grid algorithm numerically. Section 5 contains numerical results for the nonlinear problems which include the CH equation for one-dimensional, two-dimensional and three-dimensional cases. We give some concluding remarks in Sect. 6. In the "Appendices 1 and 2", we give a detailed description of the linear MG method and the nonlinear FAS MG method. A fairly complete description of the Local Mode Analysis for the bi-grid algorithm is given in "Appendix 3".

2 The Convex Splitting LDG Method

2.1 Notations

Let $\mathcal{T}_h$ denote a tessellation of Ω with shape-regular element K. Let Γ denote the union of the boundary faces of elements $K \in \mathcal{T}_h$, i.e. $\Gamma = \bigcup_{K\in\mathcal{T}_h} \partial K$, and $\Gamma_0 = \Gamma \backslash \partial\Omega$.

In order to describe the flux functions, we need to introduce some notations. Let e be a face shared by the "left" and "right" elements K_L and K_R (we refer to [36] for more details of the definition). Define the normal vectors $\boldsymbol{\nu}_L$ and $\boldsymbol{\nu}_R$ on e pointing exterior to K_L and K_R, respectively. If ψ is a function on K_L and K_R, but possibly discontinuous across e, let ψ_L denote $(\psi|_{K_L})|_e$ and ψ_R denote $(\psi|_{K_R})|_e$, the left and right trace, respectively.

Let $\mathcal{P}^k(K)$ be the space of polynomials of degree at most $k \geq 0$ on K. The finite element spaces associated with the mesh are of the form

$$V_h^k = \{v \in L^2(\Omega) : v|_K \in \mathcal{P}^k(K), \forall K \in \mathcal{T}_h\},$$
$$\Sigma_h^k = \{\mathbf{w} = (w_1, \ldots, w_d)^T \in L^2(\Omega)^d : w_l|_K \in \mathcal{P}^k(K), l = 1, \ldots, d, \forall K \in \mathcal{T}_h\}$$

Further, we define the inner product notations as

$$(w, v)_K = \int_K wv dK, \quad (w, v)_{\partial K} = \int_{\partial K} wv ds, \tag{2.1}$$

$$(\mathbf{q},\mathbf{p})_K = \int_K \mathbf{q}\cdot\mathbf{p}dK, \quad (\mathbf{q},\mathbf{p})_{\partial K} = \int_{\partial K} \mathbf{q}\cdot\mathbf{p}ds, \tag{2.2}$$

for the scalar variables w, v and vector variables $\mathbf{q}$, $\mathbf{p}$ respectively. The inner products on Ω are defined as

$$(w,v)_\Omega = \sum_K (w,v)_K, \quad (\mathbf{q},\mathbf{p})_\Omega = \sum_K (\mathbf{q},\mathbf{p})_K. \tag{2.3}$$

2.2 Full-Discrete LDG Scheme

Wise [33] introduced a convex splitting finite difference scheme for the Cahn–Hilliard–Hele–Shaw (CHHS) equation and proved the unconditional energy stability. In this subsection, we apply the convex splitting method to CH equation coupled with the LDG spatial discretization for a special choice of $\Psi(u)$. We refer the readers to [34] for the general LDG scheme. The homogeneous free energy density $\Psi(u)$ is taken as $\frac{1}{4}(1-u^2)^2$ and the full-discrete convex splitting scheme is

$$\frac{u^{n+1}-u^n}{\Delta t} = \nabla\cdot(b(u^n)\nabla(-\gamma\Delta u^{n+1} + (u^{n+1})^3 - u^n)). \tag{2.4}$$

In order to define the LDG method to the equation (2.4), we first rewrite it as a first order system:

$$\frac{u^{n+1}-u^n}{\Delta t} = \nabla\cdot\mathbf{s}^{n+1}, \tag{2.5a}$$
$$\mathbf{s}^{n+1} = b(u^n)\mathbf{p}^{n+1}, \tag{2.5b}$$
$$\mathbf{p}^{n+1} = \nabla(r^{n+1} - q^{n+1}), \tag{2.5c}$$
$$q^{n+1} = \gamma\nabla\cdot\mathbf{w}^{n+1}, \tag{2.5d}$$
$$\mathbf{w}^{n+1} = \nabla u^{n+1}, \tag{2.5e}$$
$$r^{n+1} = (u^{n+1})^3 - u^n. \tag{2.5f}$$

To simplify the notation, we still use $u^{n+1}, \mathbf{s}^{n+1}, \mathbf{p}^{n+1}, q^{n+1}, \mathbf{w}^{n+1}$ and r^{n+1} to denote the numerical solutions. The LDG scheme to solve the system (2.5) is as follows: Find $(u^{n+1}, \mathbf{s}^{n+1}, \mathbf{p}^{n+1}, q^{n+1}, \mathbf{w}^{n+1}, r^{n+1}) \in V_h^k \times \Sigma_h^k \times \Sigma_h^k \times V_h^k \times \Sigma_h^k \times V_h^k$ such that, $\forall \rho, \boldsymbol{\theta}, \boldsymbol{\eta}, \varphi, \boldsymbol{\phi}, \xi \in V_h^k \times \Sigma_h^k \times \Sigma_h^k \times V_h^k \times \Sigma_h^k \times V_h^k$

$$(\frac{u^{n+1}-u^n}{\Delta t}, \rho)_K = -(\mathbf{s}^{n+1}, \nabla\rho)_K + (\widehat{\mathbf{s}^{n+1}\cdot\boldsymbol{\nu}}, \rho)_{\partial K}, \tag{2.6a}$$
$$(\mathbf{s}^{n+1}, \boldsymbol{\theta})_K = (b(u^n)\mathbf{p}^{n+1}, \boldsymbol{\theta})_K, \tag{2.6b}$$
$$(\mathbf{p}^{n+1}, \boldsymbol{\eta})_K = -(r^{n+1} - q^{n+1}, \nabla\cdot\boldsymbol{\eta})_K + (\widehat{r}^{n+1} - \widehat{q}^{n+1}, \boldsymbol{\nu}\cdot\boldsymbol{\eta})_{\partial K}, \tag{2.6c}$$
$$(q^{n+1}, \varphi)_K = -\gamma(\mathbf{w}^{n+1}, \nabla\varphi)_K + \gamma(\widehat{\mathbf{w}^{n+1}\cdot\boldsymbol{\nu}}, \varphi)_{\partial K}, \tag{2.6d}$$
$$(\mathbf{w}^{n+1}, \boldsymbol{\phi})_K = -(u^{n+1}, \nabla\cdot\boldsymbol{\phi})_K + (\widehat{u}^{n+1}, \boldsymbol{\phi}\cdot\boldsymbol{\nu})_{\partial K}, \tag{2.6e}$$
$$(r^{n+1}, \xi)_K = ((u^{n+1})^3 - u^n, \xi)_K. \tag{2.6f}$$

The "hat" terms in (2.6) at the cell boundary from integration by parts are the so-called "numerical fluxes", which are functions defined on the edges and should be designed based on different guiding principles for different PDEs to ensure stability. It turns out that we can take the simple choices such that

$$\widehat{\mathbf{s}}^{n+1}|_e = \widehat{\mathbf{s}}_L^{n+1}, \widehat{q}^{n+1}|_e = \widehat{q}_R^{n+1}, \widehat{r}^{n+1}|_e = \widehat{r}_R^{n+1}, \widehat{\mathbf{w}}^{n+1}|_e = \widehat{\mathbf{w}}_L^{n+1}, \widehat{u}^{n+1}|_e = \widehat{u}_R^{n+1}. \quad (2.7)$$

According to the boundary condition (1.2), we take

$$\widehat{\mathbf{s}}^{n+1} = \mathbf{0}, \widehat{\mathbf{w}}^{n+1} = \mathbf{0}, \widehat{u}^{n+1} = (u^{n+1})^{\text{in}}, \widehat{q}^{n+1} = (q^{n+1})^{\text{in}}, \widehat{r}^{n+1} = (r^{n+1})^{\text{in}}, \quad (2.8)$$

at the domain boundary, where $(u^{n+1})^{\text{in}}$ means the value taking from the inside of the boundary element.

We choose a local basis in cell K, then $\mathbf{s}^{n+1}$, $\mathbf{p}^{n+1}$, $\mathbf{w}^{n+1}$ and r^{n+1} can be eliminated from equations (2.6b), (2.6c), (2.6e) and (2.6a), respectively, by simply inverting a small mass matrix in each case. Then we obtain a system of two coupled equations for $[u^{n+1}, q^{n+1}]$

$$GU = F, \quad (2.9)$$

where $U = [u^{n+1}, q^{n+1}]^T$ and F is the corresponding right hand side vector consisting of u^n and q^n.

2.3 Energy Stability

In this subsection, we will prove the energy stability for the full-discrete LDG scheme based on the convex splitting method. To simplify the notation, we use the following notations for discretization of time variable,

$$\delta_t u^{n+1} = \frac{u^{n+1} - u^n}{\Delta t},$$
$$\delta_t \mathbf{w}^{n+1} = \frac{\mathbf{w}^{n+1} - \mathbf{w}^n}{\Delta t}.$$

Proposition 1 (Energy stability) *The solution to the scheme* (2.6) *and the flux* (2.7) *satisfies the energy stability*

$$\frac{1}{\Delta t}[\frac{\gamma}{2}(\mathbf{w}^{n+1}, \mathbf{w}^{n+1})_\Omega + (\Psi(u^{n+1}), 1)_\Omega] + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_\Omega \leq \frac{1}{\Delta t}[\frac{\gamma}{2}(\mathbf{w}^n, \mathbf{w}^n)_\Omega + (\Psi(u^n), 1)_\Omega].$$

Proof We take the test functions in (2.6) as

$$\rho = r^{n+1} - q^{n+1}, \quad \boldsymbol{\theta} = -\mathbf{p}^{n+1}, \quad \boldsymbol{\eta} = \mathbf{s}^{n+1},$$
$$\varphi = \delta_t u^{n+1}, \quad \boldsymbol{\phi} = \gamma \delta_t \mathbf{w}^{n+1}, \quad \xi = -\delta_t u^{n+1}. \quad (2.10)$$

Then we have

$$\begin{aligned}
(\delta_t u^{n+1}, r^{n+1} - q^{n+1})_K &= -(\mathbf{s}^{n+1}, \nabla(r^{n+1} - q^{n+1}))_K + (\widehat{\mathbf{s}^{n+1} \cdot \boldsymbol{\nu}}, r^{n+1} - q^{n+1})_{\partial K}, \\
-(\mathbf{s}^{n+1}, \mathbf{p}^{n+1})_K &= -(b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_K, \\
(\mathbf{p}^{n+1}, \mathbf{s}^{n+1})_K &= -(r^{n+1} - q^{n+1}, \nabla \cdot \mathbf{s}^{n+1})_K + (\widehat{r}^{n+1} - \widehat{q}^{n+1}, \boldsymbol{\nu} \cdot \mathbf{s}^{n+1})_{\partial K}, \\
(q^{n+1}, \delta_t u^{n+1})_K &= -\gamma(\mathbf{w}^{n+1}, \nabla \delta_t u^{n+1})_K + \gamma(\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, \delta_t u^{n+1})_{\partial K}, \\
\gamma(\mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_K &= -\gamma(u^{n+1}, \nabla \cdot \delta_t \mathbf{w}^{n+1})_K + \gamma(\widehat{u}^{n+1}, \delta_t \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K}, \\
-(r^{n+1}, \delta_t u^{n+1})_K &= -((u^{n+1})^3 - u^n, \delta_t u^{n+1})_K.
\end{aligned} \quad (2.11)$$

Summing up equations in (2.11), we obtain

$$(\gamma \mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_K + (((u^{n+1})^3 - u^n)\delta_t u^{n+1}, 1)_K + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_K$$

$$
\begin{aligned}
&= -(r^{n+1} - q^{n+1}, \mathbf{s}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} + (\widehat{\mathbf{s}^{n+1} \cdot \boldsymbol{\nu}}, r^{n+1} - q^{n+1})_{\partial K} + (\widehat{r}^{n+1} - \widehat{q}^{n+1}, \mathbf{s}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} \\
&\quad + \frac{\gamma}{\Delta t}[-(u^{n+1}, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} + (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^{n+1})_{\partial K} + (\widehat{u}^{n+1}, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K}] \\
&\quad + \frac{\gamma}{\Delta t}[(\mathbf{w}^{n+1}, \nabla u^n)_K + (u^{n+1}, \nabla \cdot \mathbf{w}^n)_K - (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^n)_{\partial K} - (\widehat{u}^{n+1}, \mathbf{w}^n \cdot \boldsymbol{\nu})_{\partial K}].
\end{aligned}
$$

In the above analysis, we choose the test functions as $\delta_t u^{n+1}$, $\delta_t \mathbf{w}^{n+1}$, thus the cross (the time level n and $n+1$) terms $(\mathbf{w}^{n+1}, \nabla u^n)_K$, $(u^{n+1}, \nabla \cdot \mathbf{w}^n)_K$ occur. What we need to do next is to eliminate the cross terms. For equation (2.6e), we choose the test function as $\boldsymbol{\phi} = \mathbf{w}^n$, then we obtain

$$
(\mathbf{w}^{n+1}, \mathbf{w}^n)_K = -(u^{n+1}, \nabla \cdot \mathbf{w}^n)_K + (\widehat{u}^{n+1}, \mathbf{w}^n \cdot \boldsymbol{\nu})_{\partial K}. \tag{2.12}
$$

From (2.6e), we have

$$
(\mathbf{w}^n, \boldsymbol{\phi})_K = -(u^n, \nabla \cdot \boldsymbol{\phi})_K + (\widehat{u}^n, \boldsymbol{\phi} \cdot \boldsymbol{\nu})_{\partial K}. \tag{2.13}
$$

Choosing the test function $\boldsymbol{\phi} = \mathbf{w}^{n+1}$ in (2.13), we obtain

$$
(\mathbf{w}^n, \mathbf{w}^{n+1})_K = -(u^n, \nabla \cdot \mathbf{w}^{n+1})_K + (\widehat{u}^n, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K}. \tag{2.14}
$$

Then we have

$$
\begin{aligned}
&(\mathbf{w}^{n+1}, \nabla u^n)_K + (u^{n+1}, \nabla \cdot \mathbf{w}^n)_K - (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^n)_{\partial K} - (\widehat{u}^{n+1}, \mathbf{w}^n \cdot \boldsymbol{\nu})_{\partial K} \\
&\overset{(2.12)}{=} -(\mathbf{w}^{n+1}, \mathbf{w}^n)_K + (\mathbf{w}^{n+1}, \nabla u^n)_K - (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^n)_{\partial K} \\
&= -(\mathbf{w}^{n+1}, \mathbf{w}^n)_K + (\mathbf{w}^{n+1}, \nabla u^n)_K - (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^n)_{\partial K} + (u^n, \nabla \cdot \mathbf{w}^{n+1})_K \\
&\quad -(\widehat{u}^n, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} - (u^n, \nabla \cdot \mathbf{w}^{n+1})_K + (\widehat{u}^n, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} \\
&\overset{(2.14)}{=} (\mathbf{w}^{n+1}, \nabla u^n)_K - (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^n)_{\partial K} + (u^n, \nabla \cdot \mathbf{w}^{n+1})_K - (\widehat{u}^n, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K}.
\end{aligned}
$$

According to the above relations, we have

$$
\begin{aligned}
&(\gamma \mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_K + (((u^{n+1})^3 - u^n)\delta_t u^{n+1}, 1)_K + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_K \\
&= -(r^{n+1} - q^{n+1}, \mathbf{s}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} + (\widehat{s^{n+1} \cdot \boldsymbol{\nu}}, r^{n+1} - q^{n+1})_{\partial K} + (\widehat{r}^{n+1} - \widehat{q}^{n+1}, \mathbf{s}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} \\
&\quad + \frac{\gamma}{\Delta t}[-(u^{n+1}, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} + (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^{n+1})_{\partial K} + (\widehat{u}^{n+1}, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K}] \\
&\quad + \frac{\gamma}{\Delta t}[(\mathbf{w}^{n+1}, \nabla u^n)_K + (u^n, \nabla \cdot \mathbf{w}^{n+1})_K - (\widehat{u}^n, \mathbf{w}^{n+1} \cdot \boldsymbol{\nu})_{\partial K} - (\widehat{\mathbf{w}^{n+1} \cdot \boldsymbol{\nu}}, u^n)_{\partial K}].
\end{aligned}
$$

Summing up over K, with the numerical fluxes (2.7) and the above specific choices of the fluxes (2.8) at the domain boundary, we get

$$
\begin{aligned}
&(\gamma \mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_\Omega + (((u^{n+1})^3 - u^n)\delta_t u^{n+1}, 1)_\Omega + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_\Omega \\
&= (\gamma \mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_\Omega + (((u^{n+1})^3 - u^n)\delta_t u^{n+1}, 1)_\Omega - \frac{1}{4\Delta t}((1 - (u^{n+1})^2)^2, 1)_\Omega \\
&\quad + \frac{1}{4\Delta t}((1 - (u^n)^2)^2, 1)_\Omega + \frac{1}{4\Delta t}((1 - (u^{n+1})^2)^2, 1)_\Omega - \frac{1}{4\Delta t}((1 - (u^n)^2)^2, 1)_\Omega \\
&\quad + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_\Omega \\
&= (\gamma \mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_\Omega + \frac{1}{4\Delta t}((1 - (u^{n+1})^2)^2, 1)_\Omega - \frac{1}{4\Delta t}((1 - (u^n)^2)^2, 1)_\Omega \\
&\quad + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_\Omega + \frac{1}{4\Delta t}((2 + 2(u^{n+1})^2 + (u^n + u^{n+1})^2(u^n - u^{n+1})^2), 1)_\Omega \\
&= 0.
\end{aligned}
$$

Then we get

$$(\gamma \mathbf{w}^{n+1}, \delta_t \mathbf{w}^{n+1})_\Omega + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_\Omega + \frac{1}{4\Delta t}((1-(u^{n+1})^2)^2 - (1-(u^n)^2)^2, 1)_\Omega \le 0,$$

so we have the energy stability

$$\begin{aligned}&\frac{1}{\Delta t}[\frac{\gamma}{2}(\mathbf{w}^{n+1}, \mathbf{w}^{n+1})_\Omega + (\Psi(u^{n+1}), 1)_\Omega] + (b(u^n)\mathbf{p}^{n+1}, \mathbf{p}^{n+1})_\Omega \\ &\le \frac{1}{\Delta t}[\frac{\gamma}{2}(\mathbf{w}^n, \mathbf{w}^n)_\Omega + (\Psi(u^n), 1)_\Omega].\end{aligned}$$

□

3 The High Order Time Discretization Methods

The scheme (2.4) is stable regardless of the time step size Δt. Thus, we say the scheme is unconditionally energy stable, but it is first order accuracy in time. Furthermore, the convex splitting scheme is just for a special choice of $\Psi(u)(\Psi(u) = \frac{1}{4}(1-u^2)^2)$. Our purpose is to obtain a high order scheme in time and for arbitrary $\Psi(u)$, so we will explore high order time discretization methods for the CH equation in this section.

3.1 The Linearization Scheme

Xia et al. [35] explored the ARK method and found that it was an efficient method when it was coupled with the LDG spatial discretization for solving PDEs containing higher order spatial derivatives. In this section, we apply the ARK method to the CH equation (1.1), which treats the linear part implicitly and the nonlinear part explicitly. But the non-negative mobility $b(u)$ in the CH equation can be constant or degenerate. For the degenerate mobility $b(u)$, we can not apply the ARK method directly because of the nonlinear term $b(u)$, so we consider a linearization technique.

Starting from u^n (the numerical approximation of $u(t^n)$, with $u^0 = u(0)$), we give the method to calculate u^{n+1} in the following.

1. Rewrite the CH equation (1.1) in the following form

$$u_t = \nabla \cdot (b(u^n)\nabla(-\gamma \Delta u)) + \nabla \cdot ((b(u) - b(u^n))\nabla(-\gamma \Delta u)) + \nabla \cdot (b(u)\nabla(\Psi'(u))). \tag{3.1}$$

2. Apply the LDG spatial discretization similar to [34] to the equation (3.1) and obtain an ODE

$$\begin{cases} u_t = F_N(t, u(t)) + F_S(t, u(t)), & t \in [0, T], \\ u(0) = u^0, \end{cases} \tag{3.2}$$

   where $F_S(t, u(t))$ is obtained by the LDG discretization to $\nabla \cdot (b(u^n)\nabla(-\gamma \Delta u))$ and $F_N(t, u(t))$ is obtained by the remaining terms in (3.1).
3. Apply the third order ARK time discretization method to ODE (3.2) and it requires to solve three algebraic non-symmetric linear systems at each time step. What we should have in mind is that the time step Δt can not be chosen too large because of the explicit treatment of the term $\nabla \cdot ((b(u) - b(u^n))\nabla(-\gamma \Delta u))$.
4. Apply the linear MG method to solve the linear system.

3.2 The Nonlinear Scheme

For CH equation (1.1), especially for the equation with the degenerate mobility $b(u)$, we can also consider a full implicit scheme instead of the linearization technique. We first introduce the DIRK method, here we just take the third order DIRK method for an example, which is used in our numerical experiments. We apply the LDG spatial discretization [34] to the CH equation (1.1) and get an ODE

$$\begin{cases} u_t = F(t, u(t)), \quad t \in [0, T], \\ u(0) = u_0, \end{cases} \tag{3.3}$$

where $F(t, u(t))$ is a nonlinear function of u.

Starting from u^n, we give the DIRK method to calculate u^{n+1}. Following the work [1], the DIRK method is used to solve equation (3.3). They are given in the following form

$$\begin{aligned} u^{n,i} &= u^n + \Delta t \sum_{j=1}^{3} a_{ij} F(t^{n,j}, u^{n,j}), \quad i = 1, 2, 3, \\ u^{n+1} &= u^n + \Delta t \sum_{j=1}^{3} b_j F(t^{n,j}, u^{n,j}), \end{aligned} \tag{3.4}$$

where $t^{n,j} = t^n + c_j \Delta t$ and $u^{n,j}$ approximates $u(t^{n,j})$. The coefficients a_{ij}, $i = 1, 2, 3$, b_j, c_j are constrained by order of accuracy and stability considerations. For a detailed description of the method, we refer the readers to [1].

After the implicit discretization, it requires to solve three nonlinear systems of algebraic equations (3.4) at each time step. We apply the nonlinear FAS MG method to solve these nonlinear systems, which will be described in detail in "Appendix 2".

4 Local Mode Analysis of the Two-Level Algorithm

4.1 Notations

We start by introducing the basic notations to describe the general setting of a two-level or bi-grid method. Together with the family of partitions $\{\mathcal{T}_h\}_{h>0}$ used for the LDG discretization, we consider a coarse family of mesh partitions, $\{\mathcal{T}_H\}_{H>0}$ with $H > h$ and satisfying the basic assumption $\mathcal{T}_H \subset \mathcal{T}_h$. One can think $H = 2h$, since in many circumstances it will already be coarse enough. Associated to the coarse mesh partition we have the corresponding finite element space V_H^k which is defined as:

$$V_H^k = \{v \in L^2(D) : v|_D \in \mathcal{P}^k(D); \quad \forall D \in \mathcal{T}_H\}. \tag{4.1}$$

Throughout the whole description we assume the polynomial degree k is fixed.

To link functions in both spaces, we define the prolongation and restriction operators. The prolongation operator $P_{hH} : V_H \vee V_h$ is defined as the natural inclusion. The restriction operator $R_{Hh} : V_h \vee V_H$ is defined as the transpose of P_{hH} with respect to the standard L^2-inner product. That is, it is obtained by solving:

$$\sum_{D \in \mathcal{T}_H} \int_D R_{Hh}(u_h) v_H dx = \sum_{D \in \mathcal{T}_h} \int_D u_h P_{hH}(v_H) dx, \quad \forall v_H \in V_H^k. \tag{4.2}$$

We now denote S_h as a general relaxation or smoothing operator. We specify and study several choices in "Appendix 1". The basic property that S_h should have is to damp the high frequencies of the approximate solution and smooth the error. The coarse solver is defined by $A_H = R_{Hh} A_h P_{hH}$. The coarse grid correction step is to reduce the smooth components of the error that can not be reduced by the smoother.

A fairly complete description of the linear MG method and the nonlinear FAS MG method are given in "Appendices 1 and 2", respectively.

4.2 The Convergence of the Two-Grid Algorithm

In this section, we consider the Local Mode Analysis for the linearization scheme (3.1). The linear part is $\nabla \cdot (b(u^n)\nabla(-\gamma \Delta u))$, so what we do in the following is to analyze to the equation

$$\frac{u^{n+1} - u^n}{\Delta t} = \nabla \cdot (b(u^n)\nabla(-\gamma \Delta u^{n+1})). \tag{4.3}$$

The analysis for general fourth order linear equation can be found in [21]. For convenience we consider the Local Mode Analysis for the one-dimensional and two-dimensional cases. Extension to higher dimensions follows immediately by means of the tensor-product principle. For a detailed description of the Local Mode Analysis for the two-grid algorithm, we refer the readers to "Appendix 3".

We choose $b(u) = 1, \gamma = 1$ in Eq. (4.3). The convergence of the two-grid algorithm in one-dimensional and two-dimensional cases are analyzed by the Local Mode Analysis and the numerical results are shown as follows.

- One space dimension
  1) We choose $\alpha = 0.75$ with Jacobi smoother and $\alpha = 1.0$ (no damping) with Gauss–Seidel smoother for $\mathcal{P}^1$ and $\mathcal{P}^2$ approximation from Fig. 1.
  2) From Figs. 2 and 3, we can see that the Gauss–Seidel smoother has better convergence behavior than the Jacobi smoother.
  3) The MG method is not convergent for $\mathcal{P}^2$ approximation with damped Jacobi smoother according to Fig. 2.
- Two space dimension
  1) We choose $\alpha = 0.85$ with Jacobi smoother and $\alpha = 1.0$ (no damping) with Gauss–Seidel smoother for $\mathcal{P}^1$ and $\mathcal{P}^2$ approximation from Fig. 4.
  2) From Figs. 5 and 6, we can see that the Gauss–Seidel smoother has better convergence behavior than the Jacobi smoother.
  3) The MG method is not convergent for $\mathcal{P}^2$ approximation with damped Jacobi smoother according to Fig. 5.

5 Numerical Results

In this section, we perform numerical experiments of the LDG method applied to the CH equation. We use the implicit third order ARK time-marching method and the resulting linear system is solved by the linear MG method. The third order DIRK time discretization method is also applied to the CH equation and we solve the resulting nonlinear equations by the nonlinear FAS MG method. A comparison is made among these two different methods.

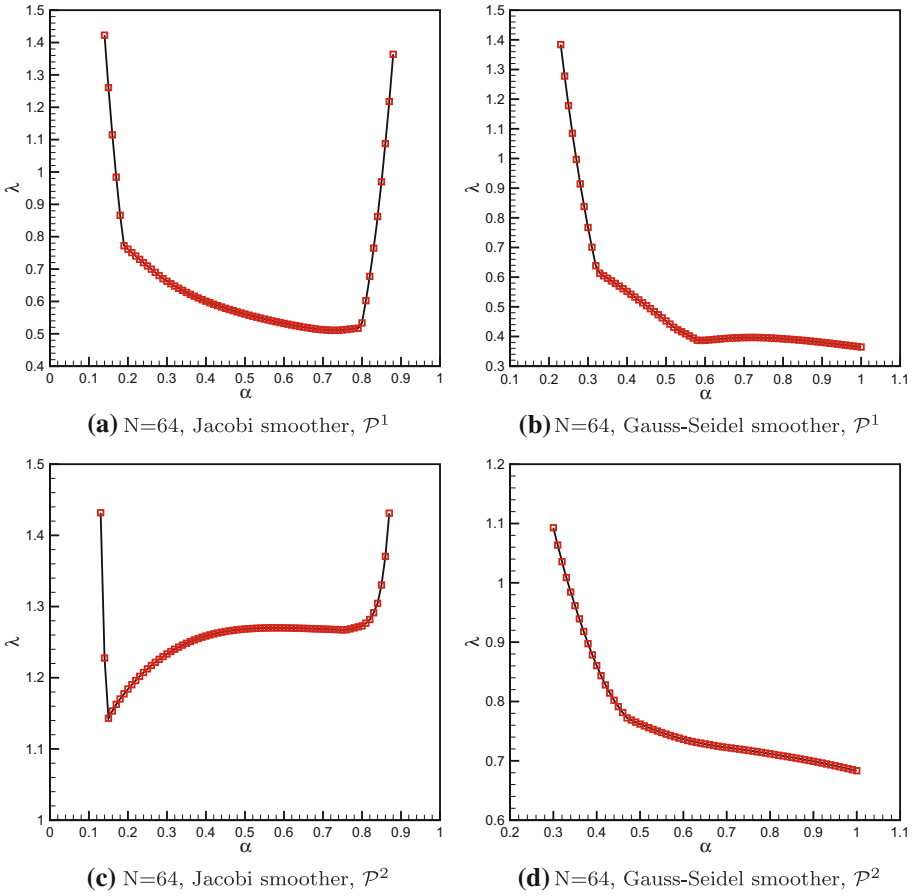


Fig. 1 The asymptotic convergence factor λ changes with the damping parameter α in one dimension

For the spatial discretization we use uniform meshes. In our numerical experiments, the number of pre- and post-relaxations is $\nu_1 = \nu_2 = 3$.

5.1 One Space Dimension

Example 1 Convex splitting scheme for degenerate mobility

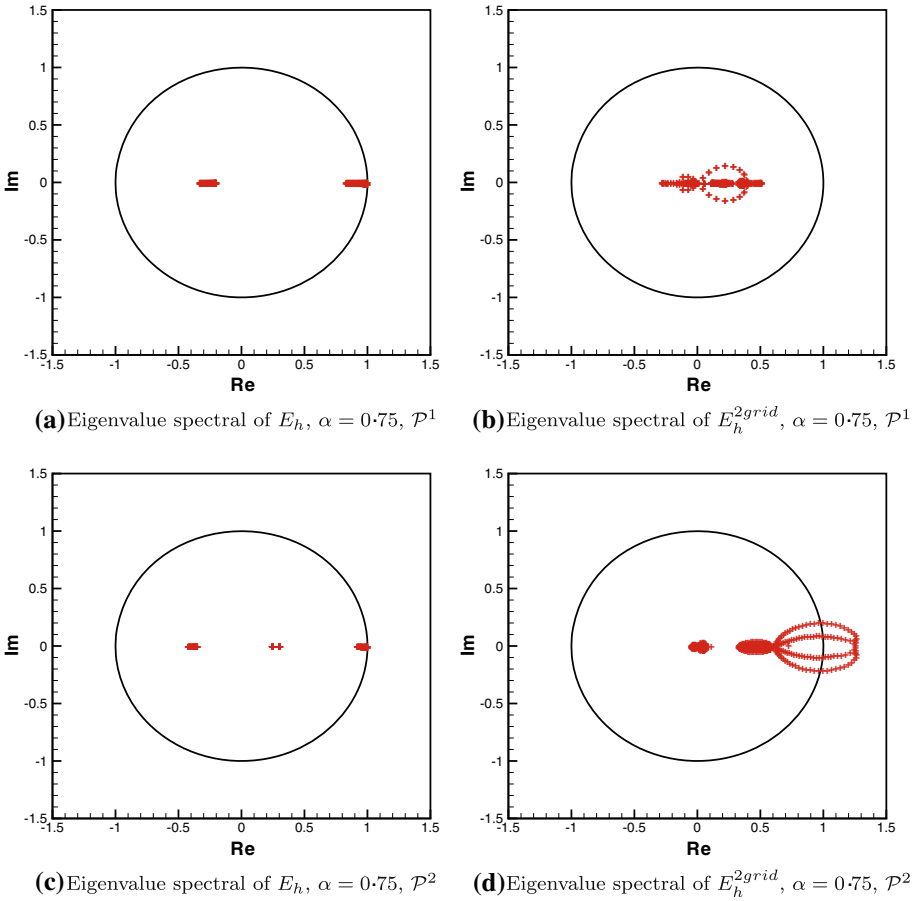
We consider the scheme (2.4) in $\Omega = (0, 2\pi)$ with degenerate mobility $b(u) = 1 - u^2$, $\gamma = 2$, $\Psi(u) = \frac{1}{4}(1 - u^2)^2$ and periodic boundary conditions. We take the exact solution of

$$u(x, t) = e^{-t} \sin(x), \tag{5.1}$$

with the source term f, where f is a given function so that (5.1) is the exact solution.

The convex splitting scheme (2.4) is the first order accuracy in time, so we just consider $\mathcal{P}^0$ approximation. The L^2 and L^∞ errors, and the numerical order of accuracy at time $t = 1.0$ with different time steps are presented in Table 1. The numerical experiments go well with the theoretical result of the unconditional energy stability for the scheme.

Example 2 ARK method for constant mobility



(a) Eigenvalue spectral of E_h, $\alpha = 0{\cdot}75$, $\mathcal{P}^1$ **(b)** Eigenvalue spectral of E_h^{2grid}, $\alpha = 0{\cdot}75$, $\mathcal{P}^1$

(c) Eigenvalue spectral of E_h, $\alpha = 0{\cdot}75$, $\mathcal{P}^2$ **(d)** Eigenvalue spectral of E_h^{2grid}, $\alpha = 0{\cdot}75$, $\mathcal{P}^2$

Fig. 2 Eigenvalue spectral of E_h and E_h^{2grid} with damped Jacobi smoother and $\mathcal{P}^1$ approximation, $\mathcal{P}^2$ approximation in one dimension

We consider the CH equation

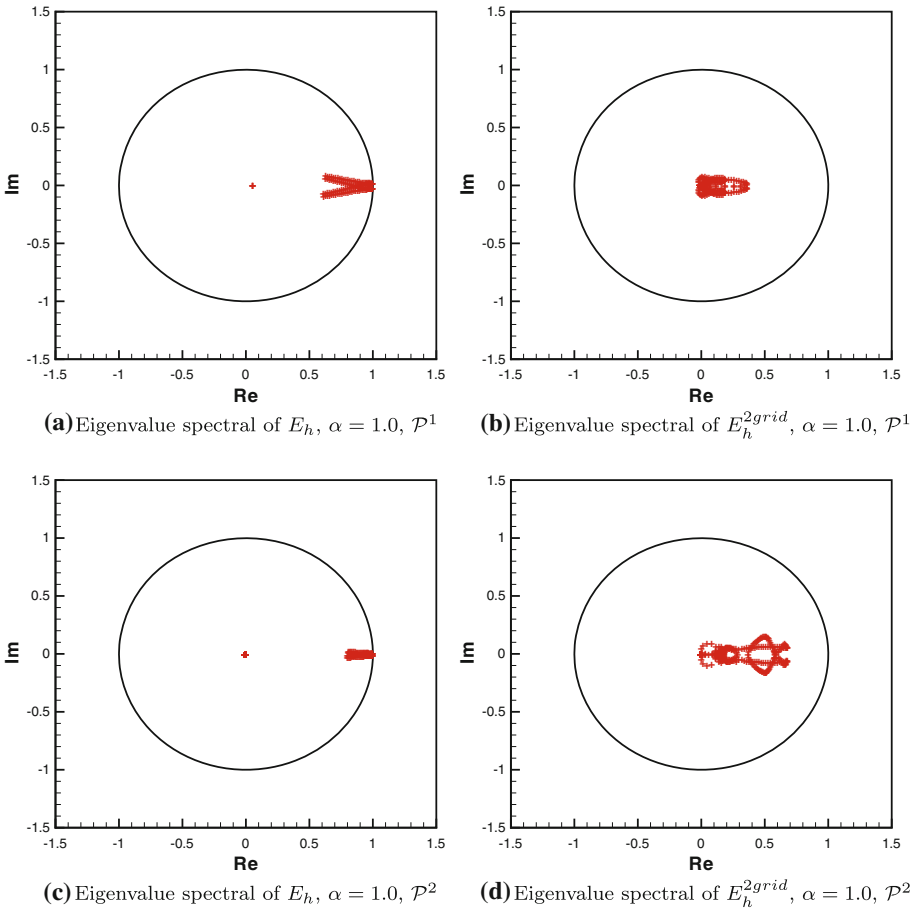
$$u_t = (b(u)(-\gamma u_{xxx} + (\Psi'(u))_x))_x, \tag{5.2}$$

with $b(u) = 1$, $\Psi(u) = \frac{3}{2}(1-u^2)$, $\gamma = 4$ in $\Omega = (0, 4\pi)$ and periodic boundary conditions. The exact solution is given by

$$u(x,t) = e^{-t}\sin(x) - e^{0.5t}\sin(0.5x). \tag{5.3}$$

The implicit third order ARK method is used in the time integration and the time step is taken as $\Delta t = 0.1\Delta x$. The L^2 and L^∞ errors, and the numerical order of accuracy at time $t = 0.5$ are presented in Table 2. We can see that the method with $\mathcal{P}^k$ elements gives a $(k+1)$-th order of accuracy.

To illustrate the superiority of the MG solver, we present the convergence rates of the method for a single time-step by using $\mathcal{P}^1$ and $\mathcal{P}^2$ elements. From Fig. 7, we can see that each iteration of the MG solver is an $O(N)$ operation. By Table 3 we see that the convergence is independent of the mesh size. We also find that the Gauss–Seidel smoother show better



(a) Eigenvalue spectral of E_h, $\alpha = 1.0$, $\mathcal{P}^1$ **(b)** Eigenvalue spectral of E_h^{2grid}, $\alpha = 1.0$, $\mathcal{P}^1$

(c) Eigenvalue spectral of E_h, $\alpha = 1.0$, $\mathcal{P}^2$ **(d)** Eigenvalue spectral of E_h^{2grid}, $\alpha = 1.0$, $\mathcal{P}^2$

Fig. 3 Eigenvalue spectral of E_h and E_h^{2grid} with Gauss–Seidel smoother and $\mathcal{P}^1$ approximation, $\mathcal{P}^2$ approximation in one dimension

convergence than the Jacobi smoother. The same information for $\mathcal{P}^2$ approximation are shown in Table 4 and Fig. 8 and we conclude that $\mathcal{P}^2$ approximation converges worse than $\mathcal{P}^1$ approximation.

Example 3 ARK and DIRK methods for constant mobility

We consider the CH equation

$$u_t = (b(u)(-\gamma u_{xxx} + (\Psi'(u))_x))_x - f(x,t), \tag{5.4}$$

with $b(u) = 1$, $\Psi(u) = \frac{1}{4}(1-u^2)^2$, $\gamma = 2$ in $\Omega = (0, 4\pi)$ and periodic boundary conditions. We take the exact solution

$$u(x,t) = e^{-t}\sin(x) - e^{0.5t}\cos(0.5x). \tag{5.5}$$

The implicit third order ARK method and the third order DIRK method are used in the time integration with the time step $\Delta t = 0.05\Delta x$ and $\Delta t = 0.1\Delta x$ respectively. The L^2 and L^∞ errors, and the numerical order of accuracy at time $t = 0.5$ are presented in Table 5.

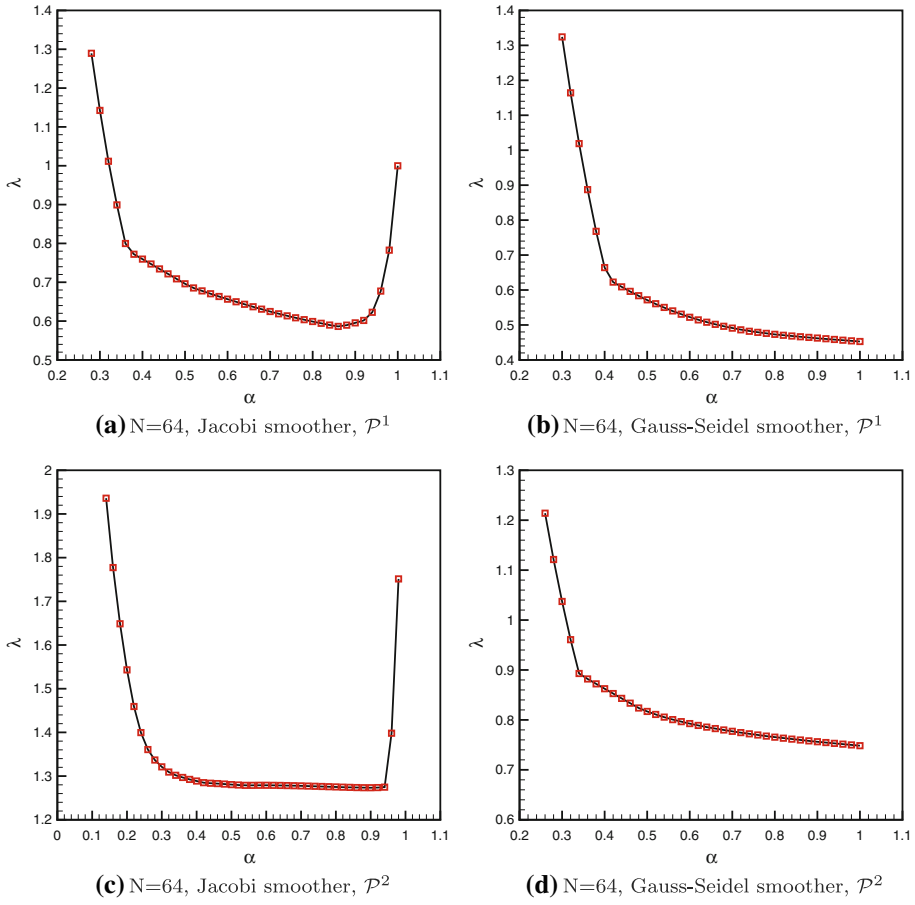


Fig. 4 The asymptotic convergence factor λ changes with the damping parameter α in two dimension

We can see that both methods with $\mathcal{P}^k$ elements give a $(k+1)$-th order of accuracy. The CPU time with these two different time discretization methods are shown in Fig. 9. We can see that the ARK method with linear MG solver cost less CPU time. The reason is that for constant mobility $b(u)$, we do not need the linearization technique, so the ARK method is more efficient.

Example 4 ARK and DIRK methods for degenerate mobility

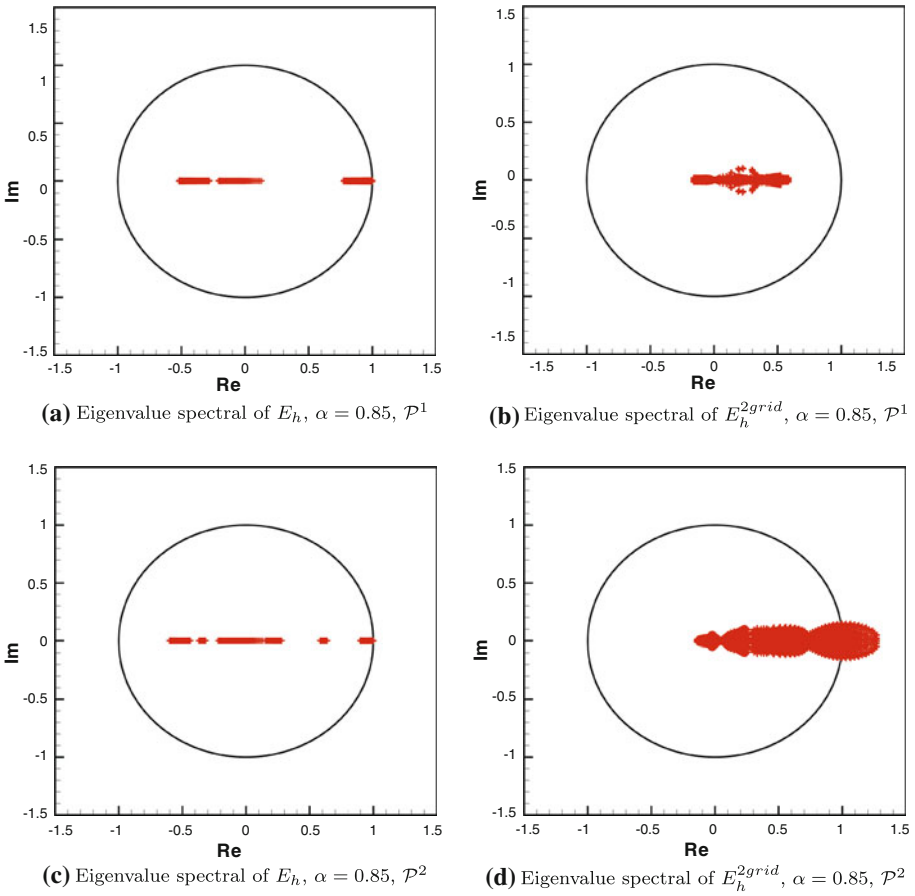
We consider the CH equation

$$u_t = (b(u)(-\gamma u_{xxx} + (\Psi'(u))_x))_x - f(x,t), \tag{5.6}$$

with $b(u) = 1-u^2$, $\Psi(u) = \frac{1}{2}(1-u^2)$, $\gamma = 2$ in $\Omega = (0, 2\pi)$ and periodic boundary conditions. We take the exact solution

$$u(x,t) = e^{-t}\sin(x). \tag{5.7}$$

The implicit third order ARK method and the third order DIRK method are used in the time integration with the time step $\Delta t = 0.001\Delta x$ and $\Delta t = 0.07\Delta x^2$ respectively. The L^2 and L^∞ errors, and the numerical order of accuracy at time $t = 0.5$ are presented in Table 6.



(a) Eigenvalue spectral of E_h, $\alpha = 0.85$, $\mathcal{P}^1$

(b) Eigenvalue spectral of E_h^{2grid}, $\alpha = 0.85$, $\mathcal{P}^1$

(c) Eigenvalue spectral of E_h, $\alpha = 0.85$, $\mathcal{P}^2$

(d) Eigenvalue spectral of E_h^{2grid}, $\alpha = 0.85$, $\mathcal{P}^2$

Fig. 5 Eigenvalue spectral of E_h and E_h^{2grid} with damped Jacobi smoother and $\mathcal{P}^1$ approximation, $\mathcal{P}^2$ approximation in two dimension

We can see that both methods with $\mathcal{P}^k$ elements give a $(k+1)$-th order of accuracy. The CPU time with the two different time discretization methods are shown in Fig. 10. We can see that the DIRK method with nonlinear FAS MG solver cost less CPU time. The reason is that for degenerate mobility $b(u)$, the linearization technique is needed and the time step can not be chosen larger because of the explicit treatment of the term $\nabla \cdot ((b(u) - b(u^n))\nabla(-\gamma \Delta u))$, so under this situation, the DIRK method is more efficient.

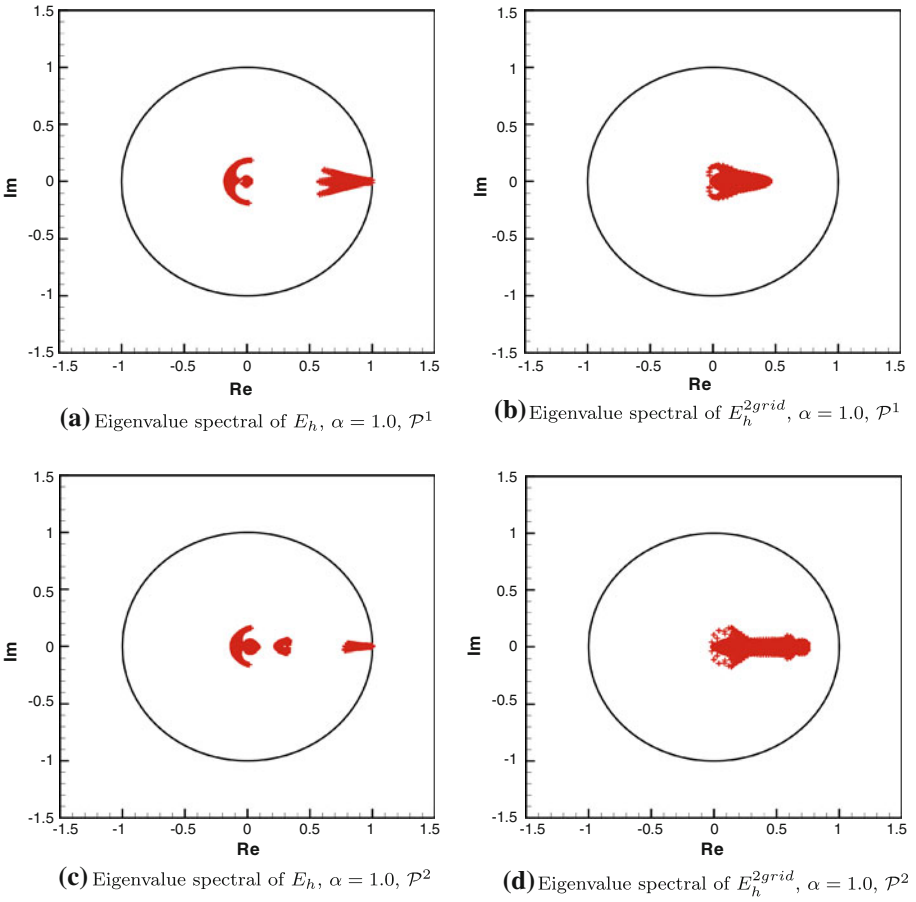
5.2 Two Space Dimension

Example 5 Convex splitting scheme for degenerate mobility

We consider the scheme (2.4) in $\Omega = (0, 2\pi) \times (0, 2\pi)$ with degenerate mobility $b(u) = 1 - u^2$, $\gamma = 1$, $\Psi(u) = \frac{1}{4}(1 - u^2)^2$ and periodic boundary conditions. We take the exact solution of

$$u(x, y, t) = e^{-2t} \sin(x) \sin(y), \tag{5.8}$$

with the source term f, where f is a given function so that (5.8) is the exact solution.



(a) Eigenvalue spectral of E_h, $\alpha = 1.0$, $\mathcal{P}^1$

(b) Eigenvalue spectral of E_h^{2grid}, $\alpha = 1.0$, $\mathcal{P}^1$

(c) Eigenvalue spectral of E_h, $\alpha = 1.0$, $\mathcal{P}^2$

(d) Eigenvalue spectral of E_h^{2grid}, $\alpha = 1.0$, $\mathcal{P}^2$

Fig. 6 Eigenvalue spectral of E_h and E_h^{2grid} with Gauss–Seidel smoother and $\mathcal{P}^1$ approximation, $\mathcal{P}^2$ approximation in two dimension

The convex splitting scheme (2.4) is the first order accuracy in time, so we just consider $\mathcal{P}^0$ approximation. The L^2 and L^∞ errors, and the numerical order of accuracy at time $t = 1.0$ with different time steps are presented in Table 7. The numerical experiments go well with the theoretical result of the unconditional energy stability for the scheme.

To illustrate the superiority of the FAS MG solver, we present the convergence rates of the method for a single time-step by using $\mathcal{P}^1$ and $\mathcal{P}^2$ elements. From Fig. 11, we can see that each iteration of the FAS MG solver is an $O(N)$ operation. From Table 8 we see that the convergence is independent of the mesh size. It is clear that the Gauss–Seidel smoother converges much faster than the Jacobi smoother. The same information for $\mathcal{P}^2$ approximation are shown in Table 9 and Fig. 12 and we conclude that $\mathcal{P}^1$ approximation converges better than $\mathcal{P}^2$ approximation.

Example 6 ARK and DIRK methods for constant mobility

We consider the CH equation

$$u_t = \nabla \cdot (b(u)\nabla(-\gamma \Delta u + \Psi'(u))), \tag{5.9}$$

Table 1 Accuracy test for the scheme (2.4) with the exact solution (5.1)

N	$\Delta t = 0.1\Delta x$				$\Delta t = 1.0\Delta x$			
	L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
16	1.19E–01	–	1.05E–01	–	3.45E–01	–	2.07E–01	–
32	5.43E–02	1.14	5.03E–02	1.07	2.02E–01	0.77	1.18E–01	0.80
64	2.59E–02	1.06	2.46E–02	1.02	1.09E–01	0.89	6.33E–02	0.90
128	1.27E–02	1.03	1.22E–02	1.01	5.68E–02	0.94	3.28E–02	0.95
	$\Delta t = 5.0\Delta x$				$\Delta t = 10.0\Delta x$			
	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
16	5.78E–01	–	3.22E–01	–	5.78E–01	–	3.22E–01	–
32	4.98E–01	0.21	2.78E–01	0.21	5.17E–01	0.16	2.89E–01	0.15
64	3.55E–01	0.49	1.96E–01	0.50	4.84E–01	0.10	2.64E–01	0.13
128	2.24E–01	0.67	1.23E–01	0.67	3.52E–01	0.47	1.94E–01	0.44

Table 2 Accuracy test for the CH equation (5.2) with the exact solution (5.3)

	N	L^2 error	Order	L^∞ error	Order
$\mathcal{P}^0$	16	5.86E–01	–	5.11E–01	–
	32	2.75E–01	1.08	2.47E–01	1.04
	64	1.35E–01	1.02	1.22E–01	1.01
	128	6.75E–02	1.00	6.13E–01	1.00
$\mathcal{P}^1$	16	8.81E–02	–	1.09E–01	–
	32	2.17E–02	2.02	2.71E–02	2.01
	64	5.40E–03	2.00	6.77E–03	2.00
	128	1.35E–03	2.00	1.69E–03	2.00
$\mathcal{P}^2$	16	4.41E–03	–	5.93E–03	–
	32	5.40E–04	3.03	7.66E–04	2.95
	64	6.72E–05	3.01	9.65E–05	2.99
	128	8.38E–06	3.00	1.20E–05	3.00

with $b(u) = 1, \Psi(u) = 600(u\ln u + (1-u)\ln(1-u)) + 1800u(1-u), \gamma = 1$. The initial condition is

$$u_0(x) = \begin{cases} 0.71, & \mathbf{x} \in \Omega_1, \\ 0.69, & \mathbf{x} \in \Omega_2, \end{cases} \tag{5.10}$$

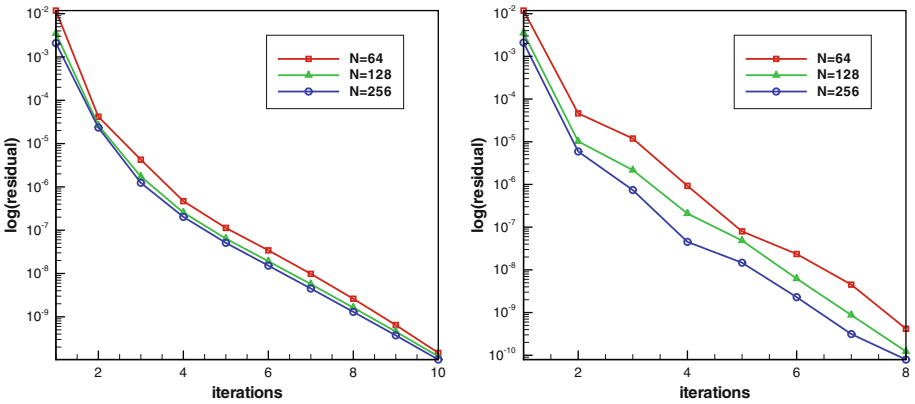
where the square domain

$$\Omega = (-0.5, 0.5) \times (-0.5, 0.5), \quad \Omega_1 = (-0.2, 0.2) \times (-0.2, 0.2), \quad \Omega_2 = \Omega - \Omega_1.$$

The boundary conditions are

$$\frac{\partial u}{\partial \nu} = b(u)\nabla w \cdot \nu = 0, \quad \text{on} \quad \partial\Omega. \tag{5.11}$$

The implicit third order ARK method and the DIRK method are used in the time integration. We use the $\mathcal{P}^1$ and $\mathcal{P}^2$ elements on the meshes with 128×128 cells, respectively. The time



(a) damped Jacobi smoother with $\alpha = 0.75$ **(b)** Gauss-Seidel smoother, no damping

Fig. 7 Convergence rates of MG solver for $\mathcal{P}^1$ approximation in one space dimension

Table 3 The number of MG iterations required to reduce the norm of the residual below the tolerance $\epsilon = 1.0 \times 10^{-9}$ for $\mathcal{P}^1$ approximation in one space dimension

Δx	Jacobi smoother	Gauss–Seidel smoother
$4\pi/32$	8	7
$4\pi/64$	9	8
$4\pi/128$	9	7
$4\pi/256$	9	7

Table 4 The number of MG iterations required to reduce the norm of the residual below the tolerance $\epsilon = 1.0 \times 10^{-9}$ for $\mathcal{P}^2$ approximation with Gauss–Seidel smoother in one space dimension

Δx	$4\pi/16$	$4\pi/32$	$4\pi/64$
number of MG iterations	9	9	16

step is taken as $\Delta t = 0.01\Delta x^2$. The contours at $t = 8 \times 10^{-5}$ are shown in Fig. 13. We can see that these two methods show the same phenomenon and the numerical results compare very well with the numerical calculations in [34].

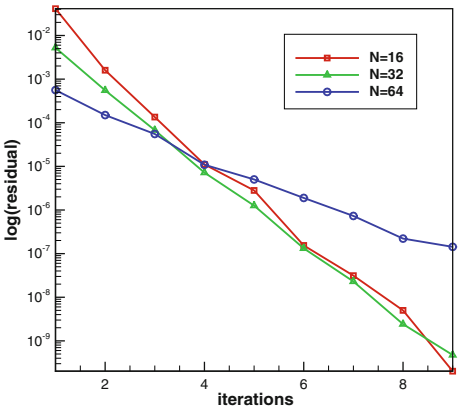


Fig. 8 Convergence rates of MG solver for $\mathcal{P}^2$ approximation with Gauss-Seidel smoother in one space dimension

Table 5 Accuracy test for the CH equation (5.4) with the exact solution (5.5)

	N	Linear MG				Nonlinear FAS MG			
		L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
$\mathcal{P}^0$	16	1.01E–00	–	9.61E–01	–	1.01E–00	–	9.61E–01	–
	32	5.08E–01	0.99	4.88E–01	0.97	5.08E–01	0.99	4.88E–01	0.97
	64	2.54E–01	1.00	2.44E–01	1.00	2.54E–01	1.00	2.44E–01	1.00
	128	1.27E–01	1.00	1.22E–01	1.00	1.27E–01	1.00	1.22E–01	1.00
$\mathcal{P}^1$	16	1.24E–01	–	1.51E–01	–	1.25E–01	–	1.51E–01	–
	32	3.11E–02	1.99	3.75E–02	2.00	3.13E–02	1.99	3.74E–02	2.01
	64	7.66E–03	2.02	9.67E–03	1.95	7.83E–03	2.00	9.47E–03	1.98
	128	1.92E–03	1.99	2.41E–03	2.00	1.95E–03	2.00	2.36E–03	2.00
$\mathcal{P}^2$	16	2.58E–02	–	3.36E–02	–	2.56E–02	–	3.61E–02	–
	32	3.23E–03	2.99	4.27E–03	2.97	3.23E–03	2.98	4.48E–03	3.01
	64	4.29E–04	2.91	5.38E–04	2.98	4.05E–04	2.99	5.57E–04	3.00
	128	5.31E–05	3.01	6.47E–05	3.05	5.08E–05	2.99	6.97E–05	3.00

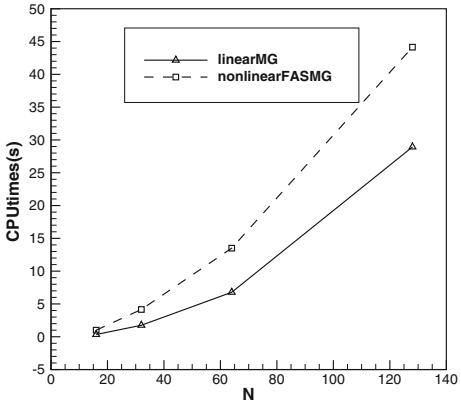


Fig. 9 The CPU time of the linear scheme and the nonlinear scheme for the CH equation (5.4) with $\mathcal{P}^1$ approximation

Example 7 ARK and DIRK methods for degenerate mobility

We consider the CH equation

$$u_t = \nabla \cdot (b(u)\nabla(-\gamma \Delta u + \Psi'(u))), \tag{5.12}$$

with $b(u) = u(1-u), \Psi(u) = 3000(u\ln u + (1-u)\ln(1-u)) + 9000u(1-u), \gamma = 1$ in $\Omega = (-0.5, 0.5) \times (-0.5, 0.5)$. The boundary conditions are taken as (5.11). The initial condition u_0 is a random perturbation of uniform state $u = 0.63$ with a fluctuation no larger than 0.05.

The implicit third order ARK method and the DIRK method are used in the time integration. We use the $\mathcal{P}^1$ and $\mathcal{P}^2$ elements on the meshes with 64×64 cells. Figure 14 shows the evolution of the concentration field. The concentration evolution can basically be categorized in two phases, the phase separation (before $t = 8 \times 10^{-6}$) stage ($\Delta t = 2 \times 10^{-7}$) and the coarsening process (from $t = 8 \times 10^{-6}$ onwards) stage ($\Delta t = 0.01 \Delta x^2$). We can see

Table 6 Accuracy test for the CH equation (5.6) with the exact solution (5.7)

	N	Linear MG				Nonlinear FAS MG			
		L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
$\mathcal{P}^0$	16	2.69E–01	–	2.52E–01	–	2.69E–01	–	2.52E–01	–
	32	1.34E–01	1.00	1.26E–01	1.00	1.34E–01	1.00	1.26E–01	1.00
	64	6.71E–02	1.00	6.31E–02	1.00	6.71E–02	1.00	6.31E–02	1.00
	128	3.35E–02	1.00	3.15E–02	1.00	3.35E–02	1.00	3.15E–02	1.00
$\mathcal{P}^1$	16	1.99E–02	–	2.77E–02	–	1.99E–02	–	2.77E–02	–
	32	4.92E–03	2.01	6.86E–03	2.01	4.93E–03	2.01	6.86E–03	2.01
	64	1.22E–03	2.01	1.70E–03	2.01	1.22E–03	2.01	1.70E–03	2.01
	128	3.06E–04	1.99	4.26E–04	1.99	3.06E–04	1.99	4.26E–04	2.00
$\mathcal{P}^2$	16	2.21E–03	–	3.49E–03	–	3.29E–03	–	5.21E–03	–
	32	2.77E–04	2.99	4.39E–04	2.99	4.13E–04	2.99	6.55E–04	2.99
	64	3.47E–05	2.99	5.49E–05	3.00	5.17E–05	3.00	8.20E–05	3.00
	128	4.36E–06	2.99	6.92E–06	2.99	6.47E–06	3.00	1.02E–05	3.00

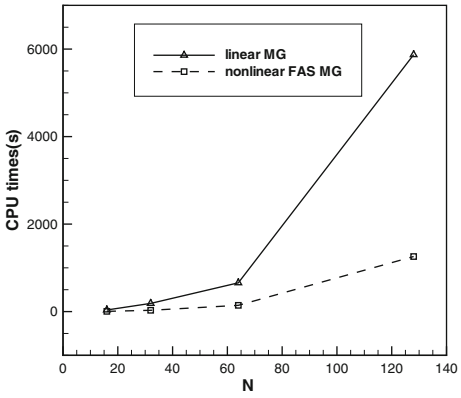


Fig. 10 The CPU time of the linear scheme and the nonlinear scheme for the CH equation (5.6) with $\mathcal{P}^1$ approximation

that these two methods show the same phenomenon and the numerical results show similar patterns with the numerical calculations in Gomez et al. [20].

5.3 Three Space Dimension

Example 8 Convex splitting scheme with degenerate mobility

We consider the scheme (2.4) in $\Omega = (0, 2\pi) \times (0, 2\pi) \times (0, 2\pi)$ with $b(u) = 1 - u^2$, $\gamma = 1$, $\Psi(u) = \frac{1}{4}(1 - u^2)^2$ and periodic boundary conditions. We take the exact solution of

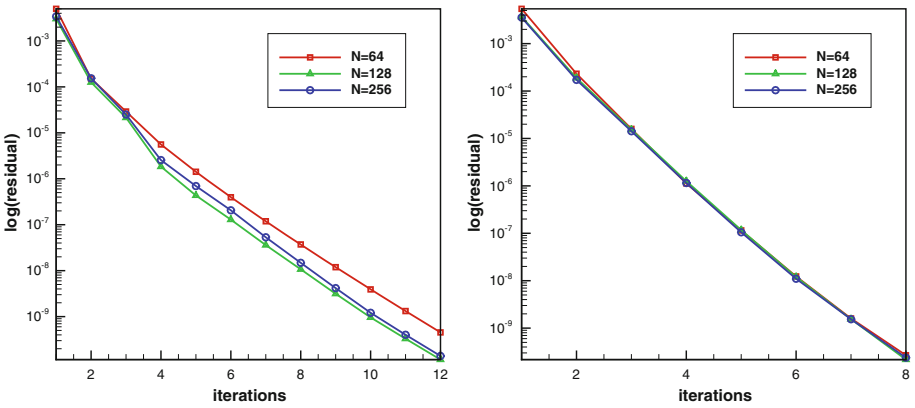
$$u(x, y, z, t) = e^{-3t} \sin(x) \sin(y) \sin(z), \tag{5.13}$$

with the source term f, where f is a given function so that (5.13) is the exact solution.

The convex splitting scheme (2.4) is the first order accuracy in time, so we just consider $\mathcal{P}^0$ approximation. The L^2 and L^∞ errors, and the numerical order of accuracy at time $t = 1.0$

Table 7 Accuracy test for the scheme (2.4) with the exact solution (5.8)

N	$\Delta t = 0.1\Delta x$				$\Delta t = 1.0\Delta x$			
	L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
16	2.23E–02	–	4.40E–02	–	1.29E–01	–	2.50E–01	–
32	9.81E–03	1.18	1.95E–02	1.17	7.72E–02	0.75	1.52E–01	0.71
64	4.56E–03	1.10	9.10E–03	1.10	4.05E–02	0.93	8.08E–02	0.92
128	2.20E–03	1.05	4.39E–03	1.04	2.06E–02	0.97	4.13E–02	0.97
	$\Delta t = 5.0\Delta x$				$\Delta t = 10.0\Delta x$			
	L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
16	2.53E–01	–	4.75E–01	–	2.53E–01	–	4.75E–01	–
32	2.36E–01	0.10	4.59E–01	0.05	2.47E–01	0.04	4.65E–01	0.03
64	1.55E–01	0.61	3.07E–01	0.58	2.34E–01	0.08	4.58E–01	0.02
128	9.14E–02	0.76	1.82E–01	0.75	1.54E–01	0.60	3.07E–01	0.58



(a) damped Jacobi smoother with $\alpha = 0.5$ **(b)** Gauss-Seidel smoother, no damping

Fig. 11 Convergence rates of FAS MG solver for $\mathcal{P}^1$ approximation in two space dimension

Table 8 The number of FAS MG iterations required to reduce the norm of the residual below the tolerance $\epsilon = 1.0 \times 10^{-9}$ for $\mathcal{P}^1$ approximation in two space dimension

Δx	Jacobi smoother	Gauss–Seidel smoother
$2\pi/32$	13	8
$2\pi/64$	12	8
$2\pi/128$	10	8
$2\pi/256$	11	8

with different time steps are presented in Table 10. The numerical experiments go well with the theoretical result of the unconditional energy stability for the scheme.

To illustrate the superiority of the FAS MG solver, we present the convergence rates of the method for a single time-step by using $\mathcal{P}^1$ and $\mathcal{P}^2$ elements. From Fig. 15, we can see that each iteration of the FAS MG solver is an $O(N)$ operation. From Table 11 we see that

Table 9 The number of FAS MG iterations required to reduce the norm of the residual below the tolerance $\epsilon = 1.0 \times 10^{-9}$ for $\mathcal{P}^2$ approximation with Gauss–Seidel smoother in two space dimension

Δx	$2\pi/16$	$2\pi/32$	$2\pi/64$
number of MG iterations	9	11	16

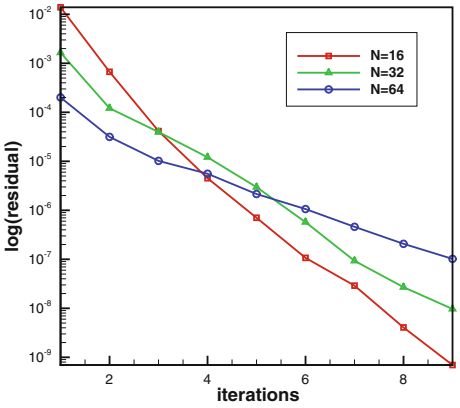


Fig. 12 Convergence rates of FAS MG solver for $\mathcal{P}^2$ approximation with Gauss-Seidel smoother in two space dimension

the number of iterations is independent of the mesh size and the Gauss–Seidel smoother has better convergence behavior than the Jacobi smoother. The convergence rates for $\mathcal{P}^2$ approximation is shown in Fig. 16 and each iteration of the FAS MG solver is also an $O(N)$ operation.

Example 9 ARK and DIRK methods for constant mobility

We consider the CH equation

$$u_t = \nabla \cdot (b(u)\nabla(-\gamma \Delta u + \Psi'(u))), \tag{5.14}$$

with $b(u) = 1, \Psi(u) = 600(u\ln u + (1-u)\ln(1-u)) + 1800u(1-u), \gamma = 1$. The initial condition is

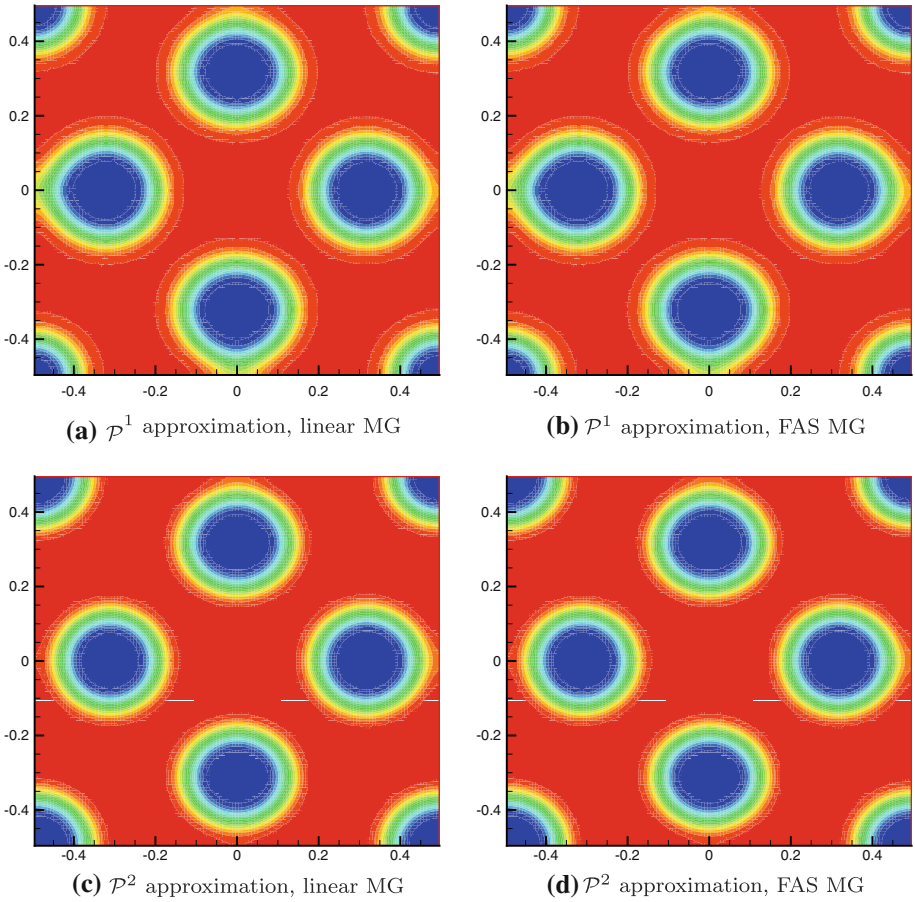
$$u_0(x) = \begin{cases} 0.71, & \mathbf{x} \in \Omega_1, \\ 0.69, & \mathbf{x} \in \Omega_2, \end{cases} \tag{5.15}$$

where the square domain

$$\Omega = (-0.5, 0.5) \times (-0.5, 0.5) \times (-0.5, 0.5),$$
$$\Omega_1 = (-0.2, 0.2) \times (-0.2, 0.2) \times (-0.2, 0.2), \quad \Omega_2 = \Omega - \Omega_1.$$

The boundary conditions are (5.11).

The implicit third order ARK method and the DIRK method are used in the time integration. We use the $\mathcal{P}^0$ and $\mathcal{P}^1$ elements on the meshes with $64 \times 64 \times 64$ cells, respectively. The time step is taken as $\Delta t = 2 \times 10^{-7}$. The contours at $t = 8 \times 10^{-5}$ are shown in Fig. 17. The numerical results show the similar phenomenon as in the two-dimensional case and we can conclude that the MG solver is efficient for the three-dimensional problems.



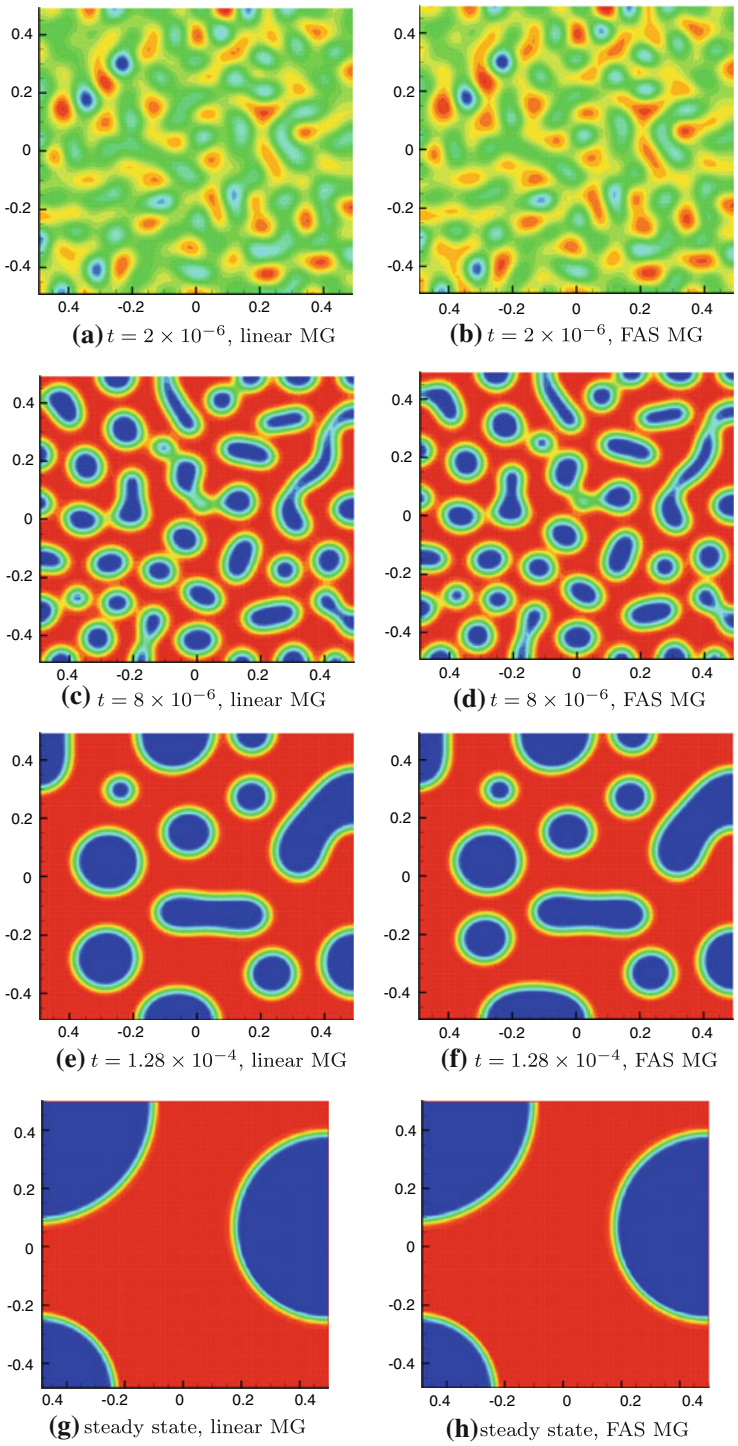
(a) $\mathcal{P}^1$ approximation, linear MG **(b)** $\mathcal{P}^1$ approximation, FAS MG

(c) $\mathcal{P}^2$ approximation, linear MG **(d)** $\mathcal{P}^2$ approximation, FAS MG

Fig. 13 The contours of $u(\mathbf{x}, t)$ for equation (5.9), $\mathcal{P}^1$ and $\mathcal{P}^2$ elements on the uniform mesh with 128×128 cells

6 Concluding Remarks

In this paper, we have presented the linear MG solver and the nonlinear FAS MG solver for the linear and nonlinear algebraic systems arising from the LDG spatial discretization and implicit time marching methods. We have studied the CH equation in one, two and three dimensions with a constant mobility and degenerate mobility. The unconditional stability of the discrete energy is proved for the CH equation with a special homogeneous free energy density $\Psi(u) = \frac{1}{4}(1-u^2)^2$ based on the convex splitting method. The convergence of the linearization scheme is studied numerically by the Local Mode Analysis. Numerical experiments show that the ARK method and the DIRK method are efficient implicit time marching methods comparing to the explicit method for the CH equation. In addition, the linear MG solver and the nonlinear FAS MG solver are efficient solvers that can solve the linear and nonlinear algebraic systems with the number of iterations independent of the problem size. The optimal or sub-optimal complexity of the MG solver for $\mathcal{P}^1$ and $\mathcal{P}^2$ approximation are numerically shown. For $\mathcal{P}^2$ approximation, we see the complexity of the



(a) $t = 2 \times 10^{-6}$, linear MG **(b)** $t = 2 \times 10^{-6}$, FAS MG

(c) $t = 8 \times 10^{-6}$, linear MG **(d)** $t = 8 \times 10^{-6}$, FAS MG

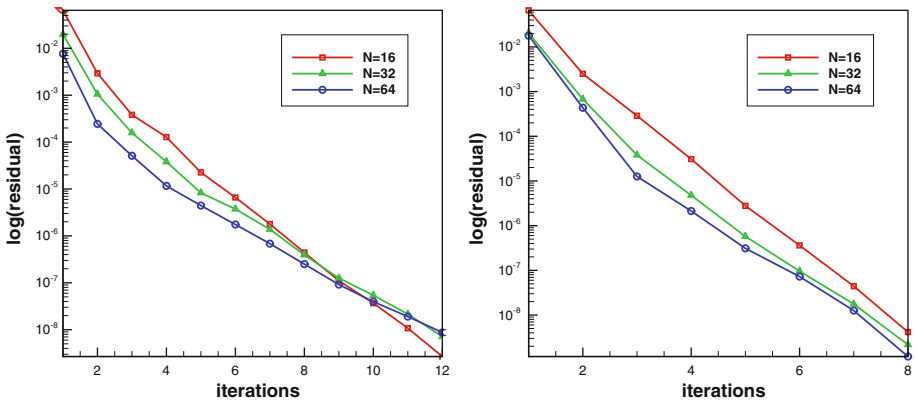
(e) $t = 1.28 \times 10^{-4}$, linear MG **(f)** $t = 1.28 \times 10^{-4}$, FAS MG

(g) steady state, linear MG **(h)** steady state, FAS MG

Fig. 14 The contours of $u(\mathbf{x}, t)$ for equation (5.12) at different time with $\mathcal{P}^2$ elements on the uniform mesh with 64×64 cells

Table 10 Accuracy test for the scheme (2.4) with the exact solution (5.13)

N	$\Delta t = 0.1\Delta x$				$\Delta t = 1.0\Delta x$			
	L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
8	2.35E–02	–	6.06E–02	–	1.29E–01	–	2.98E–01	–
16	9.30E–03	1.33	2.48E–02	1.28	6.95E–02	0.87	1.85E–01	0.69
32	4.22E–03	1.13	1.12E–02	1.14	3.52E–02	0.98	9.89E–02	0.90
64	2.00E–03	1.07	5.33E–03	1.07	1.54E–02	1.19	4.35E–02	1.18
	$\Delta t = 5.0\Delta x$				$\Delta t = 10.0\Delta x$			
	L^2 error	Order	L^∞ error	Order	L^2 error	Order	L^∞ error	Order
8	2.23E–01	–	5.10E–01	–	2.23E–01	–	5.10E–01	–
16	2.11E–01	–	5.37E–01	–	2.11E–01	–	5.37E–01	–
32	1.86E–01	0.18	5.07E–01	0.08	2.07E–01	–	5.33E–01	–
64	9.42E–02	0.98	2.61E–01	0.96	1.85E–01	0.16	5.08E–01	0.07



(a) damped Jacobi smoother with $\alpha = 0.9$ **(b)** Gauss-Seidel smoother, no damping

Fig. 15 Convergence rates of FAS MG solver for $\mathcal{P}^1$ approximation in three space dimension

Table 11 The number of FAS MG iterations required to reduce the norm of the residual below the tolerance $\epsilon = 1.0 \times 10^{-9}$ for $\mathcal{P}^1$ approximation in three space dimension

Δx	Jacobi smoother	Gauss–Seidel smoother
$2\pi/16$	13	9
$2\pi/32$	14	9
$2\pi/64$	15	9

MG solver is not always optimal. The main possible reason is that the condition number of the discretization matrix is extremely large for high order spatial discretization. Therefore, it is necessary to introduce a preconditioner such that the discretization matrix is well conditioned. We leave this topic to our future work.

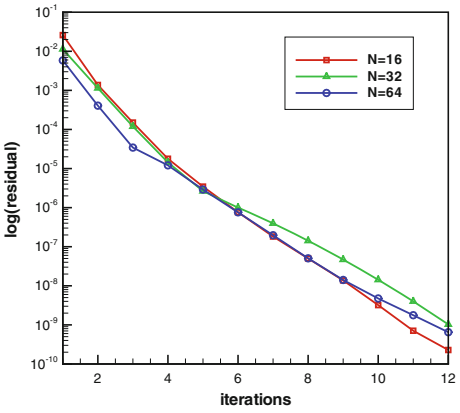
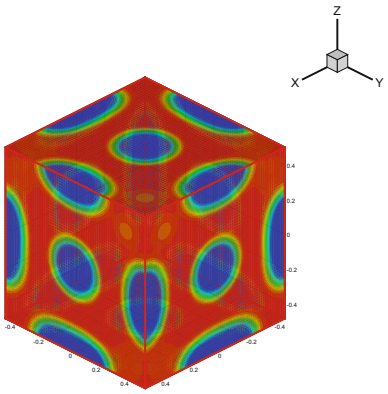
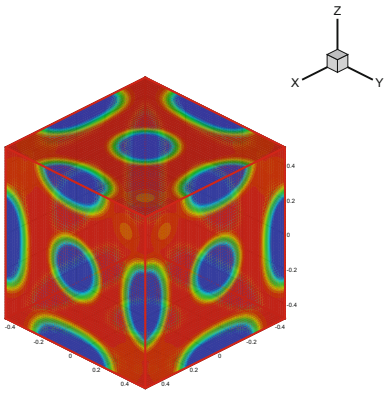


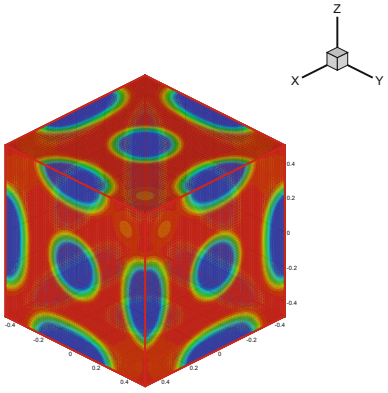
Fig. 16 Convergence rates of FAS MG solver for $\mathcal{P}^2$ approximation with Gauss-Seidel smoother in three space dimension



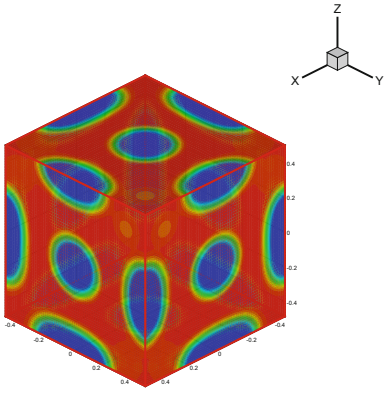
(a) $\mathcal{P}^0$, $64 \times 64 \times 64$, linear MG



(b) $\mathcal{P}^0$, $64 \times 64 \times 64$, FAS MG



(c) $\mathcal{P}^1$, $64 \times 64 \times 64$, linear MG



(d) $\mathcal{P}^1$, $64 \times 64 \times 64$, FAS MG

Fig. 17 The contours of $u(\mathbf{x}, t)$ for equation (5.14), $\mathcal{P}^0$ and $\mathcal{P}^1$ elements on the uniform mesh with $64 \times 64 \times 64$ cells

Table 12 The possible choices of the smoother operator

smoother	Jacobi	Gauss–Seidel	damped Jacobi	damped Gauss–Seidel
S_h	D_h^{-1}	$(D_h + L_h)^{-1}$	αD_h^{-1}	$\alpha(D_h + L_h)^{-1}$

7 Appendix 1: The Linear MG Solver

The linear scheme requires the solution at each time step of algebraic non-symmetric linear systems i.e.

$$A_h u_h = f_h. \tag{7.1}$$

The MG method is used to solve the system and the main points of the algorithm is the bi-grid cycle. Following [9], we can formulate the bi-grid cycle as follows:

Algorithm 1: Two-grid Cycle Starting with an initial approximation, say u^0_{PRE}:

1. **Pre-relaxation:** apply ν_1 pre-relaxation sweeps: for $m = 1 \ldots, \nu_1$, solve
$$u^m_{PRE} = u^{m-1}_{PRE} + S_h(f_h - A_h u^{m-1}_{PRE}),$$
2. **Coarse-gird correction:** update the solution $u^{\nu_1}_{PRE}$ by a coarse-grid correction step, solve the problem once on coarse grid
$$A_H v_H = R_{Hh}(f_h - A_h u^{\nu_1}_{PRE}),$$
and set $u_{CG} = u^{\nu_1}_{PRE} + P_{hH}(v_H)$.
3. **Post-relaxation:** starting with u_{CG}, apply ν_2 post-relaxation sweeps, that is, set $u^0_{POST} = u_{CG}$ and for $m = 1 \ldots, \nu_2$, solve
$$u^m_{POST} = u^{m-1}_{POST} + S_h(f_h - A_h u^{m-1}_{POST}).$$

The integers ν_1 and ν_2 are parameters in the scheme that control the number of relaxation sweeps before and after visiting the coarse grid. ν_1 and ν_2 are called the number of pre- and post- relaxations, respectively.

Solving the coarse grid problem at the second step of the above algorithm could be done again with the two-level algorithm. Hence, the V-cycle multi-level algorithm in terms of the two-level algorithm is defined by applying the two-level algorithm recursively.

Due to the block structure of A_h, we focus only on very simple block-relaxation. In particular, we decompose A_h into a strict block-lower, a block-diagonal, and a strict block-upper matrix, i.e.

$$A_h = L_h + D_h + U_h. \tag{7.2}$$

Table 12 shows some possible choices of the smoother operator.

8 Appendix 2: The Nonlinear FAS MG Solver

The nonlinear scheme (3.4) requires the solution at each time step of three algebraic nonlinear systems, i.e.

$$N_h(u_h) = f_h, \tag{8.1}$$

and the nonlinear FAS MG method is introduced to solve the system. Following [29], we illustrate the FAS bi-grid cycle as follows:

Algorithm 2: Two-grid Cycle Starting with an initial approximation, say u_h^m:

1. **Pre-relaxation:** apply ν_1 pre-relaxation sweeps to (8.1) with u_h^m the initial approximation and obtain $\bar{u}_h^m$.
2. **Coarse-gird correction:** compute the coarse level initial iterate

$$\bar{u}_H^m = R_{Hh}\bar{u}_h^m, \tag{8.2}$$

update the solution $\bar{u}_h^m$ by a coarse-grid correction step, solve the problem once on coarse grid

$$N_H(v_H) = N_H(\bar{u}_H^m) + R_{Hh}(f_h - N_h(\bar{u}_h^m)) \tag{8.3}$$

and set $u_h^{CG} = \bar{u}_h^m + P_{hH}(v_H - \bar{u}_H^m)$.
3. **Post-relaxation:** starting with u_h^{CG}, apply ν_2 post-relaxation sweeps to (8.1) and obtain u_h^{m+1}.

Solving the coarse grid problem at the second step of the above algorithm could be done again with the two-level algorithm. Hence, the nonlinear FAS MG algorithm in terms of the two-level algorithm is defined by applying the two-level algorithm recursively.

9 Appendix 3: The Local Mode Analysis

In [10], the author considered a general framework for performing the Local Mode Analysis for analyzing the convergence of two-level or bi-grid algorithms, and also provided some quantitative information about the performance and design of the solvers. Although the approach is applied to constant coefficients, linear problems and uniform grids, some general results are established, based on the fact that some Local Mode Analysis can be performed at the matrices. A different treatment has to be given to the part of the matrix associated with interior unknowns and that associated to boundary degrees of freedom. Ignoring the treatment of the boundaries, we now revise some of the results given in [10]. There, the author defined the convergence factor of the two-grid method by

$$\lambda = \sup \frac{\|u_{POST}^{\nu_2}\|}{\|u_{PRE}^{0}\|} \tag{9.1}$$

in some appropriate chosen norm that might depend on the problem. It is also shown, that under the assumptions described above (constant coefficients, linear problems, uniform grids and neglecting the boundary condition), the convergence factor might be computed in terms of the symbol of the error propagation operator.

For the linear iteration, the error propagation operator is defined as:

$$E_h := I - S_h A_h, \tag{9.2}$$

where I is the identity operator in V_h^k. The spectral radius, or some norm of this operator allows to quantify how the error is reduced at each iteration. If it is less than 1 we will get a convergent iteration. The smaller it is, the faster is the iteration.

In the case of the two level or two-grid cycle defined in **Algorithm 1**, the error propagation is:

$$E_h^{2grid} := E_h^{\nu_2}[I - P_{hH}A_H^{-1}R_{Hh}A_h]E_h^{\nu_1}. \tag{9.3}$$

Following [10], if one could choose the L^2-norm, denotes by $\widehat{E}_h(\theta)$ the symbol (in the frequency space) of the error propagation operator, from Parseval's identity it is formally obtained

$$\lambda = \sup_{\theta \neq 0} \|\widehat{E}_h(\theta)\|. \tag{9.4}$$

While for symmetric problems, the estimation of the spectral radius of E_h could be reduced to the computation of its largest eigenvalue, in the present situation, since A_h is non-symmetric and also S_h, one can not guarantee that their spectral information contain the relevant information.

To compute the spectral radius, a possible way is to compute the first singular value of E_h. In particular, one can define the *asymptotic convergence factor* (see [10]) as:

$$\lambda_{asymp} = \sup_{\theta \neq 0} \sigma_1(\widehat{E}_h(\theta)), \tag{9.5}$$

where σ_1 is the spectral radius of E, (i.e. largest absolute eigenvalue). On the other hand, a more restrictive way of ensuring that $\|E_h^{2grid}\| < 1$ would be to study the norms of each of the terms in the product. From the definition, we see that an essential requirement is to guarantee that the smoother has norm strictly less than 1.

References

1. Alexander, R.: Diagonally implicit Runge–Kutta methods for stiff O.D.E'.S. SIAM J. Numer. Anal. **14**, 1006–1021 (1977)
2. Barrett, J.W., Blowey, J.F.: Finite element approximation of a model for phase separation of a multi-component alloy with non-smooth free energy. Numer. Math. **77**, 1–34 (1997)
3. Barrett, J.W., Blowey, J.F., Garcke, H.: Finite element approximation of the Cahn–Hilliard equation with degenerate mobility. SIAM J. Numer. Anal. **37**, 286–318 (1999)
4. Barrett, J.W., Blowey, J.F., Garcke, H.: On fully practical finite element approximations of degenerate Cahn–Hilliard systems. M2ANMath. Model. Numer. Anal. **35**, 713–748 (2001)
5. Bassi, F., Ghidoni, A., Rebay, S., Tesini, P.: High-order accurate p-multigrid discontinuous Galerkin solution of the Euler equations. Int. J. Numer. Methods Fluids **60**, 847–865 (2008)
6. Bassi, F., Ghidoni, A., Rebay, S.: Optimal Runge–Kutta smoothers for the p-multigrid discontinuous Galerkin solution of the 1D Euler equations. J. Comput. Phys. **230**, 4153–4175 (2011)
7. Blowey, J.F., Elliott, C.M.: The Cahn–Hilliard gradient theory for phase separation with nonsmooth free energy. II. Numerical analysis. Eur. J. Appl. Math. **3**, 147–179 (1992)
8. Blowey, J.F., Copetti, M.I.M., Elliott, C.M.: Numerical analysis of a model for phase separation of multi-component alloy. IMA J. Numer. Anal. **16**, 111–139 (1996)
9. Brandt, A.: Multigrid techniques: 1984 guide with applications to fluid dynamics. GMD-Studien [GMD Studies], 85. Gesellschaft für Mathematik und Datenverarbeitung mbH, St. Augustin (1984)
10. Brandt, A.: Rigorous quantitative analysis of multigrid. I. Constant coefficients two-level cycle with L_2-norm. SIAM J. Numer. Anal. **31**, 1695–730 (1994)
11. Choo, S.M., Lee, Y.J.: A discontinuous Galerkin method for the Cahn–Hilliard equation. J. Appl. Math. Comput. **18**, 113–126 (2005)
12. Cockburn, B., Shu, C.-W.: The local discontinuous Galerkin method for time-dependent convection-diffusion systems. SIAM J. Numer. Anal. **35**, 2440–2463 (1998)
13. Copetti, M.I.M., Elliott, C.M.: Numerical analysis of the Cahn–Hilliard equation with logarithmic free energy. Numer. Math. **63**, 39–65 (1992)
14. Elliott, C.M., French, D.A.: A nonconforming finite-element method for the two-dimensional Cahn–Hilliard equation. SIAM J. Numer. Anal. **26**, 884–903 (1989)
15. Elliott, C.M., French, D.A., Milner, F.A.: A second order splitting method for the Cahn–Hilliard equation. Numer. Math. **54**, 575–590 (1989)
16. Eyre, D.J.: Systems of Cahn–Hilliard equations. SIAM J. Appl. Math. **53**, 1686–1712 (1993)
17. Feng, X., Prohl, A.: Analysis of a fully discrete finite element method for the phase field model and approximation of its sharp interface limits. Math. Comput. **73**, 541–567 (2004)

18. Feng, X.B., Karakashian, O.A.: Fully discrete dynamic mesh discontinuous Galerkin methods for the Cahn–Hilliard equation of phase transition. Math. Comput. **76**, 1093–1117 (2007)
19. Furihata, D.: A stable and conservative finite difference scheme for the Cahn–Hilliard equation. Numer. Math. **87**, 675–699 (2001)
20. Hector, Gomez B., Calo, Victor M., Hughes Thomas, J.R.: Isogeometric analysis of the Cahn–Hilliard phase-field model. Comput. Methods Appl. Mech. Eng. **197**, 4333–4352 (2008)
21. Guo, R., Xu, Y., Blanca Ayuso de Dios.: Multigrid methods of discontinuous Galerkin discretization for linear time-dependent fourth-order equations, Preprint
22. Kay, D., Welford, R.: A multigrid finite element solver for the Cahn–Hilliard equation. J. Comput. Phys. **212**, 288–304 (2006)
23. Kim, J., Kang, K., Lowengrub, J.: Conservative multigrid methods for Cahn–Hilliard fluids. J. Comput. Phys. **193**, 511–543 (2004)
24. Kim, J., Kang, K., Lowengrub, J.: Conservative multigrid methods for ternary Cahn–Hilliard systems. Commun. Math. Sci. **2**, 53–77 (2004)
25. Klaij, C.M., van Raalte, M.H., van der Ven, H., van der Vegt, J.J.W.: h-Multigrid for space-time discontinuous Galerkin discretizations of the compressible Navier–Stokes equations. J. Comput. Phys. **227**, 1024–1045 (2007)
26. Reed, W., Hill, T.: Triangular mesh methods for the neutrontransport equation, La-ur-73-479, Los Alamos Scientific Laboratory (1973)
27. Shahbazi, K., Mavriplis, D.J., Burgess, N.K.: Multigrid algorithms for high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations. J. Comput. Phys. **228**, 7917–7940 (2009)
28. Sun, Z.Z.: A second-order accurate linearized difference scheme for the twodimensional Cahn–Hilliard equation. Math. Comput. **64**, 1463–1471 (1995)
29. Trottenberg, U., Oosterlee, C., Schller, A.: Multigrid. Academic Press, New York (2005)
30. van der Vegt, J.J.W., Rhebergen, S.: hp-multigrid as smoother algorithm for higher order discontinuous Galerkin discretizations of advection dominated flows. Part I. Multilevel analysis. J. Comput. Phys. **231**, 7537–7563 (2012)
31. van der Vegt, J.J.W., Rhebergen, S.: hp-multigrid as smoother algorithm for higher order discontinuous Galerkin discretizations of advection dominated flows. Part II. Optimization of the Runge–Kutta smoother. J. Comput. Phys. **231**, 7564–7583 (2012)
32. Wells, G.N., Kuhl, E., Garikipati, K.: A discontinuous Galerkin method for the Cahn–Hilliard equation. J. Comput. Phys. **218**, 860–877 (2006)
33. Wise, S.M.: Unconditionally stable finite difference, nonlinear multigrid simulation of the Cahn–Hilliard–Hele–Shaw system of equations. J. Sci. Comput **44**, 38–68 (2010)
34. Xia, Y., Xu, Y., Shu, C.-W.: Local discontinuous Galerkin methods for the Cahn–Hilliard type equations. J. Comput. Phys. **227**, 472–491 (2007)
35. Xia, Y., Xu, Y., Shu, C.-W.: Efficient time discretization for local discontinuous Galerkin methods. Discret. Contin. Dyn. Syst. Ser. B **8**, 677–693 (2007)
36. Xu, Y., Shu, C.-W.: Local discontinuous Galerkin methods for high-order time-dependent partial differential equations. Commun. Comput. Phys. **7**, 1–46 (2010)