Convergence analysis of primal-dual based methods for total variation minimization with finite element approximation

WenYi Tian* Xiaoming Yuan†

Abstract

We consider a minimization model with total variational regularization, which can be reformulated as a saddle-point problem and then be efficiently solved by the primal-dual method. We utilize the consistent finite element method to discretize the saddle-point reformulation; thus possible jumps of the solution can be captured over some adaptive meshes and a generic domain can be easily treated. Our emphasis is analyzing the convergence of a more general primal-dual scheme with a combination factor for the discretized model. We establish the global convergence and derive the worst-case convergence rate measured by the iteration complexity for this general primal-dual scheme. This analysis is new in the finite element context for the minimization model with total variational regularization under discussion. Furthermore, we propose a prediction-correction scheme based on the general primal-dual scheme, which can significantly relax the step size for the discretization in the time direction. Its global convergence and the worst-case convergence rate are also established. Some preliminary numerical results are reported to verify the rationale of considering the general primal-dual scheme and the primal-dual-based prediction-correction scheme.

Keywords: Total variation minimization, saddle-point problem, finite element method, primal-dual method, convergence rate.

1 Introduction

We consider the total variation (TV) minimization model in [51]:

$$\inf_u \Big\{E(u) := \|Du\| + \frac{\alpha}{2}\|u-g\|^2_{L^2(\Omega)}\Big\}, \tag{1.1}$$

where $E : BV(\Omega) \to \mathbb{R}$ with a bounded Lipschitz domain $\Omega \subset \mathbb{R}^2$ is an energy functional, $g \in L^2(\Omega)$ is a given function, $\alpha > 0$ is a parameter, $BV(\Omega)$ is the bounded variation space consists of all functions $v \in L^1(\Omega)$ satisfying $\|Dv\| < +\infty$, and $\|Dv\|$ denotes the TV norm defined by

$$\|Dv\| := \sup\Big\{\int_\Omega v \,\mathrm{div}\varphi \,\mathrm{d}x : \varphi \in C^1_c(\Omega;\mathbb{R}^2), \|\varphi\|_\infty \le 1\Big\}. \tag{1.2}$$

In (1.2), $\|\varphi\|_\infty = \sup_{x\in\Omega}(\sum_{i=1}^2 |\varphi_i(x)|^2)^{1/2}$, "$Dv$" represents the gradient of v in the distributional sense, "div" denotes the divergence operator, and $C^1_c(\Omega;\mathbb{R}^2)$ is the set of once continuously differentiable $\mathbb{R}^2$-valued functions with compact support in Ω. The $BV(\Omega)$ space endowed with norm $\|v\|_{BV} := \|v\|_{L^1(\Omega)} + \|Dv\|$ is a Banach space. We refer the reader to, e.g., [3, 6, 7, 58], for more details. For the term $\|u-g\|^2_{L^2(\Omega)}$

*Center for Applied Mathematics, Tianjin University, Tianjin 300072, China. This author was partially supported by National Natural Science Foundation of China (Nos. 11626250, 11701416). Email: twymath@gmail.com

†Department of Mathematics, The University of Hong Kong, Hong Kong, China. This author was partially supported by the General Research Fund from Hong Kong Research Grants Council: HKBU 12300515. Email: xmyuan@hku.hk

in (1.1), it is well defined because the space $BV(\Omega)$ is continuously embedded into $L^2(\Omega)$ for $\Omega \subset \mathbb{R}^2$, see, e.g., Theorem 10.1.3 in [6]. Note that the model (1.1) has a unique solution in $BV(\Omega)$ because of the strict convexity of the quadratic term of the energy functional $E(u)$. As well studied in the literature, the TV regularization is capable of well preserving sharp edges of digital images or functions with the piecewise-smooth structure; and it has found a variety of applications in areas such as image restoration [17, 22, 27], optimal control and inverse problems [1, 23, 50], and parameter identification in partial differential equations [21, 24, 26, 40]. For other applications of the model (1.1), we refer to, e.g., geometric measure theory [11], and image denoising or function regularization [17, 25, 51, 53].

To find a numerical solution for (1.1), we can consider its formal Euler-Lagrange equation

$$-\operatorname{div}\Big(\frac{\nabla u}{|\nabla u|}\Big) + \alpha(u-g) = 0, \tag{1.3}$$

where ∇u is the gradient of u in $L^1(\Omega)$, and $|\cdot|$ denotes the Euclidean norm in $\mathbb{R}^2$. In the literature, there are various numerical schemes that are applicable to (1.3) or its regularized equation with homogeneous Neumann boundary condition. For instance, the time marching scheme in [51], the linear semi-implicit fixed-point method in [20, 28, 54], the interior-point primal-dual implicit quadratic methods in [22] and some others in [7, 16, 19, 55]. Furthermore, we can consider the L^2 gradient flow of the model (1.1) and its regularized problem constructed by evolving the Euler-Lagrange equation:

$$\partial_t u - \operatorname{div}\Big(\frac{\nabla u}{|\nabla u|}\Big) + \alpha(u-g) = 0 \tag{1.4}$$

with Neumann boundary condition and initial data from the theoretical and computational aspects. In (1.4), "∂_t" means the time partial derivative. We refer to, e.g., [4, 10, 29, 33, 34, 41, 42, 51, 52], for more discussions for (1.4). In particular, based on the methods in [8, 18], an algorithm for (1.4) with $\alpha = 0$ was studied in [10].

Replacing the term $\frac{\nabla u}{|\nabla u|}$ in (1.4) by a new variable p, we consider

$$\partial_t u - \operatorname{div} p + \alpha(u-g) = 0, \quad p \in \partial|\nabla u|, \tag{1.5}$$

supplemented by Neumann boundary conditions, where $\partial(|\cdot|)$ denotes the subdifferential of the absolute value function $|\cdot|$. Indeed, (1.5) is the L^2 gradient flow of the energy functional $E(u)$. As indicated in [8], by the equivalence

$$p \in \partial|\nabla u| \;\Leftrightarrow\; \nabla u \in \partial I_B(p),$$

where $B = \{p \in L^1(\Omega;\mathbb{R}^2) : \|p\|_\infty \le 1\}$ and $I_B(\cdot)$ denotes its indicator function, it motivates us to consider the following system of evolution equations:

$$\partial_t u - \operatorname{div} p + \alpha(u-g) = 0, \quad -\sigma\partial_t p + \nabla u \in \partial I_B(p) \tag{1.6}$$

with a parameter $\sigma > 0$ as the scale for ∇u. Note that the system (1.6) can also be regarded as the simultaneous gradient flow for the saddle-point formulation of the model (1.1)

$$\inf_u E(u) = \inf_u \sup_p \Big\{\mathcal{E}(u,p) := \frac{\alpha}{2}\|u-g\|^2_{L^2(\Omega)} - \int_\Omega u \operatorname{div} p \,\mathrm{d}x - I_B(p)\Big\}. \tag{1.7}$$

We call (1.7) a saddle-point problem because it aims at finding a solution point that minimizes the functional $\mathcal{E}(u,p)$ in the u-direction and meanwhile maximizes the same functional in the p-direction. We refer to, e.g., [5], for more backgrounds and a fundamental work of saddle-point problems.

Different from most of the existing work using finite difference discretization, we utilize the finite element method as in [8, 9] to discretize the saddle-point reformulation (1.7) so that a generic domain Ω in

(1.1) can be easily treated by finite element meshes and local effects of the solution of (1.1) such as possible jumps inside the domain Ω can be captured over some adaptive meshes. Furthermore, the solution point of (1.1) can be easily represented in an appropriate finite element space belonging to $BV(\Omega)$. Note that the solution point of the finite-element-discretized model can be guaranteed to be unique if proper finite element spaces belonging to $BV(\Omega)$ are chosen; and the error estimate for using the finite element discretization can be easily derived. Indeed, only some low order polynomials will be chosen for the finite element spaces because of the low regularity of the functions in the spaces $BV(\Omega)$ and $L^1(\Omega)$. As mentioned in [8], the piecewise constant and piecewise affine globally continuous finite element spaces are dense in $BV(\Omega)$ with respect to weak* convergence in $BV(\Omega)$. In addition, it was demonstrated in [8] that in gernal the piecewise constant finite element approximation for u cannot be expected to converge to an exact solution. Thus, the following finite element spaces

$$\begin{cases} \mathcal{S}^1(\mathcal{T}_h) = \{v_h \in C(\bar{\Omega}) : v_h|_T \text{ is affine for each } T \in \mathcal{T}_h\}, \\ \mathcal{L}^0(\mathcal{T}_h) = \{q_h \in L^1(\Omega) : q_h|_T \text{ is constant for each } T \in \mathcal{T}_h\}, \end{cases}$$

are built in [8] to approximate the functions u and p in (1.7), respectively, where $\bar{\Omega} = \Omega \cup \partial\Omega$ with $\partial\Omega$ being the boundary of the domain Ω , $\mathcal{T}_h$ denotes as a regular triangulation of Ω into triangles and $h = \max_{T\in\mathcal{T}_h} \operatorname{diam}(T)$ as the maximal diameter. Furthermore, in [8], the L^2 scalar product to equip $\mathcal{L}^0(\mathcal{T}_h)^2$, i.e., a space of piecewise constant vector fields, is based on the identity

$$\|Du_h\| = \sup_{p_h\in\mathcal{L}^0(\mathcal{T}_h)^2, \|p_h\|_\infty\le 1} \int_\Omega \nabla u_h \cdot p_h \, \mathrm{d}x.$$

Then, the TV minimization model (1.1) with finite element approximation can be reformulated as the following saddle-point problem:

$$\inf_{u_h} E(u_h) = \inf_{u_h\in\mathcal{S}^1(\mathcal{T}_h)} \sup_{p_h\in\mathcal{L}^0(\mathcal{T}_h)^2} \left\{ \mathcal{E}(u_h, p_h) = \frac{\alpha}{2}\|u_h - g\|^2_{L^2(\Omega)} + \int_\Omega \nabla u_h \cdot p_h \, \mathrm{d}x - I_B(p_h) \right\}. \quad (1.8)$$

Note that (1.8) can also be viewed as a discretized form of (1.7) because the discrete divergence operator div is the conjugate operator of $-\nabla$ satisfying $(\mathrm{div} q_h, v_h) = -(q_h, \nabla v_h)$. For the discretized saddle-point problem (1.8), it has a solution point $(u_h, p_h) \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{L}^0(\mathcal{T}_h)^2$, because $\mathcal{E}(u_h, p_h)$ is a closed, proper and convex-concave functional, see, e.g., [8].

For solving a saddle-point problem including the special form (1.8), the primal-dual method has received much attention from different areas, see, e.g., earlier work on the inexact Uzawa method [5, 14, 30, 35, 36, 49, 59] for saddle-point linear systems resulted from the numerical approximation of elasticity problems or stokes equations, and quadratic programming problems with linear constraints. Moreover, in some work such as [13, 18, 31, 56, 57], its particular applications to various image processing problems have been intensively investigated. As analyzed in [31, 56], the primal-dual method is a variant of the inexact Uzawa method [5, 30]. In [8], it is suggested to solve the saddle-point problem (1.8) by the following iterative scheme:

$$\begin{cases} \big(d_t u_h^{n+1} + \alpha(u_h^{n+1} - g), v_h\big) + \big(p_h^n, \nabla v_h\big) = 0, \quad \forall\, v_h \in \mathcal{S}^1(\mathcal{T}_h), \\ \tilde{u}_h^{n+1} = u_h^{n+1} + \tau d_t u_h^{n+1}, \\ \big(-\sigma d_t p_h^{n+1} + \nabla \tilde{u}_h^{n+1}, q_h - p_h^{n+1}\big) \le 0, \quad \forall\, q_h \in \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \end{cases} \quad (1.9)$$

where $d_t v^n = (v^{n+1} - v^n)/\tau$ with $\tau > 0$ being the step size in the time direction for any sequence $\{v^n\}_{n\in\mathbb{N}}$, and the parameter $\sigma > 0$ plays the role of the ratio of the step sizes used for the two subproblems of (1.9). The scheme (1.9) can be viewed as a semi-implicit difference scheme for (1.6) with finite element discretization in space, and the convergence of the sequence $\{u_h^n\}$ by (1.9) is analyzed in [8] under the

condition $\tau^2\|\nabla\|^2/\sigma \leq 1$. It is easy to see that the scheme (1.9) is the variational form of the following primal-dual scheme:

$$\begin{cases} u_h^{n+1} = \underset{v_h\in\mathcal{S}^1(\mathcal{T}_h)}{\arg\min}\left\{\mathcal{E}(v_h,p_h^n)+\dfrac{1}{2\tau}\|v_h-u_h^n\|^2_{L^2(\Omega)}\right\}, \\ \tilde{u}_h^{n+1} = 2u_h^{n+1}-u_h^n, \\ p_h^{n+1} = \underset{q_h\in\mathcal{L}^0(\mathcal{T}_h)^2}{\arg\max}\left\{\mathcal{E}(\tilde{u}_h^{n+1},q_h)-\dfrac{\sigma}{2\tau}\|q_h-p_h^n\|^2_{L^2(\Omega)}\right\}. \end{cases} \tag{1.10}$$

Note that τ can be understood as the step size for implementing gradient-based iterative methods for the minimization and maximization subproblems in (1.10).

Inspired by the work [18, 37], in this paper we consider a more general primal-dual scheme, which includes the scheme (1.10) as a special case, for the model (1.8). More specifically, we consider the following scheme for solving (1.8):

$$\begin{cases} u_h^{n+1} = \underset{v_h\in\mathcal{S}^1(\mathcal{T}_h)}{\arg\min}\left\{\mathcal{E}(v_h,p_h^n)+\dfrac{1}{2\tau}\|v_h-u_h^n\|^2_{L^2(\Omega)}\right\}, \\ \tilde{u}_h^{n+1} = u_h^{n+1}+\theta(u_h^{n+1}-u_h^n), \\ p_h^{n+1} = \underset{q_h\in\mathcal{L}^0(\mathcal{T}_h)^2}{\arg\max}\left\{\mathcal{E}(\tilde{u}_h^{n+1},q_h)-\dfrac{\sigma}{2\tau}\|q_h-p_h^n\|^2_{L^2(\Omega)}\right\}, \end{cases} \tag{1.11}$$

where the combination factor $\theta \in [-1,1]$. Clearly, (1.10) is a special case of (1.11) with $\theta = 1$. Note that it is in [18] that θ was extended to $[0,1]$ and then in [37] to $[-1,1]$ in the optimization context. This generalization can accelerate the convergence numerically as shown in [37], and it provides more insights in algorithmic design as shown in [48].

Our contributions can be summarized as follows. 1) We propose the general primal-dual scheme (1.11) for the discretized saddle-point problem (1.8) and prove its convergence. 2) We establish the worst-case convergence rate measured by the iteration complexity for the scheme (1.11). 3) We propose a new prediction-correction scheme in which the output of (1.11) needs to be refreshed by a correction step (5.3). This primal-dual-based prediction-correction scheme can significantly relax the restriction on the discretization step size τ from $O(h^2)$ to $O(h)$. 4) We also establish the convergence and the worst-case convergence rate measured by the iteration complexity for the primal-dual-based prediction-correction scheme.

The rest of this paper is organized as follows. In Section 2, some known results which are useful for further analysis are summarized. In Section 3, we focus on the general primal-dual scheme (1.11) and give some remarks. Then, the analysis of convergence and convergence rate for the general primal-dual scheme is presented in Section 4. In Section 5, we propose a primal-dual-based prediction-correction scheme and analyze its convergence and convergence rate. Some preliminary numerical results are reported in Section 6 to verify the effectiveness of the general primal-dual scheme and the new primal-dual-based prediction-correction scheme. Finally, some conclusions are made in Section 7.

2 Preliminary

In this section, we summarize some known results in the literature for the convenience of further analysis. Most of the results can be found in [8]. Throughout this paper, the notation $(\cdot,\cdot)$ stands for the inner product in L^2 or $(L^2)^2$.

First, the first-order optimality condition for the minimization of the energy functional $E(u)$ in $\mathcal{S}^1(\mathcal{T}_h)$ is stated in the following lemma.

Lemma 2.1 ([8]). *The function $u_h \in \mathcal{S}^1(\mathcal{T}_h)$ minimizes the energy functional $E(u)$ in $\mathcal{S}^1(\mathcal{T}_h)$ if and only if there exists $p_h \in \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2) := \{q_h \in \mathcal{L}^0(\mathcal{T}_h)^2 : \|q_h\|_\infty \le 1\}$ such that*

$$\begin{cases} (p_h, \nabla v_h) + \alpha(u_h - g, v_h) = 0, & \forall\, v_h \in \mathcal{S}^1(\mathcal{T}_h), \\ (\nabla u_h, q_h - p_h) \le 0, & \forall\, q_h \in \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \end{cases} \tag{2.1}$$

Notice that the optimality condition (2.1) can be rewritten as the following variational inequality (VI) in a compact form: Finding $\mu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2)$ such that

$$\big(F(\mu_h), \nu_h - \mu_h\big) \ge 0, \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \tag{2.2}$$

where

$$\mu_h = \begin{pmatrix} u_h \\ p_h \end{pmatrix}, \quad \nu_h = \begin{pmatrix} v_h \\ q_h \end{pmatrix}, \quad F(\mu_h) = \begin{pmatrix} -\mathrm{div} p_h + \alpha(u_h - g) \\ -\nabla u_h \end{pmatrix}. \tag{2.3}$$

It is easy to see that the mapping $F(\cdot)$ in (2.3) satisfies

$$\big(F(\mu_h) - F(\nu_h), \mu_h - \nu_h\big) = \alpha \|u_h - v_h\|^2_{L^2(\Omega)}. \tag{2.4}$$

It follows from (2.2), (2.4) and Lemma 2.1 that the first component u_h of a solution pair of (1.8) is unique. But the second one p_h is not unique in general.

The error of the finite element approximation of function u in (1.8) is given by the following theorem.

Theorem 2.1 ([8, 10]). *Let $\Omega = (0,1)^2$, we have $u_h \to u$ in $L^2(\Omega)$ as $h \to 0$. If $u \in Lip(\beta, L^2(\Omega))$ for some $0 < \beta \le 1$, then*

$$\|u - u_h\|^2_{L^2(\Omega)} \le c h^{\frac{\beta}{1+\beta}};$$

if $u \in BV(\Omega) \cap L^\infty(\Omega)$, then

$$\|u - u_h\|_{L^2(\Omega)} \le c h^{1/4}.$$

We say $u \in Lip(\beta, L^2(\Omega))$ if

$$\sup_{t>0} t^{-\beta} \sup_{|y| \le t} \Big(\int_\Omega |u(x+y) - u(x)|^2 \mathrm{d}x \Big)^{1/2} < +\infty.$$

It is mentioned in [8] that the condition $u \in Lip(\beta, L^2(\Omega))$ is satisfied if $g \in Lip(\beta, L^2(\Omega))$; and if $g \in L^\infty(\Omega)$ then $u \in Lip(\frac{1}{2}, L^2(\Omega))$.

3 A General Primal-Dual Scheme

In this section, we specify the general primal-dual scheme (1.11) for solving (1.8) and present the resulting algorithm. Let us use the notation

$$d_t v^{n+1} = \frac{v^{n+1} - v^n}{\tau}$$

for a sequence $\{v^n\}_{n\in\mathbb{N}}$, where $\tau>0$ is the discretization step size.

Algorithm 1: A general primal-dual scheme for (1.8)

Input: Choose an initial iteration $(u_h^0,p_h^0)\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{L}^0(\mathcal{T}_h)^2$. Choose constants $\tau,\sigma>0$ and $\theta\in[-1,1]$ such that

$$\Big(\theta^2+\frac{(1-\theta)^2}{2\alpha\tau}\Big)\frac{\tau^2\|\nabla\|^2}{\sigma}<1. \tag{3.1}$$

for $n=0,1,2,\cdots,$ **do**

Step 1 Update u_h^{n+1} by solving

$$u_h^{n+1}=\mathop{\arg\min}_{v_h\in\mathcal{S}^1(\mathcal{T}_h)}\Big\{\frac{\alpha}{2}\|v_h-g\|^2_{L^2(\Omega)}+\int_\Omega\nabla v_h\cdot p_h^n\,\mathrm{d}x+\frac{1}{2\tau}\|v_h-u_h^n\|^2_{L^2(\Omega)}\Big\}; \tag{3.2}$$

Step 2 Set

$$\tilde{u}_h^{n+1}=u_h^{n+1}+\theta\tau d_t u_h^{n+1}; \tag{3.3}$$

Step 3 Update p_h^{n+1} satisfying

$$p_h^{n+1}=\mathop{\arg\max}_{q_h\in\mathcal{L}^0(\mathcal{T}_h)^2}\Big\{\big(q_h,\nabla\tilde{u}_h^{n+1}\big)-I_B(q_h)-\frac{\sigma}{2\tau}\|q_h-p_h^n\|^2_{L^2(\Omega)}\Big\}. \tag{3.4}$$

end

Remark 3.1. *As mentioned, Algorithm 1 with $\theta=1$ reduces to the primal-dual scheme in [8], and it is noticed that p_h^{n+1} satisfying (3.4) is given by*

$$p_h^{n+1}=\big(p_h^n+(\tau/\sigma)\nabla\tilde{u}_h^{n+1}\big)/\max\big\{1,|p_h^n+(\tau/\sigma)\nabla\tilde{u}_h^{n+1}|\big\}.$$

Remark 3.2. *The solution points u_h^{n+1} and p_h^{n+1} of the first and third steps in Algorithm 1 satisfy the following inequalities, respectively:*

$$\big(d_t u_h^{n+1}+\alpha(u_h^{n+1}-g),v_h\big)+\big(p_h^n,\nabla v_h\big)=0,\quad\forall\,v_h\in\mathcal{S}^1(\mathcal{T}_h), \tag{3.5}$$

$$\big(-\sigma d_t p_h^{n+1}+\nabla\tilde{u}_h^{n+1},q_h-p_h^{n+1}\big)\le 0,\quad\forall\,q_h\in\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \tag{3.6}$$

and they can be viewed as the discretization of the systems

$$\begin{cases}\partial_t u_h=-\partial_v\mathcal{E}(u_h,p_h),\\ \sigma\partial_t p_h\in\partial_q\mathcal{E}(u_h,p_h),\end{cases} \tag{3.7}$$

which are the finite element approximations of the evolution systems (1.6). Moreover, the parameter τ in Algorithm 1 plays the role of a step size of the discretization in the time direction.

Remark 3.3. *An inverse estimate in [15] shows that there exists $c>0$ such that $\|\nabla v_h\|_{L^2(\Omega)}\le ch_{\min}^{-1}\|v_h\|_{L^2(\Omega)}$ for all $v_h\in\mathcal{S}^1(\mathcal{T}_h)$, where $h_{\min}=\min_{T\in\mathcal{T}_h}\mathrm{diam}(T)$. We denote*

$$\|\nabla\|=\sup_{v_h\in\mathcal{S}^1(\mathcal{T}_h)\setminus\{0\}}\frac{\|\nabla v_h\|_{L^2(\Omega)}}{\|v_h\|_{L^2(\Omega)}}\le ch_{\min}^{-1}, \tag{3.8}$$

which will be used in the theoretical analysis later. For a regular mesh $\mathcal{T}_h$, it yields from the above estimate that $\|\nabla\|\le ch^{-1}$.

4 Convergence Analysis for Algorithm 1

In this section, we prove the convergence for Algorithm 1 and establish its worst-case convergence rate measured by the iteration complexity. As in [37], our analysis follows the framework for contraction type methods in [12].

4.1 Convergence

First, for the iteration $\mu_h^{n+1} = (u_h^{n+1}; p_h^{n+1}) \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2)$ generated by Algorithm 1, it is easy to see from (3.5) and (3.6) that it satisfies the VI:

$$\big(F(\mu_h^{n+1}) + M(\mu_h^{n+1} - \mu_h^n), \nu_h - \mu_h^{n+1}\big) \geq 0, \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2) \tag{4.1}$$

with

$$M = \begin{pmatrix} \frac{1}{\tau} I & \mathrm{div} \\ -\theta\nabla & \frac{\sigma}{\tau} I \end{pmatrix}. \tag{4.2}$$

We prove two lemmas prior to the main convergence theorem. In the following lemma, a useful inequality is proved.

Lemma 4.1. *Let the sequence $\{\mu_h^{n+1} = (u_h^{n+1}; p_h^{n+1})\}$ be generated by Algorithm 1 with $\theta \in [-1, 1]$. Then, we have*

$$\begin{aligned} \big(G(\mu_h^{n+1} - \mu_h^n), \nu_h - \mu_h^{n+1}\big) \geq\ & \alpha \|v_h - u_h^{n+1}\|^2_{L^2(\Omega)} + \big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) \\ & - (1-\theta)\big(\nabla(v_h - u_h^{n+1}), p_h^n - p_h^{n+1}\big), \\ & \forall\, \nu_h = (v_h; q_h) \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \end{aligned} \tag{4.3}$$

where

$$G = \begin{pmatrix} \frac{1}{\tau} I & \theta\mathrm{div} \\ -\theta\nabla & \frac{\sigma}{\tau} I \end{pmatrix}. \tag{4.4}$$

Proof. We can rewrite (4.1) as

$$\begin{aligned} &\big(F(\mu_h^{n+1}) + G(\mu_h^{n+1} - \mu_h^n), \nu_h - \mu_h^{n+1}\big) - (1-\theta)\big(\nabla(v_h - u_h^{n+1}), p_h^{n+1} - p_h^n\big) \\ &\geq 0, \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \end{aligned} \tag{4.5}$$

Then, adding $\big(F(\nu_h), \mu_h^{n+1} - \nu_h\big)$ to both sides of (4.5) yields

$$\begin{aligned} &\big(G(\mu_h^{n+1} - \mu_h^n), \nu_h - \mu_h^{n+1}\big) \\ &\geq \big(F(\nu_h) - F(\mu_h^{n+1}), \nu_h - \mu_h^{n+1}\big) + \big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) \\ &\quad - (1-\theta)\big(\nabla(v_h - u_h^{n+1}), p_h^n - p_h^{n+1}\big), \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \end{aligned} \tag{4.6}$$

which completes the proof by using the property (2.4) of F. □

The following lemma is also useful for further analysis.

Lemma 4.2. *Let us define*

$$Q = \begin{pmatrix} \frac{1}{\tau} I & \theta\mathrm{div} \\ -\theta\nabla & \left(\frac{\sigma}{\tau} - \frac{(1-\theta)^2\|\nabla\|^2}{2\alpha}\right) I \end{pmatrix}. \tag{4.7}$$

Then both the operators G defined in (4.4) and Q in (4.7) are positive definite if (3.1) holds.

Proof. Based on the definitions of G and Q, it suffices to prove the positive definiteness of Q. For any nonzero function $\nu_h = (v_h, q_h) \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2)$, under the condition (3.1), we have

$$\begin{aligned}
\|\nu_h\|_Q^2 = (Q\nu_h, \nu_h) &= \frac{1}{\tau}\|v_h\|_{L^2(\Omega)}^2 - 2\theta(\nabla v_h, q_h) + \Big(\frac{\sigma}{\tau} - \frac{(1-\theta)^2\|\nabla\|^2}{2\alpha}\Big)\|q_h\|_{L^2(\Omega)}^2 \\
&\geq \frac{1}{\tau}\|v_h\|_{L^2(\Omega)}^2 - 2|\theta|\|\nabla\|\|v_h\|_{L^2(\Omega)}\|q_h\|_{L^2(\Omega)} + \Big(\frac{\sigma}{\tau} - \frac{(1-\theta)^2\|\nabla\|^2}{2\alpha}\Big)\|q_h\|_{L^2(\Omega)}^2 \\
&> \left(\frac{1}{\sqrt{\tau}}\|v_h\|_{L^2(\Omega)} - \sqrt{\frac{\sigma}{\tau} - \frac{(1-\theta)^2\|\nabla\|^2}{2\alpha}}\|q_h\|_{L^2(\Omega)}\right)^2 \geq 0,
\end{aligned} \tag{4.8}$$

which completes the proof. □

Using the results proved in the above lemmas, we can show that the sequence generated by Algorithm 1 is strictly contractive (see, e.g., [12, 39], for the definition) with respect to the solution set of (1.8) under some conditions. We summarize the result in the following theorem.

Theorem 4.1 (Contraction)**.** *Let μ_h be the solution point of (1.8) and $\{\mu_h^{n+1}\}$ be the sequence generated by Algorithm 1 with $\theta \in [-1, 1]$. Under the condition (3.1), we have*

$$\|\mu_h^{n+1} - \mu_h\|_G^2 \leq \|\mu_h^n - \mu_h\|_G^2 - \|\mu_h^n - \mu_h^{n+1}\|_Q^2, \tag{4.9}$$

where G and Q are defined in (4.4) and (4.7), respectively.

Proof. Using Cauchy-Schwarz inequality and the definition of $\|\nabla\|$ in (3.8) for the estimation (4.3), we have

$$\begin{aligned}
\big(G(\mu_h^{n+1} - \mu_h^n), \nu_h - \mu_h^{n+1}\big) &\geq \alpha\|v_h - u_h^{n+1}\|_{L^2(\Omega)}^2 + \big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) \\
&\quad - \frac{\alpha}{\|\nabla\|^2}\|\nabla(v_h - u_h^{n+1})\|_{L^2(\Omega)}^2 \\
&\quad - \frac{(1-\theta)^2\|\nabla\|^2}{4\alpha}\|p_h^n - p_h^{n+1}\|_{L^2(\Omega)}^2 \\
&\geq \big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) - \frac{(1-\theta)^2\|\nabla\|^2}{4\alpha}\|p_h^n - p_h^{n+1}\|_{L^2(\Omega)}^2, \\
&\forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2).
\end{aligned} \tag{4.10}$$

Applying the identity

$$\big(G(b-a), b\big) = \frac{1}{2}\big(\|b\|_G^2 - \|a\|_G^2 + \|a-b\|_G^2\big) \tag{4.11}$$

to the term on the left-hand side of (4.10) with $b = \mu_h^{n+1} - \nu_h$ and $a = \mu_h^n - \nu_h$, we derive

$$\begin{aligned}
2\big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) &\leq \|\mu_h^n - \nu_h\|_G^2 - \|\mu_h^{n+1} - \nu_h\|_G^2 \\
&\quad - \Big(\|\mu_h^n - \mu_h^{n+1}\|_G^2 - \frac{(1-\theta)^2\|\nabla\|^2}{2\alpha}\|p_h^n - p_h^{n+1}\|_{L^2(\Omega)}^2\Big), \\
&\forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2).
\end{aligned} \tag{4.12}$$

Thus, combining the last two terms in the above inequality, we obtain

$$\begin{aligned}
2\big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) \leq \|\mu_h^n - \nu_h\|_G^2 - \|\mu_h^{n+1} - \nu_h\|_G^2 - \|\mu_h^n - \mu_h^{n+1}\|_Q^2, \\
\forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2).
\end{aligned} \tag{4.13}$$

Setting $\nu_h = \mu_h$ in (4.13) and using the optimality condition (2.2), then we get the result (4.9). □

The strict contraction of the sequence generated by Algorithm 1, which is implied by the assertion (4.9), essentially means that the convergence of the sequence $\{\mu_h^{n+1}\}$. We summarize the convergence result in the following theorem.

Theorem 4.2 (Convergence). *Let $\{\mu_h^{n+1} = (u_h^{n+1}; p_h^{n+1})\}$ be the sequence generated by Algorithm 1 with $\theta \in [-1, 1]$. Under the condition (3.1), the sequence $\{u_h^{n+1}\}$ converges to the unique minimizer of the problem (1.1) in $\mathcal{S}^1(\mathcal{T}_h)$.*

Proof. According to (4.9), for any integer $N > 0$, we have

$$\sum_{n=0}^{N} \|\mu_h^n - \mu_h^{n+1}\|_Q^2 \leq \|\mu_h - \mu_h^0\|_G^2.$$

Thus, we conclude

$$\lim_{n\to\infty} \|\mu_h^n - \mu_h^{n+1}\|_Q^2 = 0.$$

As Q is positive definite under the condition (3.1), then $\lim_{n\to\infty}(\mu_h^n - \mu_h^{n+1}) = 0$. It is implied from (4.9) that the sequence $\{\mu_h^n\}$ is bounded, and we denote by μ_h^* a cluster point of $\{\mu_h^n\}$. Substituting it into (4.1), we obtain that

$$\big(F(\mu_h^*), \nu_h - \mu_h^*\big) \geq 0, \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2),$$

which means μ_h^* is a solution point of (2.2). From (2.2) and (2.4), the first component of the solution μ_h^* to (1.8) is unique. Thus, the sequence $\{u_h^{n+1}\}$ converges to the unique minimizer of energy functional E in $\mathcal{S}^1(\mathcal{T}_h)$ by Lemma 2.1. □

4.2 Convergence Rate

In this subsection, we estimate a worst-case $O(\frac{1}{N})$ convergence rate measured by the iteration complexity for Algorithm 1 with $\theta \in [-1, 1]$ under the condition (3.1), where N denotes the iteration counter. Following the seminal work [43, 45] and many others, a worst-case $O(\frac{1}{N})$ convergence rate means the accuracy to a solution under certain criteria is of the order $O(\frac{1}{N})$ after N iterations of an iterative scheme; or equivalently, it requires at most $O(\frac{1}{\epsilon})$ iterations to achieve an approximate solution with an accuracy of ϵ.

First, we introduce a criterion to measure the accuracy of an approximate solution point of the VI (2.2).

Theorem 4.3. *The solution set of VI (2.2) is convex and can be characterized as*

$$\Theta = \bigcap_{\nu_h} \big\{\tilde{\mu}_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2) : \big(F(\nu_h), \nu_h - \tilde{\mu}_h\big) \geq 0\big\}.$$

Proof. The proof can refer to Theorem 2.3.5 in [32] or Theorem 2.1 in [38]. □

According to [44], we define

$$\mathcal{D}_U(\tilde{\mu}_h) := \{\nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2) : \|\nu_h - \tilde{\mu}_h\|_U \leq 1\}$$

with U a symmetric and positive definite operator. Then, Theorem 4.3 implies that we can say that $\tilde{\mu}_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2)$ is an approximate solution of VI (2.2) with an accuracy of ϵ if

$$\big(F(\nu_h), \tilde{\mu}_h - \nu_h\big) \leq \epsilon, \quad \forall\, \nu_h \in \mathcal{D}_U(\tilde{\mu}_h). \tag{4.14}$$

The result in the following theorem shows that we can find $\tilde{\mu}_N$ such that (4.14) is satisfied with $\epsilon = O(\frac{1}{N})$ after N iterations of Algorithm 1. Therefore, a worst-case $O(\frac{1}{N})$ convergence rate is established for Algorithm 1.

Theorem 4.4 (Convergence rate in the ergodic sense)**.** *Let* $\{\mu_h^{n+1}\}$ *be the sequence generated by Algorithm 1 with* $\theta \in [-1, 1]$ *under the condition* (3.1)*. For any integer* $N > 0$*, let*

$$\tilde{\mu}_N = \frac{1}{N+1} \sum_{n=0}^{N} \mu_h^{n+1}.$$

Then, we have

$$\big(F(\nu_h), \tilde{\mu}_N - \nu_h\big) \leq \frac{1}{2(N+1)} \|\nu_h - \mu_h^0\|_G^2, \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{4.15}$$

Proof. It follows from (4.13) that

$$\big(F(\nu_h), \mu_h^{n+1} - \nu_h\big) \leq \frac{1}{2}\big(\|\nu_h - \mu_h^n\|_G^2 - \|\nu_h - \mu_h^{n+1}\|_G^2\big), \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{4.16}$$

Summarizing (4.16) with $n = 0, 1, \cdots, N$, we have

$$\big(F(\nu_h), \sum_{n=0}^{N} \mu_h^{n+1} - (N+1)\nu_h\big) \leq \frac{1}{2}\big(\|\nu_h - \mu_h^0\|_G^2 - \|\nu_h - \mu_h^{N+1}\|_G^2\big), \tag{4.17}$$

which yields the result (4.15). □

This theorem shows a worst-case $O(\frac{1}{N})$ convergence rate in the ergodic sense for Algorithm 1. The ergodic sense is because of the fact that the approximate solution with an accuracy of $O(\frac{1}{N})$ is the average of all the N iterations generated by Algorithm 1. For the special case $\theta = 1$ of Algorithm 1, i.e., the primal-dual scheme in [8], we can obtain a stronger worst-case $O(\frac{1}{N})$ convergence rate in a nonergodic sense. But it is not clear if this convergence rate in a nonergodic sense can be extended to the general case of Algorithm 1 with $\theta \in [-1, 1)$. The main reason is that the matrix-form operator M defined in (4.2) is not symmetric if $\theta \neq 1$. Hence, it becomes difficult to define a norm with this matrix-form operator to measure the progress of proximity to the solution set. We summarize the stronger worst-case $O(\frac{1}{N})$ convergence rate in a nonergodic sense for Algorithm 1 in the following theorem. This is a by-produce of our main results.

Theorem 4.5 (Convergence rate for $\theta = 1$ in a nonergodic sense)**.** *Let* μ_h *be the solution of* (1.8) *and* $\{\mu_h^{n+1}\}$ *be the sequence generated by Algorithm 1 with* $\theta = 1$ *under the condition* (3.1)*. Then for any integer* $N > 0$*, it exists*

$$\|\mu_h^N - \mu_h^{N+1}\|_G^2 \leq \frac{1}{(N+1)} \|\mu_h - \mu_h^0\|_G^2. \tag{4.18}$$

Proof. First, it follows from (4.1) when $\theta = 1$ that

$$\big(F(\mu_h^{n+1}) + G(\mu_h^{n+1} - \mu_h^n), \nu_h - \mu_h^{n+1}\big) \geq 0, \ \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{4.19}$$

It also holds

$$\big(F(\mu_h^{n+2}) + G(\mu_h^{n+2} - \mu_h^{n+1}), \nu_h - \mu_h^{n+2}\big) \geq 0, \ \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{4.20}$$

Setting $\nu_h = \mu_h^{n+2}$ in (4.19) and $\nu_h = \mu_h^{n+1}$ in (4.20), and then combining them together, we obtain

$$\begin{aligned} &\big(F(\mu_h^{n+1}) - F(\mu_h^{n+2}), \mu_h^{n+2} - \mu_h^{n+1}\big) \\ &- \big(G((\mu_h^{n+2} - \mu_h^{n+1}) - (\mu_h^{n+1} - \mu_h^n)), \mu_h^{n+2} - \mu_h^{n+1}\big) \geq 0. \end{aligned} \tag{4.21}$$

Applying the equalities (4.11) and (2.4) to (4.21) yields

$$\begin{aligned}\|\mu_h^{n+1}-\mu_h^{n+2}\|_G^2 \le& \|\mu_h^n-\mu_h^{n+1}\|_G^2 - 2\alpha\|u_h^{n+1}-u_h^{n+2}\|_{L^2(\Omega)}^2\\ &-\|(\mu_h^{n+2}-\mu_h^{n+1})-(\mu_h^{n+1}-\mu_h^n)\|_G^2\\ \le& \|\mu_h^n-\mu_h^{n+1}\|_G^2.\end{aligned} \tag{4.22}$$

With $\theta=1$ and the definition of Q in (4.7), it follows from Theorem 4.1 that

$$\|\mu_h-\mu_h^{n+1}\|_G^2 \le \|\mu_h-\mu_h^n\|_G^2 - \|\mu_h^n-\mu_h^{n+1}\|_G^2. \tag{4.23}$$

Summarizing (4.23) over $n=0,1,\cdots,N$, we obtain

$$\sum_{n=0}^{N}\|\mu_h^n-\mu_h^{n+1}\|_G^2 \le \|\mu_h-\mu_h^0\|_G^2. \tag{4.24}$$

The estimate (4.22) reveals that $\|\mu_h^n-\mu_h^{n+1}\|_G^2$ is monotonically non-increasing. Then we obtain

$$(N+1)\|\mu_h^N-\mu_h^{N+1}\|_G^2 \le \|\mu_h-\mu_h^0\|_G^2, \tag{4.25}$$

which yields the result (4.18). □

It is follows from (4.19) that μ_h^{N+1} belongs to the solution set of VI (2.2) if $\|\mu_h^N-\mu_h^{N+1}\|_G^2=0$ since G is positive definite under the condition (3.1) with $\theta=1$. In other words, if $\|\mu_h^N-\mu_h^{N+1}\|_G^2=0$, we have

$$\big(F(\mu_h^{N+1}),\nu_h-\mu_h^{N+1}\big)\ge 0,\quad \forall\,\nu_h\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2),$$

which implies μ_h^{N+1} is a solution of (1.8) characterized by VI (2.2). Then the quantity $\|\mu_h^N-\mu_h^{N+1}\|_G^2$ can be used to measure the accuracy of an approximate solution of (1.8). Thus, the assertion in Theorem 4.5 shows a worst-case $O(\frac{1}{N})$ convergence rate measured by the iteration complexity in a nonergodic sense for Algorithm 1 with $\theta=1$.

5 A Primal-Dual Based Prediction-Correction Scheme

In Section 3, we propose Algorithm 1 which is more general than the primal-dual scheme in [8]. We will show in Section 6.1 that this general scheme with $\theta\neq 1$ can accelerate the convergence numerically; it thus makes sense to consider the generalization for $\theta\in[-1,1]$. Meanwhile, we have analyzed that the convergence of Algorithm 1 can be guaranteed under the condition (3.1). As indicated in Remark 3.3, we have $\|\nabla\|^2\le ch^{-2}$ with the regular mesh $\mathcal{T}_h$; and when $\theta\neq 1$ for Algorithm 1, (3.1) indicates $\tau\le ch^2$. Here τ stands for the discretization step size in the time direction if we regard Algorithm 1 as the discretizations of (3.7) (which is the evolution systems (1.6) with finite element approximation in space). In this sense, it is interesting to consider relaxing the requirement on τ from the order of $O(h^2)$ to $O(h)$. This is the main motivation we consider the new primal-dual based prediction-correction scheme in this section.

We would reiterate that the convergence analysis for Algorithm 1 in Section 4 mainly follows the analytic framework for contraction type methods. For its analysis, a key technique is that the condition (3.1) can ensure the positive definiteness of the matrix-form operators G in (4.4) and Q in (4.7). With their positive definiteness, we can measure the progress of proximity between two consecutive iterations and eventually establish the strict contraction property for the sequence generated by Algorithm 1 which essentially implies the convergence. It is seen from (4.4) that the off-diagonal entries of G is not zero operator, which means that it renders two non-square terms in the expansion of any quadratic term associated with the G-norm. This fact

essentially raises the reason of considering the condition (3.1) to sufficiently ensure the positive definiteness of G. For the purpose of relaxing the restriction on τ, we may consider only keeping the diagonal entries of G as the matrix form operator for defining the norm when measuring the progress of proximity between two consecutive iterations. That is, the analysis is based on H-norm with

$$H := \begin{pmatrix} \frac{1}{\tau} I & 0 \\ 0 & \frac{\sigma}{\tau} I \end{pmatrix}. \tag{5.1}$$

Based on this analysis, we will follow the idea in [37] and modify Algorithm 1 as a new prediction-correction scheme. Each of its iteration consists of the primal-dual step (3.2)-(3.4) and a correction step (5.3). With the additional correction step, the requirement on τ can be relaxed from $O(h^2)$ to $O(h)$. Moreover, the worst-case convergence rate in both the ergodic and nonergodic senses can be established for the new primal-dual-based prediction-correction scheme.

5.1 Algorithm

We summarize the new primal-dual-based prediction correction scheme as follows.

Algorithm 2: A primal-dual-based prediction-correction scheme for (1.8)

Input: Choose an initial iteration $(u_h^0, p_h^0) \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{L}^0(\mathcal{T}_h)^2$. Choose constants $\gamma \in (0, 1]$, $\theta \in [-1, 1]$, and $\tau > 0$ and $\sigma > 0$ such that

$$\frac{\tau^2 \|\nabla\|^2}{\sigma} < 1. \tag{5.2}$$

for $n = 0, 1, 2, \cdots,$ **do**

Prediction step: Obtain the predictor $\bar{\mu}_h^n$ by Algorithm 1 with input μ_h^n, i.e. (3.2)-(3.4);

Correction step: Generate the new iteration μ_h^{n+1} by solving

$$\begin{aligned} &\big((u_h^{n+1} - u_h^n) + \gamma(u_h^n - \bar{u}_h^n), v_h\big) - \tau\gamma\big(\nabla v_h, p_h^n - \bar{p}_h^n\big) = 0, \quad \forall\, v_h \in \mathcal{S}^1(\mathcal{T}_h), \\ &\big((p_h^{n+1} - p_h^n) + \gamma(p_h^n - \bar{p}_h^n), q_h\big) - \gamma\theta\frac{\tau}{\sigma}\big(\nabla(u_h^n - \bar{u}_h^n), q_h\big) = 0, \ \forall\, q_h \in \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \end{aligned} \tag{5.3}$$

end

Remark 5.1. *For the correction step (5.3), it can be rewritten as the compact form*

$$\big(H(\mu_h^{n+1} - \mu_h^n), \nu_h\big) + \gamma\big(M(\mu_h^n - \bar{\mu}_h^n), \nu_h\big) = 0, \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \tag{5.4}$$

where M and H are defined in (4.2) and (5.1), respectively. It is noticed that the correction step (5.3) in Algorithm 2 is not difficult to compute because it is essentially a system of linear equations with a banded, symmetric and positive-definite mass matrix.

Remark 5.2. *The parameter $\gamma \in (0, 1]$ in (5.3) is a relaxation factor which can potentially accelerate numerical performance. Instead, we can simply take $\gamma \equiv 1$ if the number of parameters is a concern for implementation.*

5.2 Convergence

In this subsection, we prove the convergence for Algorithm 2. First, we prove an inequality which is important for the convergence analysis. Let us recall (4.1). Thus, the predictor $\bar{\mu}_h^n$ generated by Algorithm 2 satisfies

$$\big(F(\bar{\mu}_h^n) + M(\bar{\mu}_h^n - \mu_h^n), \nu_h - \bar{\mu}_h^n\big) \geq 0, \ \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{5.5}$$

Lemma 5.1. *Let $\{\mu_h^{n+1}\}$ be the sequence generated by Algorithm 2 with $\theta \in [-1,1]$ and $\gamma \in (0,1]$. Then we have*

$$\begin{aligned} 2\big(H(\mu_h^n-\mu_h^{n+1}),\mu_h^n-\nu_h\big)-\|\mu_h^n-\mu_h^{n+1}\|_H^2 \geq \frac{1}{4}\Big(1-\frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\|\mu_h^n-\mu_h^{n+1}\|_H^2 \\ +2\gamma\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big),\ \forall\,\nu_h\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \end{aligned} \tag{5.6}$$

where H is defined in (5.1).

Proof. Adding $\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big)$ to both sides of (5.5), we have

$$\big(F(\bar{\mu}_h^n)-F(\nu_h)+M(\bar{\mu}_h^n-\mu_h^n),\nu_h-\bar{\mu}_h^n\big)\geq\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big),\forall\,\nu_h\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{5.7}$$

Then, we derive

$$\begin{aligned} \big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\nu_h\big)\geq\big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\bar{\mu}_h^n\big)+\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big) \\ +\alpha\|v_h-\bar{u}_h^n\|_{L^2(\Omega)}^2,\ \forall\,\nu_h\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \end{aligned} \tag{5.8}$$

With (5.4), we obtain

$$\begin{aligned} \big(H(\mu_h^n-\mu_h^{n+1}),\mu_h^n-\nu_h\big)&=\gamma\big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\nu_h\big)\\ &\geq\gamma\big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\bar{\mu}_h^n\big)+\gamma\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big),\\ &\forall\,\nu_h\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \end{aligned} \tag{5.9}$$

Using the definition of M in (4.2) and H in (5.1), we can expand the term on the right-hand side of (5.9) as

$$\big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\bar{\mu}_h^n\big)=\|\mu_h^n-\bar{\mu}_h^n\|_H^2-(1+\theta)\big(\nabla(u_h^n-\bar{u}_h^n),p_h^n-\bar{p}_h^n\big). \tag{5.10}$$

With (5.4), we also have

$$\begin{aligned} \|\mu_h^n-\mu_h^{n+1}\|_H^2&=\gamma\big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\mu_h^{n+1}\big)\\ &=\gamma\big(H(\mu_h^n-\mu_h^{n+1}),H^{-1}M(\mu_h^n-\bar{\mu}_h^n)\big)\\ &=\gamma^2\big(M(\mu_h^n-\bar{\mu}_h^n),H^{-1}M(\mu_h^n-\bar{\mu}_h^n)\big), \end{aligned} \tag{5.11}$$

whose last term can be evaluated by the definitions of H and M as

$$\begin{aligned} \big(M(\mu_h^n-\bar{\mu}_h^n),H^{-1}M(\mu_h^n-\bar{\mu}_h^n)\big)=\|\mu_h^n-\bar{\mu}_h^n\|_H^2+\tau\|\mathrm{div}(p_h^n-\bar{p}_h^n)\|^2 \\ +\theta^2\frac{\tau}{\sigma}\|\nabla(u_h^n-\bar{u}_h^n)\|^2 \\ -2(1+\theta)\big(\nabla(u_h^n-\bar{u}_h^n),p_h^n-\bar{p}_h^n\big). \end{aligned} \tag{5.12}$$

Then, together with (5.9)-(5.12) and the definition of $\|\nabla\|$ in (3.8), we get

$$
\begin{aligned}
&2\big(H(\mu_h^n-\mu_h^{n+1}),\mu_h^n-\nu_h\big)-\|\mu_h^n-\mu_h^{n+1}\|_H^2\\
&\geq 2\gamma\big(M(\mu_h^n-\bar{\mu}_h^n),\mu_h^n-\bar{\mu}_h^n\big)-\gamma^2\big(M(\mu_h^n-\bar{\mu}_h^n),H^{-1}M(\mu_h^n-\bar{\mu}_h^n)\big)\\
&\quad+2\gamma\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big)\\
&=(2\gamma-\gamma^2)\|\mu_h^n-\bar{\mu}_h^n\|_H^2-2(1+\theta)(\gamma-\gamma^2)\big(\nabla(u_h^n-\bar{u}_h^n),p_h^n-\bar{p}_h^n\big)\\
&\quad-\gamma^2\Big(\tau\|\mathrm{div}(p_h^n-\bar{p}_h^n)\|^2+\theta^2\frac{\tau}{\sigma}\|\nabla(u_h^n-\bar{u}_h^n)\|^2\Big)+2\gamma\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big)\\
&\geq\Big((2\gamma-\gamma^2)-\big((1+\theta)(\gamma-\gamma^2)+\theta^2\gamma^2\big)\frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\frac{1}{\tau}\|u_h^n-\bar{u}_h^n\|_{L^2(\Omega)}^2\\
&\quad+\Big(\big((2\gamma-\gamma^2)-(1+\theta)(\gamma-\gamma^2)\big)-\gamma^2\frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\frac{\sigma}{\tau}\|p_h^n-\bar{p}_h^n\|_{L^2(\Omega)}^2\\
&\quad+2\gamma\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big)\\
&\geq\gamma^2\Big(1-\frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\|\mu_h^n-\bar{\mu}_h^n\|_H^2+2\gamma\big(F(\nu_h),\bar{\mu}_h^n-\nu_h\big),\\
&\forall\,\nu_h\in\mathcal{S}^1(\mathcal{T}_h)\times\mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2).
\end{aligned}
\tag{5.13}
$$

Therefore, because of (5.11) and (5.12), using Cauchy-Schwarz inequality, the definition of $\|\nabla\|$ in (3.8) and the condition (5.2), we can derive

$$
\begin{aligned}
\|\mu_h^n-\mu_h^{n+1}\|_H^2&=\gamma^2\Big(\|\mu_h^n-\bar{\mu}_h^n\|_H^2+\tau\|\mathrm{div}(p_h^n-\bar{p}_h^n)\|^2+\theta^2\frac{\tau}{\sigma}\|\nabla(u_h^n-\bar{u}_h^n)\|^2\\
&\qquad-2(1+\theta)\big(\nabla(u_h^n-\bar{u}_h^n),p_h^n-\bar{p}_h^n\big)\Big)\\
&\leq\gamma^2\Big(1+\big(\theta^2+1+\theta\big)\frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\frac{1}{\tau}\|u_h^n-\bar{u}_h^n\|_{L^2(\Omega)}^2\\
&\qquad+\gamma^2\Big(1+\frac{\tau^2\|\nabla\|^2}{\sigma}+(1+\theta)\Big)\frac{\sigma}{\tau}\|p_h^n-\bar{p}_h^n\|_{L^2(\Omega)}^2\\
&\leq 4\gamma^2\|\mu_h^n-\bar{\mu}_h^n\|_H^2.
\end{aligned}
\tag{5.14}
$$

Then the result is obtained from (5.13) and (5.14). □

Using the result in the above lemma, we can easily derive that the sequence generated by Algorithm 2 is strictly contractive with respect to the solution set of VI (2.2). We summarize it in the following theorem.

Theorem 5.1 (Contraction)**.** *Let μ_h be the solution of (1.8) and $\{\mu_h^{n+1}\}$ be the sequence generated by Algorithm 2 with $\theta\in[-1,1]$ under the condition (5.2). Then we have*

$$
\|\mu_h^{n+1}-\mu_h\|_H^2\leq\|\mu_h^n-\mu_h\|_H^2-\frac{1}{4}\Big(1-\frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\|\mu_h^n-\mu_h^{n+1}\|_H^2.
\tag{5.15}
$$

Proof. Obviously, we have

$$
\begin{aligned}
\|\mu_h^{n+1}-\nu_h\|_H^2&=\|\mu_h^n-\nu_h-(\mu_h^n-\mu_h^{n+1})\|_H^2\\
&=\|\mu_h^n-\nu_h\|_H^2-2\big(H(\mu_h^n-\mu_h^{n+1}),\mu_h^n-\nu_h\big)+\|\mu_h^n-\mu_h^{n+1}\|_H^2.
\end{aligned}
\tag{5.16}
$$

Applying the result (5.6) in Lemma 5.1 to (5.16), setting $\nu_h=\mu_h$ and noticing $\big(F(\mu_h),\bar{\mu}_h^n-\mu_h\big)\geq 0$, we obtain (5.15). □

With the strict contraction property established in the last theorem, it becomes easy to prove the convergence for Algorithm 2. The convergence of Algorithm 2 is summarized in the following theorem.

Theorem 5.2 (Convergence). *Let $\{\mu_h^{n+1} = (u_h^{n+1}, p_h^{n+1})\}$ be the sequence generated by Algorithm 2 with $\theta \in [-1, 1]$ and $\gamma \in (0, 1]$ under the condition (5.2). Then, the sequence $\{u_h^{n+1}\}$ converges to the unique minimizer of energy functional E in $\mathcal{S}^1(\mathcal{T}_h)$.*

Proof. From (5.13) and (5.16), we obtain

$$\|\mu_h^{n+1} - \mu_h\|_H^2 \leq \|\mu_h^n - \mu_h\|_H^2 - \gamma^2\Big(1 - \frac{\tau^2\|\nabla\|^2}{\sigma}\Big)\|\mu_h^n - \bar{\mu}_h^n\|_H^2. \tag{5.17}$$

Then, the above inequality and (5.15) imply that

$$\lim_{n\to\infty}(\mu_h^n - \mu_h^{n+1}) = \lim_{n\to\infty}(\mu_h^n - \bar{\mu}_h^n) = 0.$$

Thus, we have $\lim_{n\to\infty}\mu_h^{n+1} = \lim_{n\to\infty}\bar{\mu}_h^n$. In addition, it follows from (5.17) that the sequence $\{\mu_h^{n+1}\}$ is bounded. Let μ_h^* be a cluster point of the sequence $\{\mu_h^{n+1}\}$, with (5.5), we derive that μ_h^* satisfies the VI (2.2). So, the sequence $\{u_h^{n+1}\}$ converges to the unique minimizer of energy functional E in $\mathcal{S}^1(\mathcal{T}_h)$ by (2.2), (2.4) and Lemma 2.1. □

Note that Algorithm 2 requires an additional correction step compared with Algorithm 1; but it ensures the condition (5.2) to be satisfied if $\tau \leq ch$ for some $c > 0$ as $\|\nabla\| \leq ch^{-1}$. This is a significantly relaxed condition compared with the requirement $\tau \leq ch^2$ for Algorithm 1. This is the main advantage of Algorithm 2. We will numerically verify its superiority in Section 6.2.

5.3 Convergence Rate

In this subsection, we establish the worst-case $O(\frac{1}{N})$ convergence rate in both the ergodic and nonergodic senses for Algorithm 2 with $\theta \in [-1, 1]$. Recall the lack of worst-case convergence rate in a nonergodic sense of Algorithm 1 with $\theta \in [-1, 1)$. Thus, the provable worst-case convergence rate in a nonergodic sense is another theoretical advantage of Algorithm 2.

5.3.1 Convergence Rate in the Ergodic Sense

We first establish the worst-case $O(\frac{1}{N})$ convergence rate in the ergodic sense for Algorithm 2 in the following theorem. The proof is analogous to that of Theorem 4.4.

Theorem 5.3 (Convergence rate in the ergodic sense). *Let $\{\mu_h^{n+1}\}$ be the sequence generated by Algorithm 2 with $\theta \in [-1, 1]$ and $\gamma \in (0, 1]$ under the condition (5.2). For any integer N, let $\bar{\mu}_N$ be defined as*

$$\bar{\mu}_N = \frac{1}{N+1}\sum_{n=0}^{N}\bar{\mu}_h^n.$$

Then we have

$$\big(F(\nu_h), \bar{\mu}_N - \nu_h\big) \leq \frac{1}{2\gamma(N+1)}\|\nu_h - \mu_h^0\|_H^2, \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{5.18}$$

Proof. It follows from (5.16) and (5.6) that

$$\begin{aligned}\big(F(\nu_h), \bar{\mu}_h^n - \nu_h\big) \leq \frac{1}{2\gamma}\big(\|\nu_h - \mu_h^n\|_H^2 - \|\nu_h - \mu_h^{n+1}\|_H^2\big),\\ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2).\end{aligned} \tag{5.19}$$

Summarizing the inequality (5.19) for the cases of $n = 0, 1, \cdots, N$, we obtain

$$\big(F(\nu_h), \sum_{n=0}^{N} \bar{\mu}_h^n - (N+1)\nu_h\big) \le \frac{1}{2\gamma}\big(\|\nu_h - \mu_h^0\|_H^2 - \|\nu_h - \mu_h^{N+1}\|_H^2\big), \quad \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \tag{5.20}$$

which leads to the result (5.18). □

5.3.2 Convergence Rate in a Nonergodic Sense

Now we establish the worst-case $O(\frac{1}{N})$ convergence rate in a nonergodic sense for Algorithm 2. The analysis is based on the strict contraction property (5.15) and the monotonicity of the sequence $\{\|\mu_h^n - \mu_h^{n+1}\|_H^2\}$.

Lemma 5.2. *Let $\{\mu_h^{n+1}\}$ be the sequence generated by Algorithm 2 with $\theta \in [-1, 1]$ and $\gamma \in (0, 1]$. Then, we have*

$$\|\mu_h^{n+1} - \mu_h^{n+2}\|_H^2 \le \|\mu_h^n - \mu_h^{n+1}\|_H^2. \tag{5.21}$$

Proof. Because of the optimality condition (5.5) of the prediction step, we have

$$\big(F(\bar{\mu}_h^n) + M(\bar{\mu}_h^n - \mu_h^n), \nu_h - \bar{\mu}_h^n\big) \ge 0, \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2), \tag{5.22}$$

and

$$\big(F(\bar{\mu}_h^{n+1}) + M(\bar{\mu}_h^{n+1} - \mu_h^{n+1}), \nu_h - \bar{\mu}_h^{n+1}\big) \ge 0, \ \forall\, \nu_h \in \mathcal{S}^1(\mathcal{T}_h) \times \mathcal{B}_1(\mathcal{L}^0(\mathcal{T}_h)^2). \tag{5.23}$$

Taking $\nu_h = \bar{\mu}_h^{n+1}$ in (5.22) and $\nu_h = \bar{\mu}_h^n$ in (5.23), and adding them together, we have

$$\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \bar{\mu}_h^n - \bar{\mu}_h^{n+1}\big) \ge 0. \tag{5.24}$$

Adding the term

$$\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big)\big)$$

to both sides of (5.24), it yields

$$\begin{aligned} &\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \mu_h^n - \mu_h^{n+1}\big) \\ &\ge \big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big)\big), \end{aligned} \tag{5.25}$$

together with which, it follows from (5.4) that

$$\begin{aligned} &\big(H\big((\mu_h^n - \mu_h^{n+1}) - (\mu_h^{n+1} - \mu_h^{n+2})\big), \mu_h^n - \mu_h^{n+1}\big) \\ &\ge \gamma\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big)\big). \end{aligned} \tag{5.26}$$

Now we use the identity

$$\|a\|_H^2 - \|b\|_H^2 = 2(H(a-b), a) - \|a-b\|_H^2$$

with $a = \mu_h^n - \mu_h^{n+1}$ and $b = \mu_h^{n+1} - \mu_h^{n+2}$ and thus get

$$\begin{aligned} &\|\mu_h^n - \mu_h^{n+1}\|_H^2 - \|\mu_h^{n+1} - \mu_h^{n+2}\|_H^2 \\ &= 2\big(H\big((\mu_h^n - \mu_h^{n+1}) - (\mu_h^{n+1} - \mu_h^{n+2})\big), \mu_h^n - \mu_h^{n+1}\big) \\ &\quad - \|(\mu_h^n - \mu_h^{n+1}) - (\mu_h^{n+1} - \mu_h^{n+2})\|_H^2. \end{aligned} \tag{5.27}$$

Then, it follows from (5.26), (5.27) and (5.4) that

$$\begin{aligned}
&\|\mu_h^n - \mu_h^{n+1}\|_H^2 - \|\mu_h^{n+1} - \mu_h^{n+2}\|_H^2 \\
&\geq 2\gamma\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big)\big) \\
&\quad - \|(\mu_h^n - \mu_h^{n+1}) - (\mu_h^{n+1} - \mu_h^{n+2})\|_H^2 \\
&= 2\gamma\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), \big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big)\big) \\
&\quad - \gamma^2\big(M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big), H^{-1}M\big((\mu_h^n - \bar{\mu}_h^n) - (\mu_h^{n+1} - \bar{\mu}_h^{n+1})\big)\big).
\end{aligned} \tag{5.28}$$

We can show that the right-hand side term of (5.28) is nonnegative, just as the same approach in (5.13). The result (5.21) is thus proved. □

Next we establish the worst-case $O(\frac{1}{N})$ convergence rate in a nonergodic sense for Algorithm 2. We summarize the result in the following theorem.

Theorem 5.4 (Convergence rate in a nonergodic sense)**.** *Let μ_h be the solution of (1.8) and $\{\mu_h^{n+1}\}$ be the sequence generated by Algorithm 2 with $\theta \in [-1, 1]$ and $\gamma \in (0, 1]$ under the condition (5.2). Then for any integer $N > 0$, it holds*

$$\|\mu_h^N - \mu_h^{N+1}\|_H^2 \leq \frac{1}{r(N+1)}\|\mu_h - \mu_h^0\|_H^2, \tag{5.29}$$

where r is

$$r = \frac{1}{4}\Big(1 - \frac{\tau^2\|\nabla\|^2}{\sigma}\Big) > 0.$$

Proof. It follows from (5.15) that

$$r\|\mu_h^n - \mu_h^{n+1}\|_H^2 \leq \|\mu_h - \mu_h^n\|_H^2 - \|\mu_h - \mu_h^{n+1}\|_H^2. \tag{5.30}$$

Summarizing the inequalities (5.30) for the cases $n = 0, \cdots, N$, we have

$$r\sum_{n=0}^{N}\|\mu_h^n - \mu_h^{n+1}\|_H^2 \leq \|\mu_h - \mu_h^0\|_H^2 - \|\mu_h - \mu_h^{N+1}\|_H^2. \tag{5.31}$$

From the result (5.21) of Lemma 5.2, we know that $\|\mu_h^n - \mu_h^{n+1}\|_H^2$ is monotonically non-increasing. Therefore, it yields

$$(N+1)\|\mu_h^N - \mu_h^{N+1}\|_H^2 \leq \sum_{n=0}^{N}\|\mu_h^n - \mu_h^{n+1}\|_H^2. \tag{5.32}$$

Then the assertion (5.29) is obtained by (5.31) and (5.32). □

6 Numerical Experiments

In this section, we report some preliminary numerical results to show the efficiency of the proposed algorithms. The rationale of considering the general primal-dual scheme (1.11) and the new primal-dual-based prediction-correction scheme is thus verified. Our main purpose is to illustrate: 1) the combination factor $\theta \neq 1$ sometimes can accelerate the convergence of Algorithm 1 with $\theta = 1$; and 2) Algorithm 2 with a relaxed requirement on τ could be numerically faster than Algorithm 1. Note that our work uses the finite element method to discretize the saddle-point reformulation of (1.1), and it is not comparable with those work (e.g. [18, 46, 47]) using finite difference discretization. Thus no comparison results with these finite difference discretization work are reported below.

All codes were written in C++ based on the finite element library `AFEPack`[1], and all experiments were run on a Linux desktop with i5-4570s Intel 2.9GHz four Processors and 8GB Memory. The stopping criterion for implementing Algorithms 1 and 2 is throughout chosen as

$$\frac{\|u_h^{n+1} - u_h^n\|_{L^2(\Omega)}}{\|u_h^{n+1}\|_{L^2(\Omega)}} \leq \texttt{Tol},$$

with the tolerance $\texttt{Tol} > 0$.

6.1 Numerical Results for Algorithm 1

We first verify that the combination factor $\theta \neq 1$ may accelerate the convergence of Algorithm 1 with $\theta = 1$. Recall that Algorithm 1 with $\theta = 1$ is the method considered in [8]. The example to be tested for this purpose is similar as the one tested in [8].

Example 6.1. *Let $B(0,r) := \{x \in \mathbb{R}^2 : |x| \leq r\}$, Ω be a regular octagon inscribed in the circle $B(0,0.5)$, $\alpha = 200$, and $g = g_0 + \delta\xi_h$ with $g_0 = \iota_{B(0,0.2)}$, which is the indicator function of $B(0,0.2)$, and ξ_h is a mesh-dependent perturbation function.*

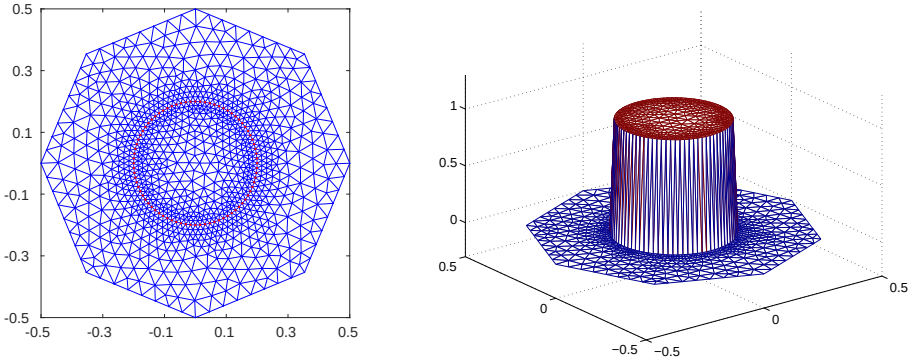


Figure 1: Triangular mesh over Ω (left) and the function g_0 (right) for Example 6.1.

The domain Ω is partitioned by the triangulation mesh $\mathcal{T}_h$ with 1023 nodes and 1980 elements, as shown in the left part of Figure 1. The right part of Figure 1 shows the plot of g_0 over the mesh $\mathcal{T}_h$. The discretized function $g_h \in \mathcal{L}^0(\mathcal{T}_h)$ is defined by $g_h|_T = g_0(x_T) + \delta\xi_h|_T$ for each $T \in \mathcal{T}_h$, where the perturbation function $\xi_h \in \mathcal{L}^0(\mathcal{T}_h)$ is a normally distributed random value in each element $T \in \mathcal{T}_h$. Note that using our mesh, we have $1/\|\nabla\|^2 \approx 1.0 \times 10^{-5}$ and empirically we use $1/\|\nabla\|^2 = 1.0 \times 10^{-5}$ to determine τ in our experiments. The initial guess u_h^0 is taken as the projection of function g_h onto the finite element space $\mathcal{S}^1(\mathcal{T}_h)$, and p_h^0 is initialized as zero function. The tolerance in the stopping criterion is set as $\texttt{Tol} = 1.0 \times 10^{-4}$ in the experiments for Example 6.1.

To see the effectiveness of the combination factor θ, for Algorithm 1 we test the cases of $\theta \in [-1, 1]$ with an equal distance of 0.1 and plot the iteration numbers and the values of the step size τ in Figure 2. From this figure, we see that some cases of $\theta \in [-0.5, 1)$ require less iterations. Thus the numerical efficiency of Algorithm 1 with $\theta \neq 1$ is demonstrated. We can also observe in Figure 2 that the value of τ plays a key role for the total iteration number of Algorithm 1 under the same stopping criterion, a larger value of τ satisfying the convergence condition leads to less iteration numbers. It should be noticed that the largest step size τ to guarantee the convergence for $\theta \in [-1, 1)$ can be larger than that of $\theta = 1.0$ when the mesh size h is not

[1] http://dsec.pku.edu.cn/~rli/source_code/AFEPack.tar.gz

small enough, although $\tau = O(h)$ for $\theta = 1$ and $\tau = O(h^2)$ for $\theta \in [-1, 1)$ are required to guarantee the convergence, just as shown in Figure 2, that the step size τ associated with $\theta \in [-0.5, 1)$ is larger than that with $\theta = 1.0$.

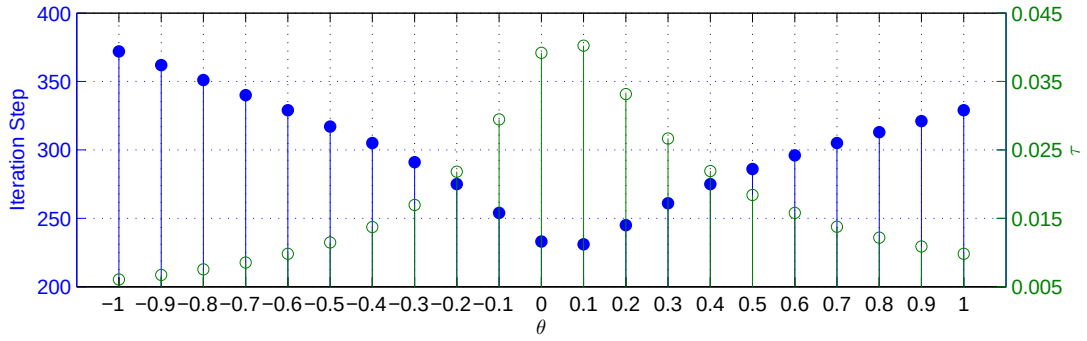


Figure 2: Iteration step number ('•') and step size τ ('∘') with different θ for Example 6.1 with noise level $\delta = 10\%$ by Algorithm 1 with $\sigma = 10.0$.

Notice that the condition (3.1) can be rewritten as

$$\tau < 2\frac{\sigma}{\|\nabla\|^2} \Big/ \left(\frac{(1-\theta)^2}{2\alpha} + \sqrt{\frac{(1-\theta)^4}{4\alpha^2} + 4\theta^2 \frac{\sigma}{\|\nabla\|^2}} \right).$$

As suggested by one referee, based on this inequality, we can seek a numerically "optimal" value of θ in sense of maximizing the function $\zeta(\theta)$ within the interval $\theta \in [-1, 1]$ where

$$\zeta(\theta) := 2\frac{\sigma}{\|\nabla\|^2} \Big/ \left(\frac{(1-\theta)^2}{2\alpha} + \sqrt{\frac{(1-\theta)^4}{4\alpha^2} + 4\theta^2 \frac{\sigma}{\|\nabla\|^2}} \right). \tag{6.1}$$

We define by θ^* the numerical approximation to the value of θ that maximizes $\zeta(\theta)$ approached by implementing Newton method; and Algorithm 1 with $\theta = \theta^*$ is denoted by "Alg1(θ^*)". In the following, we compare Alg1(θ^*) with Algorithm 1 with $\theta = 1$ ("denoted by Alg1(1)"), i.e., the method in [8]. For both Alg1(1) and Alg1(θ^*), we take $\tau = 0.98\zeta(\theta)$.

In Table 1, we compare the iteration numbers of Alg1(1) and Alg1(θ^*) for Example 6.1 with different noise levels. For $\sigma = 1.0, 2.0, 5.0, 10.0, 20.0$, as listed in this table, the values of θ^* are 0.303336, 0.199953, 0.10102, 0.0556989, 0.029435, respectively. Since the computation per iteration is the same for Alg1(1) and Alg1(θ^*), we just compare their iteration numbers and omit the comparison in computing time. According to this table, we see improvements of Algorithm 1 with the well-chosen parameter θ^* over the special case $\theta = 1$ in [8], especially for relatively large parameter σ and noise level δ. Additionally, the numerical results in Table 1 reveal that the case of $\theta = 1.0$ seems sensitive to the parameter σ and the noise level of input data, while the case with θ^* is more robust.

In Table 2, we report the comparison of the iteration numbers for Alg1(1) and Alg1(θ^*). We focus on the case of Example 6.1 with $\sigma = 10.0$ and test different noise levels and triangular meshes: "`TVo`" (1023 nodes and 1980 elements), "`TVo1`" (2126 nodes and 4154 elements), "`TVo2`" (3667 nodes and 7204 elements), "`TVo3`" (5621 nodes and 11080 elements). Note that "`TVo`" denotes the mesh shown in Figure 1; "`TVo1`","`TVo2`","`TVo3`" are refined meshes with approximate values of $1/\|\nabla\|^2$ as 4.4×10^{-6}, 2.5×10^{-5} and 1.5×10^{-5}, respectively. These experimental results verify acceleration of Algorithm 1 via choosing the combination parameter $\theta \neq 1$.

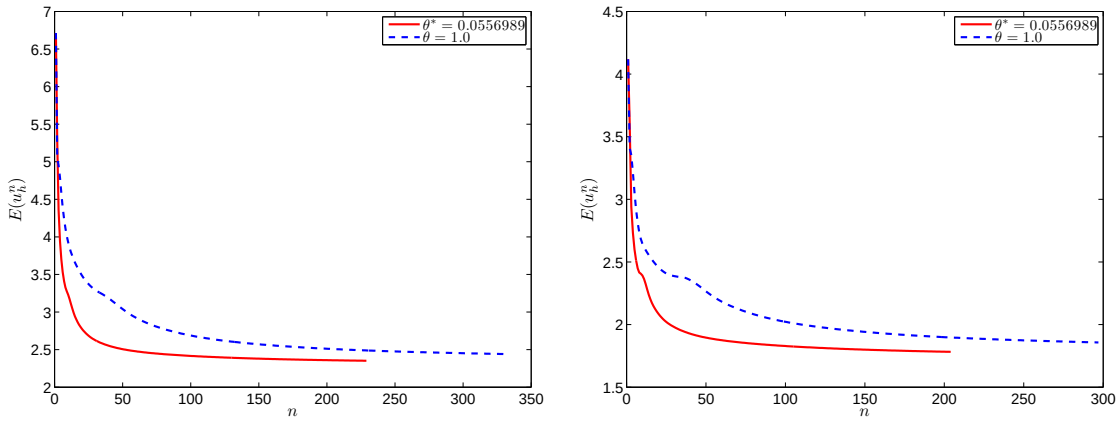
Table 1: Comparison in iteration numbers of Alg1(1) and Alg1(θ^*) for Example 6.1 with different noise levels.

	$\delta = 20\%$		$\delta = 10\%$		$\delta = 5\%$		$\delta = 1\%$	
σ	Alg1(1)	Alg1(θ^*)	Alg1(1)	Alg1(θ^*)	Alg1(1)	Alg1(θ^*)	Alg1(1)	Alg1(θ^*)
1.0	317	247	252	207	224	184	218	181
2.0	363	259	276	216	245	192	236	189
5.0	407	274	305	225	272	200	262	196
10.0	434	270	329	228	296	203	285	199
20.0	466	273	359	230	325	205	313	200

Table 2: Comparison in iteration numbers of Alg1(1) and Alg1(θ^*) with $\sigma = 10.0$ for Example 6.1 with different meshes and noise levels.

	$\delta = 20\%$		$\delta = 10\%$		$\delta = 5\%$		$\delta = 1\%$	
mesh	Alg1(1)	Alg1(θ^*)	Alg1(1)	Alg1(θ^*)	Alg1(1)	Alg1(θ^*)	Alg1(1)	Alg1(θ^*)
`TVo`	434	270	329	228	296	203	285	199
`TVo1`	426	300	365	268	343	259	334	258
`TVo2`	453	350	393	311	373	298	364	294
`TVo3`	448	363	408	336	397	330	394	330

In Figure 3, we plot the values of $E(u_h^n)$ at the iterations u_h^n for the cases of $\delta = 10\%$ and 5%. The curves of this figure further show that the energy decreases more quickly for Algorithm 1 with θ^* than $\theta = 1.0$. The acceleration of Algorithm 1 with $\theta \neq 1$ is thus verified. We show the iterations u_h^n for the cases of $\theta^* = 0.0556989$ and $\theta = 1.0$ in Figure 4, the cases with $n = 0, 20, 40$ and n^{stop} are listed from top to bottom, where n^{stop} stands for the scenarios where the iteration is terminated, i.e., the iteration numbers are 228 and 329 for the cases of θ^* and $\theta = 1$, respectively.



(a) noise level $\delta = 10\%$ (b) noise level $\delta = 5\%$

Figure 3: Energy $E(u_h^n)$ of iterations u_h^n by Algorithm 1 with $\theta^* = 0.0556989$ and $\theta = 1.0$, respectively, for Example 6.1 with $\sigma = 10.0$.

We also notice that there are a series of papers (e.g., [29, 34, 33, 41]) that consider finding numerical

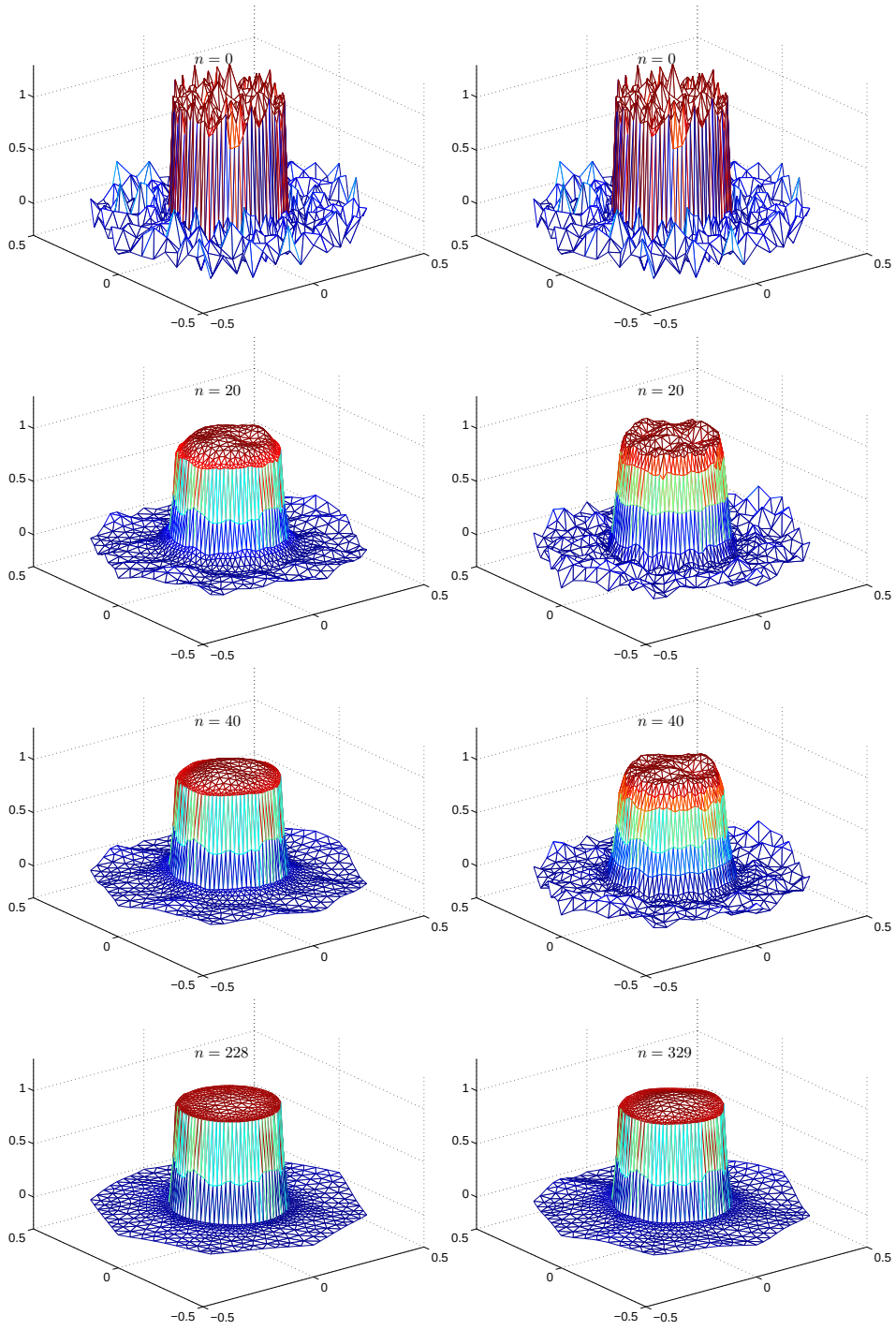


Figure 4: Iterations u_h^n with $n = 0, 20, 40, n^{stop}$ by Algorithm 1 with $\sigma = 10.0$ for optimal $\theta^* = 0.0556989$ (left) and $\theta = 1.0$ (right) for Example 6.1 with $\delta = 10\%$.

solutions of the model (1.1) by solving its regularized L^2 gradient flow, i.e.,

$$\partial_t u - \operatorname{div}\left(\frac{\nabla u}{\sqrt{|\nabla u|^2+\lambda}}\right) + \alpha(u-g) = 0. \tag{6.2}$$

This approach has been shown to be able to overcome the singularity of the diffusion term in (1.4). Particularly, in [34, 33], the following fully discretized implicit finite element scheme (IFEM) was proposed for solving (6.2):

$$\left(\frac{u_h^{n+1}-u_h^n}{\tau}, v_h\right) + \left(\frac{\nabla u_h^{n+1}}{\sqrt{|\nabla u_h^{n+1}|^2+\lambda}}, \nabla v_h\right) + \alpha(u_h^{n+1}-g, v_h) = 0, \ \ \forall\, v_h \in \mathcal{S}^1(\mathcal{T}_h). \tag{6.3}$$

Since (6.3) is a nonlinear system and it is generally not easy to solve it, the following semi-implicit finite element scheme (SIFEM) was proposed in [29, 41]:

$$\left(\frac{u_h^{n+1}-u_h^n}{\tau}, v_h\right) + \left(\frac{\nabla u_h^{n+1}}{\sqrt{|\nabla u_h^{n}|^2+\lambda}}, \nabla v_h\right) + \alpha(u_h^{n+1}-g, v_h) = 0, \ \ \forall\, v_h \in \mathcal{S}^1(\mathcal{T}_h), \tag{6.4}$$

which results in a linear equation with respect to u_h^{n+1}.

In the following, we shall compare Algorithm 1 with the SIFEM (6.4) as well to further verify its efficiency. We take $\lambda = 1.0 \times 10^{-3}$ for the SIFEM (6.4) and the same step size τ as Algorithm 1.

Table 3: Comparison in computing time in seconds of Alg1(θ^*) and the SIFEM (6.4) for Example 6.1 with different noise levels.

	$\delta = 20\%$		$\delta = 10\%$		$\delta = 5\%$		$\delta = 1\%$	
σ	Alg1(θ^*)	SIFEM	Alg1(θ^*)	SIFEM	Alg1(θ^*)	SIFEM	Alg1(θ^*)	SIFEM
0.01	0.63	3.14	0.57	2.50	0.52	2.51	0.51	2.45
0.05	0.67	2.30	0.62	1.84	0.56	1.76	0.57	1.77
0.1	0.73	2.09	0.68	1.71	0.62	1.70	0.61	1.66
0.2	0.84	1.53	0.74	1.22	0.67	1.19	0.66	1.18
0.5	0.98	1.19	0.84	1.03	0.74	0.99	0.73	1.00
1.0	1.04	1.13	0.87	0.91	0.78	0.93	0.77	0.94

In Table 3, we list the computing time in seconds for Alg1(θ^*) and the SIFEM (6.4) for Example 6.1 with different noise levels. For the cases of $\sigma = 0.01, 0.05, 0.1, 0.5, 1.0$, the values of θ^* are $0.881256, 0.754343, 0.672078, 0.420204, 0.303336$, respectively. Table 3 shows that Algorithm 1 performs much more efficiently than the SIFEM (6.4) for Example 6.1. Since Algorithm 1 and the SIFEM (6.4) are in different natures and the computation per iteration is different, we just compare them in terms of the computing time and omit the iteration numbers.

6.2 Numerical Results for Algorithm 2

Then, we test Example 6.2 to verify that Algorithm 2 with a relaxed requirement on τ could be numerically faster than Algorithm 1. The efficiency of Algorithm 2 is demonstrated by some comparisons with Algorithm 1.

Example 6.2. *Let $\alpha = 400$, $g = g_0 + \delta\xi_h$ and ξ_h be a mesh-dependent perturbation function, where g_0 be the solution at $t = 1.0$ of the 2D Allen-Cahn equation [2] over $\Omega = (0,1)^2$ subjected to periodic boundary condition,*

$$\partial_t u = D\big(\epsilon\Delta u - \frac{1}{\epsilon}F'(u)\big), \tag{6.5}$$

where $F(u) = \frac{1}{4}(u^2-1)^2$*, the initial value be taken as the following random initial value*

$$u(x,0) = 0.05(2\texttt{rand} - 1).$$

We obtain g_0 by solving the Allen-Cahn equation (6.5), which describes the process of phase separation in multi-component alloy systems, including order-disorder transitions. The image of g_0 over the mesh $\mathcal{T}_h$ with $10,201$ nodes and $20,000$ elements is shown in Figure 5, which represents the concentration of two metallic components of the alloy at time $t = 1$, respectively separated in the red and blue regions. It is piecewise constant and exhibits jumps on the edges, and we consider it as an appropriate data for the minimization model (1.1). The perturbation function $\xi_h \in \mathcal{S}^1(\mathcal{T}_h)$ evaluated at each node of mesh $\mathcal{T}_h$ is a random value sampled from the normal distribution, the noise level δ is 0.2, the initial guess u_h^0 is set by function g_h and p_h^0 is chosen as 0.

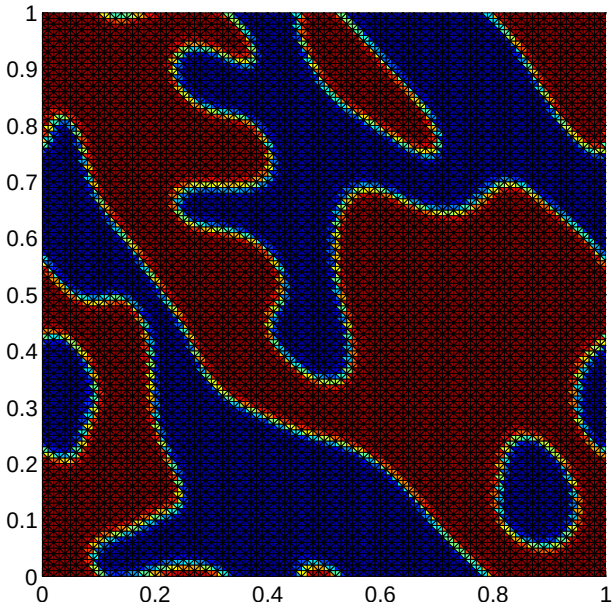


Figure 5: The function g_0 over the uniformly triangular mesh with square edge length $1/100$.

We fix $\sigma = 0.1$ and $\gamma = 1.0$ for Algorithms 1 and 2; and focus on some cases with $\theta \in [-0.9, -0.2]$ to compare the difference of these two algorithms (because the difference of these two algorithms for Example 6.2 is less significant when θ is not that small, e.g., $\theta \in [-0.1, 1]$). In Table 4, the iteration numbers and computing time in seconds are listed for the cases of $\theta \in [-0.9, -0.2]$ with an equal distance of 0.1, where the tolerance in the stopping criterion is set as $\texttt{Tol} = 1.0 \times 10^{-4}$. According to the table, we see that Algorithm 2 converges with much larger values of τ than those for Algorithm 1. This coincides with our theoretical assertion of $\tau \sim O(h^2)$ for Algorithm 1 and $\tau \sim O(h)$ for Algorithm 2; see (3.1) and (5.2), respectively. Moreover, the results in Table 4 show that Algorithm 1 seems to be more sensitive if the value of τ is near its upper bound theoretically given by (3.1). The comparison of iteration numbers and computing time in seconds is displayed in Figure 6. We test the cases of $\theta = -0.2, -0.4, -0.6, -0.8$ with more values of the step size τ; and plot the results in Figure 7. This figure shows that the iteration numbers and computing time of Algorithm 1 decrease first and then increase once it is convergent; while those of Algorithm 2 are monotonically increasing with small values of the step size τ. In Figure 8, we plot the input u_h^0 and outputs (first row: input u_h^0; second row: output by Algorithm 1; third row: output by Algorithm 2) when $\theta = -0.4$ (left) and -0.8 (right) for Example 6.2 with $\texttt{Tol} = 1.0 \times 10^{-5}$.

Table 4: Iteration step numbers and CPU time in seconds obtained by Algorithm 1 and 2 for Example 6.2 with $\theta = -0.2, \cdots, -0.9$ and different step size τ ('-' means not convergent).

$\theta = -0.2$					$\theta = -0.3$					$\theta = -0.4$					$\theta = -0.5$				
	Alg 1		Alg 2			Alg 1		Alg 2			Alg 1		Alg 2			Alg 1		Alg 2	
τ	step	CPU(s)	step	CPU(s)	τ	step	CPU(s)	step	CPU(s)	τ	step	CPU(s)	step	CPU(s)	τ	step	CPU(s)	step	CPU(s)
1.0/1200.0	-	-	-	-	1.0/1200.0	-	-	-	-	1.0/1200.0	-	-	-	-	1.0/1200.0	-	-	-	-
1.0/1600.0	-	-	**108**	**5.41**	1.0/1600.0	-	-	**108**	**5.43**	1.0/1600.0	-	-	**108**	**5.38**	1.0/1600.0	-	-	**108**	**5.37**
1.0/2000.0	393	16.51	118	6.04	1.0/2000.0	-	-	118	5.97	1.0/2400.0	-	-	126	6.32	1.0/2400.0	-	-	126	6.41
1.0/2400.0	218	9.25	126	6.25	1.0/2400.0	-	-	126	6.31	1.0/3200.0	-	-	141	7.04	1.0/3200.0	-	-	141	7.04
1.0/2800.0	186	7.95	133	6.66	1.0/3200.0	362	15.18	141	7.07	1.0/4000.0	583	24.66	154	7.64	1.0/4800.0	947	39.82	166	8.17
1.0/3200.0	178	7.57	141	7.03	1.0/3600.0	281	11.88	147	7.40	1.0/4800.0	345	14.49	166	8.23	1.0/6000.0	400	16.84	183	9.34
1.0/3600.0	**175**	**7.52**	147	7.36	1.0/4000.0	250	10.68	154	7.69	1.0/5600.0	289	12.17	177	8.70	1.0/7000.0	336	14.11	197	9.66
1.0/4000.0	179	7.60	154	7.59	1.0/4400.0	235	9.92	160	8.05	1.0/6400.0	283	11.99	189	9.26	1.0/8000.0	326	13.68	212	10.65
1.0/4400.0	183	7.78	160	7.96	1.0/4800.0	231	9.84	166	8.36	1.0/7200.0	**280**	**11.79**	200	9.81	1.0/9000.0	**322**	**13.81**	227	11.12
1.0/4800.0	189	8.41	166	8.26	1.0/5200.0	231	9.89	171	8.46	1.0/8000.0	284	12.16	212	10.45	1.0/10000.0	333	14.48	241	11.92
1.0/5200.0	195	8.40	171	8.53	1.0/5600.0	**230**	**9.72**	177	8.72	1.0/8800.0	293	12.36	224	10.93	1.0/11000.0	338	14.17	256	12.62
1.0/5600.0	201	8.60	177	8.77	1.0/6000.0	234	9.87	183	9.12	1.0/9600.0	302	12.68	235	11.52	1.0/12000.0	350	14.82	270	13.35
1.0/6000.0	207	8.76	183	9.10	1.0/6400.0	238	10.04	189	9.36	1.0/10400.0	311	13.18	247	12.05	1.0/13000.0	363	15.22	285	13.86
1.0/6400.0	215	9.10	189	9.42	1.0/6800.0	242	10.20	194	9.53	1.0/11200.0	320	13.45	258	12.56	1.0/14000.0	376	15.79	299	14.60
1.0/6800.0	220	9.44	194	9.71	1.0/7200.0	245	10.40	200	9.92	1.0/12000.0	332	14.04	270	13.20	1.0/15000.0	386	16.23	313	15.46

$\theta = -0.6$					$\theta = -0.7$					$\theta = -0.8$					$\theta = -0.9$				
	Alg 1		Alg 2			Alg 1		Alg 2			Alg 1		Alg 2			Alg 1		Alg 2	
τ	step	CPU(s)	step	CPU(s)	τ	step	CPU(s)	step	CPU(s)	τ	step	CPU(s)	step	CPU(s)	τ	step	CPU(s)	step	CPU(s)
1.0/1200.0	-	-	-	-	1.0/1200.0	-	-	-	-	1.0/1200.0	-	-	-	-	1.0/1200.0	-	-	-	-
1.0/1600.0	-	-	**108**	**5.46**	1.0/1600.0	-	-	**108**	**5.44**	1.0/1600.0	-	-	**108**	**5.39**	1.0/1600.0	-	-	**109**	**5.48**
1.0/3800.0	-	-	151	7.48	1.0/4400.0	-	-	160	7.95	1.0/4800.0	-	-	166	8.17	1.0/5400.0	-	-	174	8.57
1.0/6000.0	741	31.45	183	9.10	1.0/7200.0	724	30.47	200	9.79	1.0/8000.0	899	37.80	212	10.45	1.0/9200.0	875	36.45	229	11.51
1.0/7200.0	453	19.01	200	10.02	1.0/8400.0	500	21.17	218	10.67	1.0/9200.0	592	24.80	229	11.16	1.0/10400.0	621	26.45	247	12.24
1.0/8400.0	385	16.15	218	10.93	1.0/9600.0	426	17.95	235	11.58	1.0/10400.0	493	20.87	247	12.27	1.0/11600.0	532	22.48	264	13.16
1.0/9600.0	372	15.63	235	11.54	1.0/10800.0	418	17.57	253	12.38	1.0/11600.0	454	19.43	264	12.86	1.0/12800.0	498	20.80	282	13.88
1.0/10800.0	**366**	**15.41**	253	12.51	1.0/12000.0	**406**	**16.98**	270	13.17	1.0/12800.0	451	18.88	282	13.87	1.0/14000.0	493	20.63	299	14.70
1.0/12000.0	378	15.88	270	13.18	1.0/13200.0	416	17.54	288	14.07	1.0/14000.0	**447**	**18.71**	299	14.53	1.0/15200.0	**485**	**20.33**	316	15.72
1.0/13200.0	386	16.24	288	14.03	1.0/14400.0	422	18.39	305	15.02	1.0/15200.0	455	19.02	316	15.34	1.0/16400.0	490	20.69	333	16.44
1.0/14400.0	399	16.93	305	14.86	1.0/15600.0	432	18.31	322	15.91	1.0/16400.0	462	19.64	333	16.58	1.0/17600.0	495	20.78	350	17.33
1.0/15600.0	412	17.23	322	15.68	1.0/16800.0	444	18.81	339	16.84	1.0/17600.0	473	20.00	350	17.28	1.0/18800.0	505	21.35	367	18.06
1.0/16800.0	426	17.85	339	16.47	1.0/18000.0	456	19.12	355	17.43	1.0/18800.0	481	20.59	367	17.86	1.0/20000.0	511	21.40	384	18.90
1.0/18000.0	442	18.53	355	17.24	1.0/19200.0	471	20.12	372	18.28	1.0/20000.0	497	20.81	384	18.61	1.0/21200.0	527	22.32	400	19.35
1.0/19200.0	455	19.03	372	18.05	1.0/20400.0	483	20.71	389	19.03	1.0/21200.0	506	21.29	400	19.62	1.0/22400.0	534	22.67	416	20.17

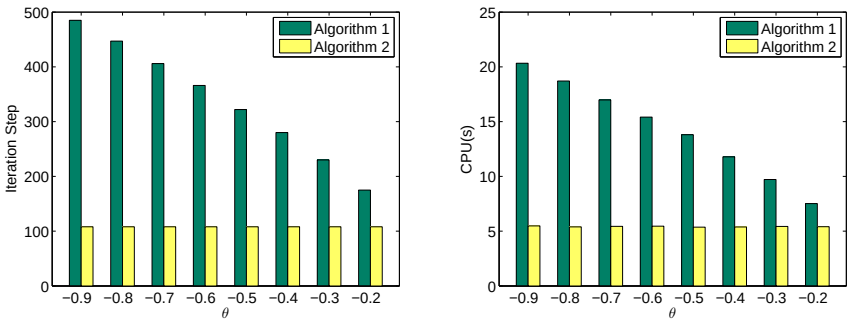


Figure 6: The iteration step (left) and CPU time (right) in seconds for Example 6.2 by Algorithm 1 and Algorithm 2 with $\theta = -0.9, \cdots, -0.2$ and their corresponding step size τ marked in Table 4.

7 Conclusions

In this paper, we focus on the application of primal-dual schemes to the saddle-point reformulation of a minimization model with total variation regularization. We consider the context of using the consistent finite element discretization and focus on the convergence study for some primal-dual schemes. We first generalize the primal-dual scheme in [8], and then prove its convergence and establish its worst-case convergence rate measured by the iteration complexity. Then, we propose a new primal-dual scheme in the prediction-correction framework, whose requirement of the step size τ can be significantly relaxed. This new primal-dual-based prediction-correction scheme keeps the same convergence and may perform better than the generalized primal-dual scheme. We report some preliminary numerical results to verify the theoretical assertions.

References

[1] Acar, R., Vogel, C.R.: Analysis of bounded variation penalty methods for ill-posed problems. Inverse Problems **10**(6), 1217–1229 (1994)

[2] Allen, S.M., Cahn, J.W.: A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening. Acta Metall. **27**(6), 1085–1095 (1979)

[3] Ambrosio, L., Fusco, N., Pallara, D.: Functions of Bounded Variation and Free Discontinuity Problems. Clarendon Press, Oxford (2000)

[4] Andreu, F., Ballester, C., Caselles, V., Mazón, J.M.: Minimizing total variation flow. Differential Integral Equations **14**(3), 321–360 (2001)

[5] Arrow, K.J., Hurwicz, L., Uzawa, H.: Studies in Linear and Non-linear Programming. Stanford University Press, Stanford (1958)

[6] Attouch, H., Buttazzo, G., Michaille, G.: Variational Analysis in Sobolev and BV Spaces: Applications to PDEs and Optimization. SIAM/MPS, Philadelphia, PA (2006)

[7] Aubert, G., Kornprobst, P.: Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations, 2nd edn. Springer, New York (2006)

[8] Bartels, S.: Total variation minimization with finite elements: convergence and iterative solution. SIAM J. Numer. Anal. **50**(3), 1162–1180 (2012)

[9] Bartels, S.: Broken Sobolev space iteration for total variation regularized minimization problems. IMA J. Numer. Anal. **36**(2), 493–502 (2016)

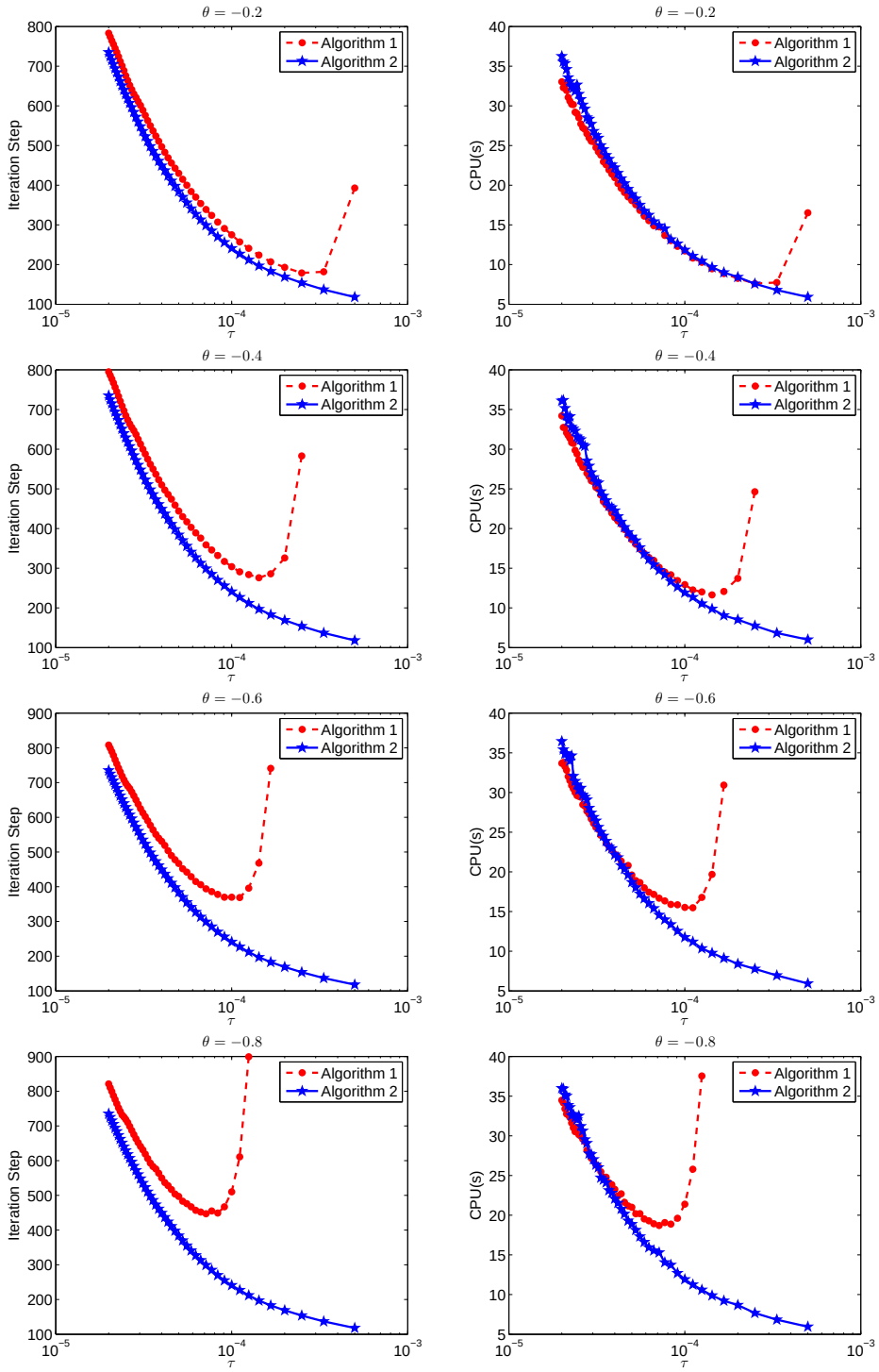


Figure 7: The iteration step (left) and CPU time (right) in seconds for Example 6.2 by Algorithm 1 and Algorithm 2 with $\theta = -0.2, -0.4, -0.6, -0.8$ and $\tau =$ `1./[1000:1000:50000]`.

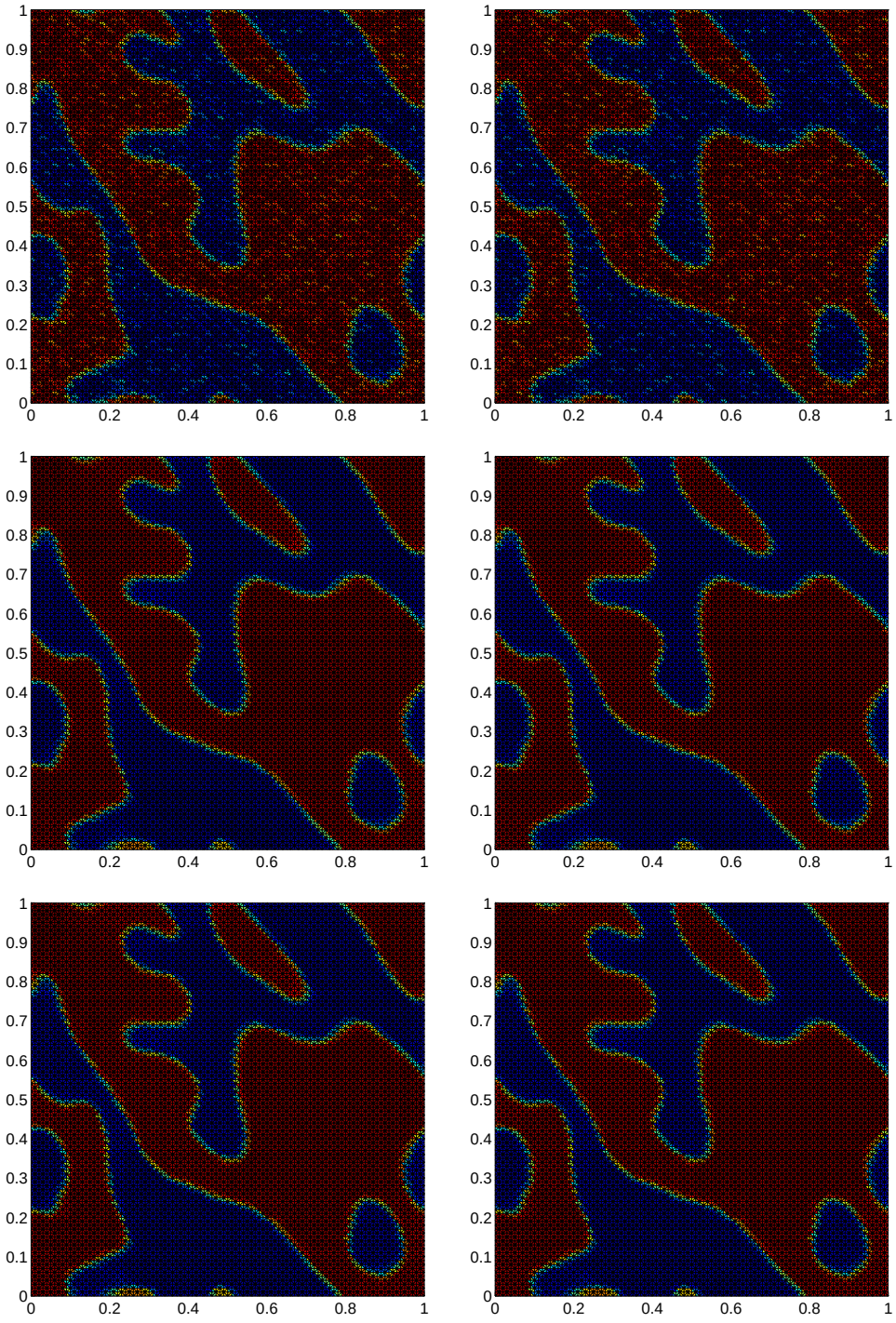


Figure 8: Input u_h^0 and Outputs for $\theta = -0.4$ (left) and -0.8 (right) for Example 6.2 with noise level $\delta = 0.2$ and $\texttt{Tol} = 1.0 \times 10^{-5}$. (First row: input u_h^0; Second row: Output by Algorithm 1; Third row: Output by Algorithm 2)

[10] Bartels, S., Nochetto, R.H., Salgado, A.J.: Discrete total variation flows without regularization. SIAM J. Numer. Anal. **52**(1), 363–385 (2014)

[11] Bellettini, G., Caselles, V., Novaga, M.: The total variation flow in $\mathbb{R}^n$. J. Differential Equations **184**(2), 475–525 (2002)

[12] Blum, E., Oettli, W.: Mathematische Optimierung. Springer, Berlin (1975)

[13] Boţ, R.I., Csetnek, E.R.: On the convergence rate of a forward-backward type primal-dual splitting algorithm for convex optimization problems. Optimization **64**(1), 5–23 (2015)

[14] Bramble, J.H., Pasciak, J.E., Vassilev, A.T.: Analysis of the inexact Uzawa algorithm for saddle point problems. SIAM J. Numer. Anal. **34**(3), 1072–1092 (1997)

[15] Brenner, S.C., Scott, L.R.: The Mathematical Theory of Finite Element Methods, 3rd edn. Springer, New York (2008)

[16] Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imaging Vision **20**(1-2), 89–97 (2004)

[17] Chambolle, A., Lions, P.-L.: Image recovery via total variation minimization and related problems. Numer. Math. **76**(2), 167–188 (1997)

[18] Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vision **40**(1), 120–145 (2011)

[19] Chan, R.H., Chen, K.: A multilevel algorithm for simultaneously denoising and deblurring images. SIAM J. Sci. Comput. **32**(2), 1043–1063 (2010)

[20] Chan, T.F., Mulet, P.: On the convergence of the lagged diffusivity fixed point method in total variation image restoration. SIAM J. Numer. Anal. **36**(2), 354–367 (1999)

[21] Chan, T.F., Tai, X.-C.: Identification of discontinuous coefficients in elliptic problems using total variation regularization. SIAM J. Sci. Comput. **25**(3), 881–904 (2003)

[22] Chan, T.F., Golub, G.H., Mulet, P.: A nonlinear primal-dual method for total variation-based image restoration. SIAM J. Sci. Comput. **20**(6), 1964–1977 (1999)

[23] Chavent, G., Kunisch, K.: Regularization of linear least squares problems by total bounded variation. ESAIM Control Optim. Calc. Var. **2**, 359–376 (1997)

[24] Chen, Z., Zou, J.: An augmented Lagrangian method for identifying discontinuous parameters in elliptic systems. SIAM J. Control Optim. **37**(3), 892–910 (1999)

[25] Dobson, D., Scherzer, O.: Analysis of regularized total variation penalty methods for denoising. Inverse Problems **12**(5), 601–617 (1996)

[26] Dobson, D.C., Santosa, F.: An image-enhancement technique for electrical impedance tomography. Inverse Problems **10**(2), 317–334 (1994)

[27] Dobson, D.C., Santosa, F.: Recovery of blocky images from noisy and blurred data. SIAM J. Appl. Math. **56**(4), 1181–1198 (1996)

[28] Dobson, D.C., Vogel, C.R.: Convergence of an iterative method for total variation denoising. SIAM J. Numer. Anal. **34**(5), 1779–1791 (1997)

[29] Elliott, C.M., Smitheman, S.A.: Numerical analysis of the TV regularization and H^{-1} fidelity model for decomposing an image into cartoon plus texture. IMA J. Numer. Anal. **29**(3), 651–689 (2008)

[30] Elman, H.C., Golub, G.H.: Inexact and preconditioned Uzawa algorithms for saddle point problems. SIAM J. Numer. Anal. **31**(6), 1645–1661 (1994)

[31] Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. SIAM J. Imaging Sci. **3**(4), 1015–1046 (2010)

[32] Facchinei, F., Pang, J.-S.: Finite-dimensional Variational Inequalities and Complementarity Problems. Vol. I. Springer, New York (2003)

[33] Feng, X.B., Prohl, A.: Analysis of total variation flow and its finite element approximations. ESAIM Math. Model. Numer. Anal. **37**(3), 533–556 (2003)

[34] Feng, X., von Oehsen, M., Prohl, A.: Rate of convergence of regularization procedures and finite element approximations for the total variation flow. Numer. Math. **100**(3), 441–456 (2005)

[35] Fortin, M., Glowinski, R.: Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary Value Problems. North-Holland Publishing Co., Amsterdam (1983)

[36] Glowinski, R.: Numerical Methods for Nonlinear Variational Problems. Springer, New York (1984)

[37] He, B.S., Yuan, X.M.: Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. SIAM J. Imaging Sci. **5**(1), 119–149 (2012)

[38] He, B.S., Yuan, X.M.: On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method. SIAM J. Numer. Anal. **50**(2), 700–709 (2012)

[39] He, B.S., Liu, H., Wang, Z., Yuan, X.M.: A strictly contractive Peaceman-Rachford splitting method for convex programming. SIAM J. Optim. **24**(3), 1011–1040 (2014)

[40] Keung, Y.L., Zou, J.: Numerical identifications of parameters in parabolic systems. Inverse Problems **14**(1), 83–100 (1998)

[41] Li, B., Sun, W.: Linearized FE approximations to a nonlinear gradient flow. SIAM J. Numer. Anal. **52**(6), 2623–2646 (2014)

[42] Marquina, A., Osher, S.: Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal. SIAM J. Sci. Comput. **22**(2), 387–405 (2000)

[43] Nemirovsky, A.S., Yudin, D.B.: Problem Complexity and Method Efficiency in Optimization. John Wiley & Sons, Inc., New York (1983)

[44] Nesterov, Y.: Gradient methods for minimizing composite functions. Math. Program. **140**(1), 125–161 (2013)

[45] Nesterov, Y.E.: A method for solving the convex programming problem with convergence rate $O(1/k^2)$. Dokl. Akad. Nauk SSSR **269**(3), 543–547 (1983). (In Russian. Translated in Soviet Math. Dokl., 27 (1983), pp. 372-376.)

[46] Ng, M.K., Qi, L., Yang, Y.-F., Huang, Y.-M.: On semismooth Newton-methods for total variation minimization. J. Math. Imaging Vision **27**(3), 265–276 (2007)

[47] Osher, S., Burger, M., Goldfarb, D., Xu, J., Yin, W.: An iterative regularization method for total variation-based image restoration. Multiscale Model. Simul. **4**(2), 460–489 (2005)

[48] Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 1762–1769 (2011)

[49] Queck, W.: The convergence factor of preconditioned algorithms of the Arrow-Hurwicz type. SIAM J. Numer. Anal. **26**(4), 1016–1030 (1989)

[50] Ring, W.: Structural properties of solutions to total variation regularization problems. ESAIM Math. Model. Numer. Anal. **34**(04), 799–810 (2000)

[51] Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D **60**, 259–268 (1992)

[52] Sapiro, G., Caselles, V.: Histogram modification via differential equations. J. Differential Equations **135**(2), 238–268 (1997)

[53] Strong, D., Chan, T.: Edge-preserving and scale-dependent properties of total variation regularization. Inverse Problems **19**(6), 165–187 (2003)

[54] Vogel, C.R., Oman, M.E.: Iterative methods for total variation denoising. SIAM J. Sci. Comput. **17**(1), 227–238 (1996)

[55] Zhang, J., Chen, K., Yu, B.: An iterative Lagrange multiplier method for constrained total-variation-based image denoising. SIAM J. Numer. Anal. **50**(3), 983–1003 (2012)

[56] Zhang, X., Burger, M., Osher, S.: A unified primal-dual algorithm framework based on Bregman iteration. J. Sci. Comput. **46**(1), 20–46 (2011)

[57] Zhu, M., Chan, T.F.: An efficient primal-dual hybrid gradient algorithm for total variation image restoration. CAM Report 08-34, UCLA, Los Angeles, CA (2008)

[58] Ziemer, W.P.: Weakly Differentiable Functions: Sobolev Spaces and Functions of Bounded Variation. Springer, New York (1989)

[59] Zulehner, W.: Analysis of iterative methods for saddle point problems: a unified approach. Math. Comp. **71**(238), 479–505 (2002)