Fall 2014

An iterative algorithm for variational data assimilation problems

Xin Shen

Department:

AN ITERATIVE ALGORITHM FOR VARIATIONAL DATA ASSIMILATION PROBLEMS

by

XIN SHEN

A THESIS

Presented to the Graduate Faculty of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

IN

MATHEMATICS

2014

Approved by

Yanzhi Zhang, Advisor
Xiaoming He, Co-Advisor
John Singler

ABSTRACT

Data assimilation is a very powerful and efficient tool to use collected raw data for improving model prediction in numerical weather forecasting, hydrology, and many other areas of geosciences. In this thesis, an iterative algorithm [23] of variational data assimilation with finite element method is utilized to study different models. One motivation for this fundamental mathematical study is to provide a potential tool for simulation of CO_2 sequestration by extending it to more realistic and sophisticated models in the future. The basic idea of variational data assimilation is to utilize the framework of optimal control problems. We apply the iterative algorithm with corresponding discretization formulation of the model equations to approximate the optimal control in the variational data assimilation problems. We conduct a group of comprehensive numerical experiments for both the second order parabolic equation and Stokes equation. These two models are critical to study Darcy's law and Stokes-Darcy problems for CO_2 sequestration, especially for the CO_2 storage in fractured reservoir and the leakage around the natural faults.

One key point for this method of data assimilation is the derivation of the adjoint models. For the convenience of computation, we discretize the adjoint models in the operator formulation into the corresponding discretized matrix formulation. We focus on the application of the iterative algorithm to the second order parabolic equation and Stokes equation with different numerical tests for the parameter sensitivity, convergence, accuracy, and efficiency of the algorithm.

At each step of the iteration, there are three major stages: solving original forward equation with the current control, solving backward adjoint equation with the observation and the current solution of the forward equation, and updating the control for the next iteration step. Finite elements are utilized for the spatial discretization, finite difference schemes are utilized for temporal discretization, and the conjugate gradient method is utilized for solving the control equation in order to update the control. The numerical results illustrate that the iterative algorithm is stable and efficient for variational data assimilation problems.

ACKNOWLEDGMENT

Foremost, I would like to express my sincere gratitude to my advisor Dr. Yanzhi Zhang and co-advisor Dr. Xiaoming He. Thank you for bringing me into this computational mathematics family. For the past two years, your guidance helped me with my research and writing of this thesis. Your patience, motivation and immense knowledge have a great impact on me. I could not imagine having better advisors and mentors for my master study.

Besides them, I also would like to thank Dr. John Singler for the encouragement, insightful comments and advice.

I want to thank Dr. Stephen L. Clark and Dr. V. A. Samaranayake for giving me this opportunity and providing this great academic environment for me to study.

Also, my sincere thanks goes to my dear classmates, Siwei Duo and Haowei Chen. They have helped me a lot in all aspects of my academic study. Also I thank Dr. Chuang Zheng, for providing the very helpful advice on my research and thesis.

Last but not the least, I would like to thank my parents for loving me, supporting me all the way, my dear friend Xue Yu for supporting me mentally and throughout my daily life. Without them, I will never become who I am.

This material is based upon work partially supported by the Department of Energy under Award Number DE-FE0009843.

TABLE OF CONTENTS

Page

LIST OF TABLES

1. INTRODUCTION OF DATA ASSIMILATION

In this section, we first introduce the background and the motivation of the data assimilation and then review the history and existing methods for data assimilation.

1.1. BACKGROUND AND MOTIVATION

Data assimilation is a very powerful and efficient tool to use collected raw data for improving model prediction in numerical weather forecasting, hydrology, and many other areas of geosciences.

One motivation for us to study data assimilation technique is to develop a fundamental mathematical tool for monitoring of CO_2 sequestration. Energy generation by use of fossil fuels produces large volumes of CO_2, which has been shown to have a significant undesirable impact on our environment [2]. Hence, it is envisioned to capture and sequester a substantial fraction of the produced CO_2 since subsurface geologic formations provide a potential long-term storage location for CO_2 sequestration [1, 3, 21]. Currently, industrial professionals and scientists are developing different methods to sequestrate and monitor the CO_2 in the deep geological formations. Several mechanisms are discussed by scientists to help keep the CO_2 trapped in the deep subsurface. The storage formation should be deep enough (typically at depths ranging from 1000 to 4000 meters [28]) to keep the CO_2 in the supercritical state, preventing it from arising into the shallower regions where it might do harm to the water resources or even escape to the atmosphere.

However, the inaccessibility and complexity of the potential storage formation and the sealing formations in the subsurface, the wide range of scales of variability, and the coupled nonlinear processes impose tremendous challenges to determine the transport and predict the fate of the stored CO_2 [20,24,29,30], especially the long term retention of the CO_2 in the geological formations. Therefore, it is critical to develop a robust, accurate long-term monitoring system for CO_2 sequestration.

Currently, an interdisciplinary team is currently working on a DOE project: "Robust Ceramic Coaxial Cable Down-Hole Sensors for Long-Term In Situ Monitoring of Geologic CO_2 Injection and Storage" (DE-FE0009843). Due to the low cost of ceramic coaxial cable sensors, a large array of sensors can be deployed in

the system to improve the accuracy and stability. One of the major tasks of this project is to improve model prediction based on the data measured by the developed sensors. In order to accomplish this task, we need to develop a fundamental mathematical tool, for which we study an iterative algorithm of variational data assimilation in this thesis.

1.2. A BRIEF INTRODUCTION FOR THE HISTORY OF DATA ASSIMILATION

The basic idea of data assimilation was first introduced in numerical weather prediction and has been developed rapidly ever since. In this section, we briefly introduce the history of the development of data assimilation, including the general ideas of different data assimilation techniques.

1.2.1. Subjective Analysis. Subjective analysis was first started in the 19th century. It was a labor-consuming process since the initial values for the grids were determined by subjectively drawing charts and interpolating between isolines. Although the process was subjective, it was a kind of data assimilation, where the local observations were combined with the experience to provide the map [5].

1.2.2. Richarson's Numerical Weather Prediction. The numerical weather predicion was first attemped by Lewis F. Richardson in 1922 [25] . He made it by hand in 1917 since it was before digital computers. His trial failed due to the fact that the observational data had not been assimilated properly which led to an unbalanced initial state [5]. But in [16], Lynch showed that with an appropriate smoothing of the initial condition, Richardson's prediction could have been accurate. Richarson's general philosophy of weather forecasting is still being used today.

1.2.3. Successive Correction Methods. This approach is in an iterative manner: The variable at each grid point is updated iteratively based on the first guess and the observation surrounding the grid point. Within a specified tolerance, the variable is updated by the following formula [22]:

$$f_i^{n+1} = f_i^n + \sum_{k=1}^{K_i^n} w_{ij}^n (f_k^0 - f_k^n) + \sum_{k=1}^{K_i^n} w_{ik}^n + \varepsilon^2$$

where f_i^n is the value of the variable at the i^{th} grid point at the n^{th} iteration,

f_k^0 is the k^{th} observation surrounding the grid point, f_k^n is the value of n^{th} field estimate calculated at observation point k derived by interpolation from nearest grid points, w_{ik} is a weighting function, and ε^2 is an estimate of the ratio of the observation error to the first guess field error.

There are two commonly used schemes for the successive correction methods: Cressman's scheme which is also referred to as Cressman's objective analysis [7] and the Barnes scheme [4]. The Cressman scheme defines the weighting function as:

$$\begin{aligned} w_{ik}^n &= \tfrac{R_n^2 - r_{ik}^2}{R_n^2 + r_{ik}^2}, \quad \text{if} \quad r_{ik}^2 < R_n^2 \\ w_{ik}^n &= 0, \quad \text{if} \quad r_{ik}^2 > R_n^2 \end{aligned}$$

where r_{ik} is the distance from the observation to the grid point and R_n is the radius of influence. According to the weighting formula, observations whose distance is larger than the radius of influence will not be used to update field variables. In the Cressman's scheme, the ε^2 is set to be 0 which means the observations are perfect. On the other hand, Barnes(1964) defined the weights to follow a Gaussian or normal distribution [4]

$$W_{ij} = \begin{cases} exp - \left(\frac{r_{ik}^2}{d^2}\right) & \text{if } r_{ik} \leq d \\ 0 \text{ otherwise }, \end{cases}$$

where d is the radius of influence. The Barnes scheme is most used when there is no available first guess field(such as when analyzing small scale phenomena).

1.2.4. Nudging. Nudging is also known as Newtonian relaxation and dynamic initialization. The standard nudging algorithm adds a feedback term to the state equations of a dynamical system. If we have a model written as follow

$$\frac{dx}{dt} = M(x),$$

then the nudging equation is

$$\frac{dx}{dt} = M(x) + \alpha(y - x)$$

where y is a direct observation of x [5]. This method is very easy to be implemented but has several drawbacks: the relaxation coefficient α must be determined; the

method is not applicable with undirect observations; this method of nudging is used for some specific applications, mostly when observational data is not real observation but data from an analysis.

1.2.5. Optimal Interpolation. The optimal interpolation aims at minimizing the total error of all the observations to form an optimized weighting for the observations. Based on [22], we write the analysis equation in the following form:

$$X_a = X_b + K(y - H[X_b]),$$

where K is a linear operator referred to as gain or weight matrix of the analysis and is given by

$$K = BH^T(HBH^T + R)^{-1},$$

where X_a is the analysis model state, H is an observation operator, B is the covariance matrix of the background errors $(X_b - X)$, X is the time model state, X_b is the background model state, and R is the covariance matrix of observation errors. The analysis error covariance matrix is

$$A = (I - KH)B(I - KH)^T + KRK^{-1}$$

It is showed that the best linear unbiased estimator may be obtained as the solution of the following variational optimization problem [6, 27]:

$$min\ J \ = \ (X - X_b)^T B^{-1}(X - X_b) + (y - H(X))^T R^{-1}(y - H(X)).$$

The advantage of Optimal Interpolation is that it is simple to implement and costs small if appropriate assumptions are made on observations. The disadvantage is that noise is produced during the process since different observations are used on different parts of the model state.

1.2.6. The Kalman Filter and Ensemble Kalman Filter. The Kalman Filter [14], also known as linear quadratic estimation, is an algorithm that uses a series of measurements observed over time, and produces estimates of unknown varibales.

It was first introduced by R.E. Kalman in 1960. Since then, It has been developed rapidly with numerous applications in different fields such as signal processing and econometrics.

The Kalman Filter is a group of equations that work in a recursive way to estimate the state of a process in the way of minimizing the mean of the squared error. The algorithm works in a two-step process by using a form of feedback control. In the prediction step, the Kalman filter produces estimates of the current state variables, along with their uncertainties. Once the outcome of the next (noisy) measurement is observed, these estimates are updated using a weighted average, with more weight being given to estimates with higher certainty. Because of the algorithm's recursive nature, it can run in real time using only the present input measurements and the previously calculated state and its uncertainty matrix; no additional past information is required.

The Ensemble Kalman filter [10] originated as a version of the Kalman filter for large problems and is an important data assimilation component of ensemble forecasting. It is a Monte Carlo implementation of the Bayesian update problem. There are numerous applications involving the emsemble Kalman filter include the initial work done by Evenson. Some examples are Lorenz equations, ocean model and two-layer shallow water model which are included in [11].

1.2.7. Variational Data Assimilation. Based on [22], the goal of variational data assimilation is to find the solution of numerical forecast model which best fits the observation distributed in space over a finite time interval. Assuming that our numerical forecast model is given as

$$B\frac{dX}{dt} + A(X) = 0$$

with B being identity for a dynamic model or null operator for steady state model. A can be linear or nonlinear operator. We define U as control variables which may consist of initial conditions, boundary conditions or model parameters. The major step consists in formulating the cost function J which measures the distance between model trajectory and observation as well as the background field at initial time during a finite time-interval. The minimization of the cost function can be viewed both in the perspective of finding its gradient in (a) Lagrangian approach, (b) adjoint operator approach and (c) a general synthesis of optimality conditions in the framework of optimal control theory approach.

2. REVIEW OF A VARIATIONAL DATA ASSIMILATION METHOD

Variational data assimilation aims to provide an optimal estimate of the initial state of a dynamic system by minimizing the cost functional that measures the difference between the observation and the modeled state over time [8, 26]. In this section, we review a variational data assimilation method [18, 23], including the target problem with optimal control via initial conditions and an iterative algorithm for approximating the optimal control.

2.1. TARGET PROBLEM WITH OPTIMAL CONTROL

We consider an evolution problem of the following form [9]:

$$\begin{cases} \frac{\partial \varphi}{\partial t} = \mathcal{F}(\varphi), \quad t \in (0, T) \\ \varphi \mid_{t=0} = u \end{cases} \tag{1}$$

where $\varphi = \varphi(t)$ is the unknown function belonging for any t to a Hilbert space H, $u \in H$, $\mathcal{F}$ is an nonlinear operator mapping H into H. Let $Y = L_2(0, T; H)$, $\|\cdot\|_Y = (\cdot, \cdot)_Y^{1/2}$. Let us introduce the functional:

$$J(u) = \frac{\alpha}{2}\|u\|_H^2 + \frac{1}{2}\int_0^T \|C\varphi - \varphi_{obs}\|_H^2 dt$$

where $\alpha = const \geq 0$, $\varphi_{obs} \in Y_{obs}$ is the observation, Y_{obs} is the subspace of Y, and $C : Y \to Y_{obs}$ is a linear operator. From the problem (1) we can formulate the data assimilation problem: find the optimal control u to minimize the cost functional $S(u)$ subject to equation (1). Therefore, the formulated problem can be written as:

$$\begin{cases} \frac{\partial \varphi}{\partial t} = \mathcal{F}(\varphi), t \in (0, T) \\ \varphi \mid_{t=0} = u \\ J(u) = \inf\limits_{v \in H} J(v) \end{cases} \tag{2}$$

Following [9] and references therein, the necessary optimality condition reduces

the problem (2) to the system:

$$\begin{cases} \frac{\partial \varphi}{\partial t} = \mathcal{F}(\varphi), t \in (0,T) \\ \varphi \mid_{t=0} = u \\ -\frac{\partial \varphi^*}{\partial t} - (\mathcal{F}'(\varphi))^* \varphi^* = -C^*(C\varphi - \varphi_{obs}), t \in (0,T) \\ \varphi^* \mid_{t=T} = 0 \\ \alpha u - \varphi^* \mid_{t=0} = 0 \end{cases} \tag{3}$$

where φ, φ^* are unknowns, $(\mathcal{F}'(\varphi))^*$is the adjoint to the Frechet derivative of $\mathcal{F}$, and C^* is the adjoint to C.

For simplification, in this thesis, we mainly consider the linear version of the above problem [18, 23]:

$$\begin{cases} \frac{\partial \varphi}{\partial t} + \mathcal{A}(t)\varphi = f, t \in (0,T) \\ \varphi \mid_{t=0} = u \end{cases} \tag{4}$$

where $\mathcal{A}(t)$ is a linear operator acting in Hilbert space H with domain of definition $D(A)$, $f \in L_2(0,T;H), u \in H$. With the same cost function defined above, we can formulate the corresponding minimization problem as follow:

$$\begin{cases} \frac{\partial \varphi}{\partial t} + \mathcal{A}(t)\varphi = f, t \in (0,T) \\ \varphi \mid_{t=0} = u \\ J(u) = \inf\limits_{v \in H} J(v) \end{cases} \tag{5}$$

According to [17, 18], the problem (5) is solvable and equivalent to the system for seeking the functions $\varphi = \varphi(t), \varphi^* = \varphi^*(t)$ and the control u in the form:

$$\begin{cases} \frac{\partial \varphi}{\partial t} + \mathcal{A}(t)\varphi = f, t \in (0,T) \\ \varphi \mid_{t=0} = u \\ -\frac{\partial \varphi^*}{\partial t} + \mathcal{A}^*(t)\varphi^* = \hat{\varphi} - \varphi, t \in (0,T) \\ \varphi^* \mid_{t=T} = 0 \\ \alpha u - \varphi^* \mid_{t=0} = 0 \end{cases} \tag{6}$$

where $\mathcal{A}^*(t)$ is the adjoint operator to $\mathcal{A}(t)$, $\hat{\varphi}$ is the observational data.

In order to approximate the above problem with respect to spatial variables, we can either apply finite difference methods or finite element methods for the spatial discretization based on a grid with grid size h. In general, after the semi-discretization, the system should have the following form:

$$\begin{aligned}
&\frac{d\varphi_h}{dt} + \mathcal{A}_h(t)\varphi = f_h, t \in (0,T)\\
&\varphi_h \mid_{t=0} = u\\
&-\frac{d\varphi_h^*}{dt} + \mathcal{A}_h^*(t)\varphi_h^* = \hat{\varphi}_h - \varphi_h\\
&\varphi_h^* \mid_{t=T} = 0\\
&\alpha u - \varphi_h^* \mid_{t=0} = 0
\end{aligned}$$

where Ω_h is a grid domain, $\mathcal{A}_h(t)$ is the grid operator that is obtained by approximating the linear operator $\mathcal{A}$ from (2.1), $\mathcal{A}_h^*(t)$ is the operator adjoint to $\mathcal{A}_h(t)$; $\varphi_h = \varphi_h(t)$,$\varphi_h^* = \varphi_h^*(t)$, $f_h = f_h(t)$, $\hat{\varphi}_h = \hat{\varphi}_h(t)$ are grid functions.

2.2. ITERATIVE ALGORITHM FOR APPROXIMATING THE OPTIMAL CONTROL

Assume we have an uniform partition for $[0,T]$ with temporal step size $\Delta t = \frac{T}{M}$. Then recall the iterative algorithm with iteration index j for the above system [18, 23]

$$\begin{aligned}
&\frac{\varphi_h^{k+1(j)} - \varphi_h^{k(j)}}{\Delta t} + \mathcal{A}_h^{k+1/2}\frac{\varphi_h^{k+1(j)} + \varphi_h^{k(j)}}{2} = f_h^{k+1/2}, k = 0, ..., M-1\\
&\varphi_h^{0(j)} = u^j\\
&-\frac{\varphi_h^{*k+1(j)} - \varphi_h^{*k(j)}}{\Delta t} + \mathcal{A}_h^{*k+1/2}\frac{\varphi_h^{*k+1(j)} + \varphi_h^{*k(j)}}{2} = \hat{\varphi}_h^{k+1/2} - \frac{\varphi_h^{k+1(j)} + \varphi_h^{k(j)}}{2}\\
&\varphi_h^{*M(j)} = 0, k = 0, ..., M\\
&u^{j+1} = u^j + \alpha_{j+1}B_j(\varphi^{*0(j)} - \alpha u^j) + \beta_{j+1}C_j(u^j - u^{j-1})
\end{aligned}$$

where $\alpha_{j+1}, \beta_{j+1}$ are iterative parameters, $\varphi^{k(j)}, \varphi^{*k(j)}, u^j$ are iterative sequences;B_j and C_j are symmetric positive definite matrices.

For the computation of $\alpha_{j+1}, \beta_{j+1}$, we follow [23] to apply the conjugate gradient method:

$$\alpha_{j+1} = 1/q_{j+1}, \beta_{j+1} = e_j/q_{j+1}$$
$$e_j = \begin{cases} q_j = 0, j = 0 \\ q_j \|\xi^j\|^2 / \|\xi^{j-1}\|^2, j > 0 \end{cases}$$
$$q_{j+1} = \|\xi^j\|_L^2 / \|\xi^{j-1}\|^2, j = 0, 1...$$

Here $\xi^j = \alpha u^j - \varphi^{*0(j)}$, $\|\xi^j\|_L = (L\xi^j, \xi^j)^{1/2}$. $L\xi^j$ is obtained from the successive solution of the following problems:

$$\frac{\phi_h^{k+1(j)} - \phi_h^{k(j)}}{\Delta t} + \mathcal{A}_h^{k+1/2} \frac{\phi_h^{k+1(j)} + \phi_h^{k(j)}}{2} = 0$$
$$\phi_h^0 = \xi^j$$
$$-\frac{\phi_h^{*k+1(j)} - \phi_h^{*k(j)}}{\Delta t} + \mathcal{A}_h^{*k+1/2} \frac{\phi_h^{*k+1(j)} + \phi_h^{*k(j)}}{2} = -\frac{\phi_h^{k+1(j)} + \phi_h^{k(j)}}{2}$$
$$\phi_h^{*M} = 0, k = 0, ..., M-1$$
$$L\xi^j = \alpha\xi^j - \phi_h^{*0}$$

In our study, we choose $B_j = C_j = E$ where E is the identity matrix.

3. APPLICATION FOR SECOND ORDER PARABOLIC EQUATION

In this section, we recall the standard derivation of the adjoint system (6) for a second order parabolic equation. Then the operator formulation of the iterative algorithm reviewed in Section 2 is discretized into the corresponding matrix formulation for practical computation. Two groups of numerical experiments are conducted to validate the iterative algorithm and approximate the optimal control.

Consider the following second order parabolic equation, which is also the second order formulation of Darcy's law:

$$\frac{\partial \varphi}{\partial t} - \nabla \cdot (K \nabla \varphi) = f, \quad \text{on } \Omega \times [0, T] \tag{7}$$

$$\varphi \mid_{t=0} = u, \quad \text{on } \Omega \tag{8}$$

$$\varphi = 0, \quad \text{on } \partial\Omega \times [0, T] \tag{9}$$

with the observation $\hat{\varphi}$ for the data assimilation problem introduced in Section 2. Here Ω is a 2D bounded domain.

3.1. DERIVATION OF THE ADJOINT SYSTEM FOR SECOND ORDER PARABOLIC EQUATION

In this section, we will recall the derivation of the adjoint equation (6). First, we use $\varphi(u)$ to denote the solution with the initial function u. Then

$$J(u) = \frac{\alpha}{2} \int_{\Omega} u^2 dxdy + \frac{1}{2} \int_0^T \int_{\Omega} \left[\hat{\varphi} - \varphi(u)\right]^2 dxdydt.$$

With the cost function defined above, we can formulate the minimization problem as follow:

$$\begin{cases} \frac{\partial \varphi}{\partial t} - \nabla \cdot (K \nabla \varphi) = f, & \text{on } \Omega \times [0, T] \\ \varphi \mid_{t=0} = u, & \text{on } \Omega \\ \varphi = 0, & \text{on } \partial\Omega \times [0, T] \\ J(u) = \inf\limits_{v \in H} J(v), & \text{on } \Omega \end{cases}$$

If $\tilde{u}$ is the minimizer of $J(u)$, then

$$\lim_{h \to 0} \frac{J(\tilde{u} + hv) - J(\tilde{u})}{h} = 0$$

for any v in the admissible set of controls. Hence

$$\begin{aligned} 0 &= \lim_{h\to 0} \frac{\frac{\alpha}{2}\int_\Omega [(\tilde{u}+hv)^2-\tilde{u}^2]\,dxdy}{h} \\ &\quad + \frac{\frac{1}{2}\int_0^T \int_\Omega \{[\hat{\varphi}-\varphi(\tilde{u}+hv)]^2-[\hat{\varphi}-\varphi(\tilde{u})]^2\}dxdydt}{h} \end{aligned}$$

Note that

$$(\tilde{u}+hv)^2-\tilde{u}^2 = 2\tilde{u}hv+h^2v^2$$

and

$$\begin{aligned} & [\hat{\varphi}-\varphi(\tilde{u}+hv)]^2-[\hat{\varphi}-\varphi(\tilde{u})]^2 \\ = \;& [\hat{\varphi}-\varphi(\tilde{u})+\varphi(\tilde{u})-\varphi(\tilde{u}+hv)]^2-[\hat{\varphi}-\varphi(\tilde{u})]^2 \\ = \;& [\hat{\varphi}-\varphi(\tilde{u})]^2+2\,[\hat{\varphi}-\varphi(\tilde{u})]\,[\varphi(\tilde{u})-\varphi(\tilde{u}+hv)] \\ & +[\varphi(\tilde{u}-\varphi(\tilde{u}+hv)]^2-[\hat{\varphi}-\varphi(\tilde{u})]^2 \\ = \;& 2\,[\hat{\varphi}-\varphi(\tilde{u})]\,[\varphi(\tilde{u})-\varphi(\tilde{u}+hv)]+[\varphi(\tilde{u})-\varphi(\tilde{u}+hv)]^2\,. \end{aligned}$$

Using the above three equations, we obtain

$$\begin{aligned} 0 &= \lim_{h\to 0}\int_\Omega \alpha\tilde{u}v dxdy+\frac{\alpha h}{2}\int_\Omega v^2 dxdy \\ &\quad -\int_0^T\int_\Omega [\hat{\varphi}-\varphi(\tilde{u})]\,\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}dxdydt \\ &\quad +h\int_0^T\int_\Omega \left[\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}\right]^2 dxdydt. \end{aligned} \tag{10}$$

Recall that $\varphi(u)$ is the solution with initial function u. Then

$$\begin{cases} \frac{\partial\varphi(\tilde{u})}{\partial t}-\nabla\cdot(K\nabla\varphi(\tilde{u}))=f, & \text{on } \Omega\times[0,T] \\ \varphi(\tilde{u})\mid_{t=0}=\tilde{u}, & \text{on } \Omega \\ \varphi(\tilde{u})=0, & \text{on } \partial\Omega\times[0,T] \end{cases}$$

and

$$\begin{cases} \frac{\partial\varphi(\tilde{u}+hv)}{\partial t}-\nabla\cdot(K\nabla\varphi(\tilde{u}+hv))=f, & \text{on } \Omega\times[0,T] \\ \varphi(\tilde{u}+hv)=0, & \text{on } \partial\Omega\times[0,T] \\ \varphi(\tilde{u}+hv)\mid_{t=0}=\tilde{u}+hv, & \text{on } \Omega \end{cases}$$

Hence

$$\begin{cases} \frac{\partial}{\partial t}\left(\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}\right) - \nabla\cdot\left(K\nabla(\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u}}{h}\right) = 0, & \text{on } \Omega\times[0,T] \\ \frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h} = 0, & \text{on } \partial\Omega\times[0,T] \\ \frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h} \mid_{t=0} = v, & \text{on } \Omega \end{cases} \tag{11}$$

Define

$$\bar{\varphi}(v) \triangleq \lim_{h\to 0}\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}.$$

Then

$$\begin{cases} \frac{\partial\bar{\varphi}(v)}{\partial t} - \nabla\cdot(K\nabla\bar{\varphi}(v)) = 0, & \text{on } \Omega\times[0,T] \\ \bar{\varphi}(v) = 0, & \text{on } \partial\Omega\times[0,T] \\ \bar{\varphi}(v) \mid_{t=0} = v, & \text{on } \Omega \end{cases} \tag{12}$$

Since $\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}$ is the solution of (11), we can obtain that $\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}$ is bounded and the upper bound is independent of h. Hence

$$\lim_{h\to 0} h\int_0^T\int_\Omega\left[\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h}\right]^2 dxdydt = 0.$$

Since $\lim_{h\to 0}\frac{\varphi(\tilde{u}+hv)-\varphi(\tilde{u})}{h} \triangleq \bar{\varphi}(v)$ and $\lim_{h\to 0}\frac{\alpha h}{2}\int_\Omega v^2 dxdy = 0$, then we can simplify (10) to be

$$\int_\Omega \alpha\tilde{u}v dxdy - \int_0^T\int_\Omega\left[\hat{\varphi}-\varphi(\tilde{u})\right]\bar{\varphi}(v)dxdydt = 0. \tag{13}$$

Multiplying the first equation in (12) by φ^* and taking the integral on $\Omega\times[0,T]$, we obtain

$$\int_0^T\int_\Omega\frac{\partial\bar{\varphi}(v)}{\partial t}\varphi^* dxdydt - \int_0^T\int_\Omega\nabla\cdot(K\nabla\bar{\varphi}(v))\,\varphi^* dxdydt = 0. \tag{14}$$

Using integration by parts, we get

$$\begin{aligned}
&\int_0^T \int_\Omega \frac{\partial \bar{\varphi}(v)}{\partial t} \varphi^* dxdydt \\
= &\int_\Omega \left[\bar{\varphi}\varphi^*\right] |_{t=0}^T \, dxdy - \int_0^T \int_\Omega \bar{\varphi}(v) \frac{\partial \varphi^*}{\partial t} dxdydt \\
= &\int_\Omega \left[\bar{\varphi}(v)\varphi^*\right] |_{t=T} \, dxdy - \int_\Omega \left[\bar{\varphi}(v)\varphi^*\right] |_{t=0} \, dxdy \\
&- \int_0^T \int_\Omega \bar{\varphi}(v) \frac{\partial \varphi^*}{\partial t} dxdydt
\end{aligned} \tag{15}$$

and

$$\begin{aligned}
&\int_0^t \int_\Omega \nabla \cdot (K \nabla \bar{\varphi}(v)) \varphi^* dxdydt \\
= &\int_0^T \int_{\partial\Omega} K \frac{\partial \bar{\varphi}(v)}{\partial n} \varphi^* dsdt - \int_0^T \int_\Omega K \nabla \bar{\varphi}(v) \cdot \nabla \varphi^* dxdydt \\
= &\int_0^T \int_{\partial\Omega} K \frac{\partial \bar{\varphi}(v)}{\partial n} \varphi^* dsdt \\
&- \left[\int_0^T \int_{\partial\Omega} K \bar{\varphi}(v) \frac{\partial \varphi^*}{\partial n} dsdt - \int_0^T \int_\Omega \bar{\varphi}(v) \nabla \cdot (K \nabla \varphi^*) dxdydt \right] \\
= &\int_0^T \int_{\partial\Omega} K \left[\frac{\partial \bar{\varphi}(v)}{\partial n} \varphi^* - \bar{\varphi}(v) \frac{\partial \varphi^*}{\partial n} \right] dsdt \\
&+ \int_0^T \int_\Omega \bar{\varphi}(v) \nabla \cdot (K \nabla \varphi^*) dxdydt.
\end{aligned} \tag{16}$$

Plugging (15) and (16) into (14), we get

$$\begin{aligned}
&\int_\Omega \left[\bar{\varphi}(v)\varphi^*\right] |_{t=T} \, dxdy - \int_\Omega \left[\bar{\varphi}(v)\varphi^*\right] |_{t=0} \, dxdy \\
&+ \int_0^T \int_\Omega \bar{\varphi}(v) \left[-\frac{\partial \varphi^*}{\partial t} - \nabla \cdot (K \nabla \varphi^*) \right] dxdydt \\
&- \int_0^T \int_{\partial\Omega} K \left[\frac{\partial \bar{\varphi}(v)}{\partial n} \varphi^* - \bar{\varphi}(v) \frac{\partial \varphi^*}{\partial n} \right] dsdt = 0.
\end{aligned} \tag{17}$$

By comparing (17) with (13), we can see that the equation for φ^* should be defined as

$$\begin{cases}
-\frac{\partial \varphi^*}{\partial t} - \nabla \cdot (K \nabla \varphi^*) = \hat{\varphi} - \varphi(\tilde{u}), & \text{on } \Omega \times [0, T] \\
\varphi^* = 0, & \text{on } \partial\Omega \times [0, T] \\
\varphi^* \, |_{t=T} = 0, & \text{on } \Omega
\end{cases}$$

Note that $\bar{\varphi}(v) = 0$ on $\partial\Omega$ and $\bar{\varphi}(v) \, |_{t=0} = v$ on Ω. Then (17) can be simplified to

be

$$-\int_{\Omega} v\varphi^*(0)dxdy + \int_0^T \int_{\Omega} \bar{\varphi}(v)\left[\hat{\varphi} - \varphi(\tilde{u})\right] dxdydt = 0. \tag{18}$$

Adding (18) to (13) we obtain $\int_{\Omega}\left[\alpha\tilde{u} - \varphi^*(0)\right] vdxdy = 0$ for any v in the admissible set of controls. Hence $\alpha\tilde{u} - \varphi^*(0) = 0$.

Based on the above derivation, we can obtain the following system for seeking the optimal control u, function φ, and adjoint function φ^*:

$$\begin{cases} \frac{\partial\varphi}{\partial t} - \nabla\cdot(K\nabla\varphi) = f, & \text{on } \Omega\times[0,T] \\ \varphi\mid_{t=0} = u, & \text{on } \Omega \\ \varphi = 0, & \text{on } \partial\Omega\times[0,T] \\ -\frac{\partial\varphi^*}{\partial t} - \nabla\cdot(K\nabla\varphi^*) = \hat{\varphi} - \varphi, & \text{on } \Omega\times[0,T] \\ \varphi^*\mid_{t=T} = 0, & \text{on } \Omega \\ \varphi^* = 0, & \text{on } \partial\Omega\times[0,T] \\ \alpha u - \varphi^*\mid_{t=0} = 0, & \text{on } \Omega. \end{cases} \tag{19}$$

3.2. WEAK FORMULATION AND FINITE ELEMENT DISCRETIZATION

In this subsection, we shortly recall the weak formulation and the finite element formulation for the above system. First, we multiply a test function v on both sides of the original equation

$$\frac{\partial\varphi}{\partial t} - \nabla\cdot(K\nabla\varphi) = f, \text{ in } \Omega\times[0,T]$$

and take the integral on Ω to obtain

$$\int_{\Omega}\varphi_t v dxdy - \int_{\Omega}\nabla\cdot(K\nabla\varphi)vdxdy = \int_{\Omega} fvdxdy,$$

Based on Green's formula and the given boundary condition, we get

$$\int_{\Omega}\varphi_t v dxdy + \int_{\Omega} K\nabla\varphi\cdot\nabla v dxdy = \int_{\Omega} fvdxdy.$$

Define $H^1(0,T;H^1(\Omega)) = \{v(t,\cdot), \frac{\partial v}{\partial t}(t,\cdot)\in H^1(\Omega),\ \forall t\in[0,T]\}$. Then the weak

formulation is to find $\varphi \in H^1(0,T;H^1(\Omega))$ such that

$$\int_\Omega \varphi_t v dxdy + \int_\Omega K\nabla\varphi \cdot \nabla v dxdy = \int_\Omega f v dxdy$$

for any $v \in H_0^1(\Omega)$. The weak formulation of the adjoint problem can be obtained similarly.

Assume $U_h \subset H^1(\Omega)$ is a finite element space based on a grid with size h. Then the finite element formulation is to find $\varphi_h \in H^1(0,T;U_h)$ such that

$$\int_\Omega \varphi_{ht} v_h dxdy + \int_\Omega K\nabla\varphi_h \cdot \nabla v_h dxdy = \int_\Omega f v_h dxdy$$

for any $v_h \in U_h$. Assume $U_h = span\{\phi_j\}_{j=1}^{N_b}$ where $\{\phi_j\}_j^{N_b}$ are the global finite element basis functions and N_b is the number of the global basis functions. Since $\varphi_h \in H^1(0,T;U_h)$, then we can assume $\varphi_h = \sum_{j=1}^{N_b} \varphi_j(t)\phi_j$. Choose $v_h = \phi_i (i = 1, ..., N_b)$. Then we can get,

$$\begin{aligned}
&\int_\Omega \left(\sum_{j=1}^{N_b} \varphi_j\phi_j\right)_t \phi_i dxdy + \int_\Omega K\nabla\left(\sum_{j=1}^{N_b} \varphi_j\phi_j\right)\cdot\nabla\phi_i dxdy \\
= &\int_\Omega f\phi_i dxdy \\
&\Rightarrow \sum_{j=1}^{N_b} \varphi_j' \left[\int_\Omega \phi_j\phi_i dxdy\right] + \sum_{j=1}^{N_b} \varphi_j \left[\int_\Omega K\nabla\phi_j\cdot\nabla\phi_i dxdy\right] \\
= &\int_\Omega f\phi_i dxdy,\ i = 1, ..., N_b.
\end{aligned}$$

Define the stiffness matrix

$$A_h = \left[\int_\Omega K\nabla\phi_j\cdot\nabla\phi_i dxdy\right]_{i,j=1}^{N_b},$$

the mass matrix

$$M_h = \left[\int_\Omega \phi_j\phi_i dxdy\right]_{i,j=1}^{N_b},$$

the load vector

$$\overrightarrow{b} = \left[\int_\Omega f\phi_i dxdy\right]_{i=1}^{N_b},$$

and the unknown vector

$$\overrightarrow{X} = [\varphi]_{j=1}^{N_b}.$$

Then we obtain the system:

$$M_h \frac{d\overrightarrow{X}}{dt} + A_h \overrightarrow{X} = \overrightarrow{b} \tag{20}$$

Similarly, we can obtain the following matrix formulation of the adjoint system:

$$-M_h \frac{d\overrightarrow{X}^*}{dt} + A_h \overrightarrow{X}^* = \overrightarrow{\tilde{b}} \tag{21}$$

where

$$\overrightarrow{\tilde{b}} = \left[\int_\Omega (\hat{\varphi} - \varphi)\, \phi_i dxdy\right]_{i=1}^{N_b}, \quad \overrightarrow{X}^* = \left[\varphi_j^*\right]_{j=1}^{N_b}.$$

Note that the adjoint matrix $A_h^* = A_h$ because the matrix A is a symmetric real matrix.

3.3. ITERATIVE ALGORITHM

Assume we have an uniform partition for $[0, T]$ with temporal step size $\Delta t = \frac{T}{M}$. Based on the above matrix formulation arising from the finite element discretization, we can apply the iterative algorithm, which was recalled in Section 2.2 with the iteration index j, in the following matrix formulation:

$$\begin{aligned}
&M_h \frac{\overrightarrow{X}^{k+1(j)} - \overrightarrow{X}^{k(j)}}{\Delta t} + \theta A_h^{k+1} \overrightarrow{X}^{k+1(j)} + (1-\theta) A_h^k \overrightarrow{X}^{k(j)} \\
&= \theta\, \overrightarrow{b}^{k+1} + (1-\theta)\, \overrightarrow{b}^{k}, \quad k = 0, ..., M-1 \qquad (22) \\
&\overrightarrow{X}^{0(j)} = \overrightarrow{u}^j \\
&-M_h \frac{\overrightarrow{X}^{*k+1(j)} - \overrightarrow{X}^{*k(j)}}{\Delta t} + \theta A_h^k \overrightarrow{X}^{*k(j)} + (1-\theta) A_h^{k+1} \overrightarrow{X}^{*k+1(j)} \\
&= \theta\, \overrightarrow{\tilde{b}}^{\,k(j)} + (1-\theta)\, \overrightarrow{\tilde{b}}^{\,k+1(j)}, \quad k = 0, ..., M-1 \qquad (23) \\
&\overrightarrow{X}^{*M(j)} = 0, \\
&\overrightarrow{u}^{j+1} = \overrightarrow{u}^j + \alpha_{j+1}(\overrightarrow{X}^{*0(j)} - \alpha\, \overrightarrow{u}^j) + \beta_{j+1}(\overrightarrow{u}^j - \overrightarrow{u}^{j-1}) \qquad (24)
\end{aligned}$$

where $\alpha_{j+1}, \beta_{j+1}$ are iterative parameters and $\theta \in [0,1]$ is the parameter for different time discretization scheme.

In order to compute the iterative parameters $\alpha_{j+1}, \beta_{j+1}$ in our iterative algorithm, we can apply the conjugate gradient method recalled in the previous section

$$\begin{aligned}
&\alpha_{j+1} = 1/q_{j+1}, \beta_{j+1} = e_j/q_{j+1} \\
&e_j = \begin{cases} 0, j = 0 \\ q_j \| \overrightarrow{\xi}^j \|^2 / \| \overrightarrow{\xi}^{j-1} \|^2, j > 0 \end{cases} \\
&q_{j+1} = \| \overrightarrow{\xi}^j \|_L^2 / \| \overrightarrow{\xi}^{j-1} \|^2, j = 0, 1...
\end{aligned}$$

Here $\overrightarrow{\xi}^j = \alpha \overrightarrow{u}^j - \overrightarrow{X}^{*0(j)}, \| \overrightarrow{\xi}^j \|_L = \left(L \overrightarrow{\xi}^j, \overrightarrow{\xi}^j \right)^{1/2}$. $L \overrightarrow{\xi}^j$ is obtained as a successive solution of the problems:

$$\begin{aligned}
&M_h \frac{\overrightarrow{Y}^{k+1(j)} - \overrightarrow{Y}^{k(j)}}{\Delta t} + \theta A_h^{k+1} \overrightarrow{Y}^{k+1(j)} + (1-\theta) A_h^k \overrightarrow{Y}^{k(j)} = 0, k = 0, ..., M-1 \\
&\overrightarrow{Y}^0 = \overrightarrow{\xi}^j \\
&-M_h \frac{\overrightarrow{Y}^{*k+1(j)} - \overrightarrow{Y}^{*k(j)}}{\Delta t} + \theta A_h^k \overrightarrow{Y}^{*k(j)} + (1-\theta) A_h^{k+1} \overrightarrow{Y}^{*k+1(j)} = -\theta \overrightarrow{b}^{k(j)} \\
&-(1-\theta) \overrightarrow{b}^{k+1(j)} \\
&\overrightarrow{Y}^{*M} = 0, k = 0, ..., M-1 \\
&L \overrightarrow{\xi}^j = \alpha \overrightarrow{\xi}^j - \overrightarrow{Y}^{*0(j)}
\end{aligned}$$

where $\overrightarrow{b}^{k(j)}$ is the discretization of $\overrightarrow{b} = \left[\int_\Omega \varphi \phi_i dx dy \right]_{i=1}^{N_b}$ with the finite element solution of φ at the j^{th} step of the iteration and time step k.

3.4. NUMERICAL RESULTS FOR SECOND ORDER PARABOLIC EQUATION

In this subsection, we carry out two numerical experiments to validate the iterative algorithm for the second order parabolic equation arising from the single-phase Darcy's law. The first numerical experiment is designed with known analytic solutions in order to demonstrate the properties of the iterative algorithm, including parameter sensitivity, convergence, accuracy, and efficiency, by comparing the number of iteration steps and the errors between the numerical solutions and the analytic solutions.

The second numerical experiment is a more realistic one for approximating the optimal control of the variational data assimilation problem.

3.4.1. Numerical Experiment For Validating The Iterative Algorithm. In the first numerical experiment, we consider the following system with a given observation function $\hat{\varphi}$ for the iterative algorithm:

$$\begin{cases} \frac{\partial \varphi}{\partial t} - \Delta\varphi = f, & \text{in } \Omega \times [0,T] \\ \varphi \mid_{t=0} = u, & \text{on } \Omega \\ \varphi = 0, & \text{on } \partial\Omega \times [0,T] \\ -\frac{\partial \varphi^*}{\partial t} + \Delta\varphi^* = \hat{\varphi} - \varphi + \tilde{f}, & \text{on } \Omega \times [0,T] \\ \varphi^* \mid_{t=T} = 0, & \text{on } \Omega \\ \varphi^* = 0, & \text{on } \partial\Omega \times [0,T] \\ \alpha u - \varphi^* \mid_{t=0} = 0, & \text{on } \Omega. \end{cases}$$

Compared with the original system (19) for the iterative algorithm, we artificially add the function $\tilde{f}$ here. It does not affect the convergence property of the iterative algorithm but provides the convenience to set up the first numerical experiment for the convergence of the iterative solution to the analytic solution given below(not the optimal control). Then we can compute the errors between the numerical solution φ_h and the analytic solution φ in order to illustrate the properties of the iterative algorithm. The influence of this additional function $\tilde{f}$ on the iterative algorithm in the discretized matrix formulation of Section 3.3 is to add the discretization of the following term to the right-hand side of the equation (23):

$$\overrightarrow{\tilde{b}} = \left[\int_\Omega \tilde{f} \phi_i dxdy \right]_{i=1}^{N_b}, \tag{25}$$

and obtain

$$\begin{aligned} &-M_h \frac{\overrightarrow{X}^{*k+1(j)} - \overrightarrow{X}^{*k(j)}}{\Delta t} + \theta A_h^k \overrightarrow{X}^{*k(j)} + (1-\theta) A_h^{k+1} \overrightarrow{X}^{*k+1(j)} \\ &= \theta \left(\overrightarrow{\tilde{b}}^{\,k(j)} + \overrightarrow{\hat{b}}^{\,k} \right) + (1-\theta) \left(\overrightarrow{\tilde{b}}^{\,k+1(j)} + \overrightarrow{\hat{b}}^{\,k+1} \right), \quad k = 0, ..., M-1 \end{aligned} \tag{26}$$

We will use (26) to replace (23).

Set $\Omega = [0,1]^2$, $\alpha = 1$, $\hat{\varphi} = sin(\pi x) sin(\pi y) e^t$. The problem is set up with

analytic solutions

$$\begin{aligned} \varphi &= sin(\pi x)sin(\pi y)e^t, \\ \varphi^* &= sin(\pi x)sin(\pi y)e^t(1-t). \end{aligned}$$

Hence

$$\begin{aligned} f &= (2\pi^2+1)sin(\pi x)sin(\pi y)e^t, \\ \tilde{f} &= sin(\pi x)sin(\pi y)e^t(2\pi^2(1-t)+t+1). \end{aligned}$$

Choose linear finite element for the spatial discretization with step size h, and Crank-Nicolson scheme for the temporal discretization with step size Δt. The tolerance to stop the iteration is set to be 10^{-6}. Then we can obtain the following numerical results for the iterative algorithm.

The first step is to test the effects of the initial guess and the mesh size on the iterative algorithm. Tables 3.1-3.5 provide the numerical errors for the solution φ at the initial time, which is in fact the control u in different norms and the number k of iteration steps with respect to the initial vector $\overrightarrow{u}^0$ arising from $u^0(x,y) = x^2y^2, -1, 1, 10, 100$ evaluated at all the nodes and the mesh size $h = \Delta t = 1/4, 1/8, 1/16, 1/32$ respectively. Crank-Nicolson scheme has second order accuracy for the time discretization. Linear finite element has second order accuracy in L^∞/L^2 norms and first order accuracy in H^1 norm for the spatial discretization. Therefore, when we choose $h = \Delta t$, we expect the second order accuracy in L^∞/L^2 norms and first order accuracy in H^1 norm for our numerical solution, which can be clearly observed from Tables 3.1-3.5. Using linear regression, the results in Tables 3.1-3.5 satisfy

$$\begin{aligned} \|u_h - u\|_\infty &= 0.7404h^{2.0418}, \\ \|u_h - u\|_0 &= 1.0613h^{1.9451}, \\ \|u_h - u\|_1 &= 3.2379h^{0.9691}. \end{aligned}$$

Furthermore, the small numbers of iteration steps clearly indicate the high efficiency of the iterative algorithm.

The numbers of iteration steps also stay almost the same for different initial guesses and different step sizes $h(= \Delta t)$, which indicates that the iterative algorithm is not sensitive to the initial guess for the iteration.

Table 3.1. Numerical results with the initial guess $u^0(x, y) = x^2y^2$

h ,Δt	L^∞ error	L^2 error	H^1 error	k
1/4	4.37098e-02	7.35758e-02	8.43646e-01	5
1/8	1.05882e-02	1.93837e-02	4.32501e-01	6
1/16	2.57620e-03	5.10365e-03	2.20566e-01	6
1/32	6.25718e-04	1.34350e-03	1.12520e-01	6

Table 3.2. Numerical results with the initial guess $u^0(x, y) = -1$

h ,Δt	L^∞ error	L^2 error	H^1 error	k
1/4	4.37098e-02	7.35758e-02	8.43646e-01	7
1/8	1.05882e-02	1.93837e-02	4.32501e-01	7
1/16	2.57620e-03	5.10365e-03	2.20566e-01	7
1/32	6.25718e-04	1.34350e-03	1.12520e-01	7

Table 3.3. Numerical results with the initial guess $u^0(x, y) = 1$

h ,Δt	L^∞ error	L^2 error	H^1 error	k
1/4	4.37098e-02	7.35758e-02	8.43646e-01	7
1/8	1.05882e-02	1.93837e-02	4.32501e-01	7
1/16	2.57620e-03	5.10365e-03	2.20566e-01	7
1/32	6.25718e-04	1.34350e-03	1.12520e-01	7

Table 3.4. Numerical results with the initial guess $u^0(x, y) = 10$

h ,Δt	L^∞ error	L^2 error	H^1 error	k
1/4	4.37098e-02	7.35758e-02	8.43646e-01	8
1/8	1.05882e-02	1.93837e-02	4.32501e-01	8
1/16	2.57620e-03	5.10365e-03	2.20566e-01	8
1/32	6.25718e-04	1.34350e-03	1.12520e-01	8

Table 3.5. Numerical results with the initial guess $u^0(x, y) = 100$

h ,Δt	L^∞ error	L^2 error	H^1 error	k
1/4	4.37098e-02	7.35758e-02	8.43646e-01	8
1/8	1.05882e-02	1.93837e-02	4.32501e-01	8
1/16	2.57620e-03	5.10365e-03	2.20566e-01	8
1/32	6.25718e-04	1.34350e-03	1.12520e-01	8

The second step is to test the effect of the accuracy of the observational data function on the iterative algorithm. We add several different random perturbations $r\epsilon$ to the observational data function $\hat{\varphi} = sin(\pi x)sin(\pi y)e^t$ where r is a random number between $[0, 1]$ and then repeat the same numerical experiment with a fixed iteration step $k = 10$. For small perturbations we obtain the numerical results in Tables 3.6-3.8, which indicates that the iterative algorithm is optimally convergent as long as the observational data is accurate enough. It is also observed from Tables 3.9-3.10 that larger perturbations deteriorate the numerical solutions.

Table 3.6. Numerical results with $\epsilon = 10^{-6}$

h ,Δt	L^∞ error	L^2 error	H^1 error
1/4	4.83787e-02	7.54182e-02	8.43692e-01
1/8	1.42433e-02	2.10126e-02	4.32563e-01
1/16	4.20582e-03	5.80278e-03	2.19013e-01
1/32	1.24141e-03	1.61593e-03	1.12474e-01

Table 3.7. Numerical results with $\epsilon = 10^{-4}$

h ,Δt	L^∞ error	L^2 error	H^1 error
1/4	4.83787e-02	7.54182e-02	8.43692e-01
1/8	1.42433e-02	2.10126e-02	4.32563e-01
1/16	4.20582e-03	5.80278e-03	2.19013e-01
1/32	1.24141e-03	1.61593e-03	1.12474e-01

Table 3.8. Numerical results with $\epsilon = 10^{-2}$

h ,Δt	L^∞ error	L^2 error	H^1 error
1/4	4.80414e-02	7.52649e-02	8.43683e-01
1/8	1.38939e-02	2.08364e-02	4.32549e-01
1/16	4.00673e-03	5.83168e-03	2.21661e-01
1/32	1.16039e-03	1.61461e-03	1.13659e-01

Table 3.9. Numerical results with $\epsilon = 1$

h ,Δt	L^∞ error	L^2 error	H^1 error
1/4	4.97241e-02	7.63238e-02	8.59217-01
1/8	1.47615e-02	2.17281e-02	4.54129e-01
1/16	4.13253e-03	5.95713e-03	2.34172e-01
1/32	1.25147e-03	1.74562e-03	1.23434e-01

Table 3.10. Numerical results with $\epsilon = 10^2$

h ,Δt	L^∞ error	L^2 error	H^1 error
1/4	3.32543e+00	1.62783e+00	8.38266e+00
1/8	3.47997e+00	1.89088e+00	9.02452e+00
1/16	3.49123e+00	1.90345e+00	9.10526e+00
1/32	3.52193e+00	2.04170e+00	9.17321e+00

3.4.2. Numerical Experiment For Approximating The Optimal Control of A More Realistic Problem of The Second Order Parabolic Equation. In the second numerical experiment, we consider the following original system (19) with given observation function $\hat{\varphi}$ for seeking the optimal control u. The system in (19) does not include the function $\tilde{f}$ which was artificially added in the first numerical experiment. Set $\Omega = [0,1]^2$, $\hat{\varphi} = sin(\pi x)sin(\pi y)e^t + 10^{-2}r$, and $f = (2\pi^2 + 1)sin(\pi x)sin(\pi y)e^t$ where r is a random number between 0 and 1. The analytic solution $\varphi = sin(\pi x)sin(\pi y)e^t$. Here we take the initial vecotr $\overrightarrow{u}^0$ with all entries equal to 1 and the tolerance for stopping the iteration to be 10^{-6} with $h = \Delta t = 1/16$. Table 3.11 provides the numerical errors in the solution φ at the initial time, which is in fact the control u and the number k of iteration steps for seeking the optimal control u with different parameter.

Note that the cost functional was defined in Section 2 as

$$J(\varphi) = \frac{\alpha}{2}\|u\|^2 + \frac{1}{2}\int_0^T \|\hat{\varphi} - \varphi\|^2 dt$$

where the wight coefficient $\alpha > 0$, $\hat{\varphi}(t)$ is a given function generally defined by the priori observational data, and $\|\cdot\|$ is the norm in a Hilbert space H.

Since α is the weight coefficient of the cost of the control in the cost function, we expect that smaller α can improve the accuracy with an increased cost. This is verified by the decreased errors and increased number of iteration steps in Table 3.11.

Table 3.11. Numerical results for different α

α	L^∞ error	L^2 error	H^1 error	k
1	9.74315e-01	4.87463e-01	2.36452e+00	6
0.5	9.49825e-01	4.75508e-01	2.31017e+00	7
0.2	8.82750e-01	4.42763e-01	2.16156e+00	10
0.1	7.88512e-01	3.96747e-01	1.95357e+00	14
0.01	2.83542e-01	1.64310e-01	1.44247e+00	51
0.001	4.00371e-02	2.35214e-02	0.94573e-01	76

4. ITERATIVE ALGORITHM FOR STOKES EQUATION

In this section, we apply the iterative algorithm based on the discretized matrix formulation in Section 3 to Stokes equation, which is an important preparation for the Stokes-Darcy model in order to study the CO_2 storage in fractured reservoir and the leakage around the nature faults in the future work. We conduct two groups of numerical experiments to validate the iterative algorithm and approximate the optimal control.

4.1. TARGET PROBLEM AND ITS WEAK FORMULATION

We consider the Stokes equation

$$\begin{aligned}
\overrightarrow{\varphi}_t - \nabla \cdot T(\overrightarrow{\varphi}, p) &= \overrightarrow{f} && \text{on } \Omega \times [0,T] && (27)\\
\nabla \cdot \overrightarrow{\varphi} &= 0 && \text{on } \Omega \times [0,T] && (28)\\
\overrightarrow{\varphi} \mid_{t=0} &= \overrightarrow{w} && \text{on } \Omega\\
p \mid_{t=0} &= p_0 && \text{on } \Omega\\
\overrightarrow{\varphi} &= 0, && \text{on } \partial\Omega \times [0,T]
\end{aligned}$$

with the observation $\overrightarrow{\hat{\varphi}}$ for the data assimilation problem introduced in Section 2. Here $\overrightarrow{\varphi}$ and p are the velocity and pressure of the fluid flow respectively, stress tensor $T(\overrightarrow{\varphi}, p) = -pI + 2\nu D(\overrightarrow{\varphi})$, deformation tensor $D(\overrightarrow{\varphi}) = \frac{1}{2}(\nabla\overrightarrow{\varphi} + \nabla\overrightarrow{\varphi}')$, I is the identity matrix, and Ω is a 2D domain. With the cost function defined as

$$J(\overrightarrow{w}) = \frac{\alpha}{2}\|\overrightarrow{w}\|^2 + \frac{1}{2}\int_0^T \|\overrightarrow{\hat{\varphi}} - \overrightarrow{\varphi}\|^2 dt,$$

we can formulate the minimization problem as follow:

$$\begin{cases}
\overrightarrow{\varphi}_t - \nabla \cdot T(\overrightarrow{\varphi}, p) = \overrightarrow{f} & \text{on } \Omega \times [0,T]\\
\nabla \cdot \overrightarrow{\varphi} = 0 & \text{on } \Omega \times [0,T]\\
\varphi \mid_{t=0} = \overrightarrow{w} & \text{on } \Omega\\
p \mid_{t=0} = p_0 & \text{on } \Omega\\
\overrightarrow{\varphi} = 0, & \text{on } \partial\Omega \times [0,T]\\
J(\overrightarrow{w}) = \inf\limits_{\overrightarrow{v} \in H} J(\overrightarrow{v}) & \text{on } \Omega
\end{cases}$$

In order to derive the weak formulation, we first test function (27) with a vector function $\overrightarrow{v}$ and take the integral in Ω on both sides of the equation,

$$\int_\Omega \overrightarrow{\varphi}_t \cdot \overrightarrow{v} dxdy - \int_\Omega [\nabla \cdot T(\overrightarrow{\varphi}, p)] \cdot \overrightarrow{v} dxdy = \int_\Omega \overrightarrow{f} \cdot \overrightarrow{v} dxdy.$$

Applying the divergence theory with the given boundary condition,we can get

$$\int_\Omega \overrightarrow{\varphi}_t \cdot \overrightarrow{v} dxdy + \int_\Omega 2\nu D\overrightarrow{\varphi} : D\overrightarrow{v} dxdy - \int_\Omega p\nabla \cdot \overrightarrow{v} dxdy = \int_\Omega \overrightarrow{f} \cdot \overrightarrow{v} dxdy,$$

where $D\overrightarrow{\varphi} : D\overrightarrow{v} = \varphi_{1x}v_{1x} + \varphi_{2y}v_{2y} + \frac{1}{2}\varphi_{1y}v_{1y} + \frac{1}{2}\varphi_{1y}v_{2x} + \frac{1}{2}\varphi_{2x}v_{1y} + \frac{1}{2}\varphi_{2x}v_{2x}$. Then we test equation (28) by multiplying a function q and take the integral in Ω on both sides of the equation to get

$$-\int_\Omega q\nabla \cdot \overrightarrow{\varphi} dxdy = 0.$$

Define

$$H^1(0,T;\left[H^1(\Omega)\right]^2) = \left\{ \overrightarrow{v} : \overrightarrow{v}(t,\cdot), \frac{\partial \overrightarrow{v}}{\partial t}(t,\cdot) \in \left[H^1(\Omega)\right]^2,\ \forall t \in [0,T] \right\},$$
$$L^2(0,T;L^2(\Omega)) = \left\{ \phi : \phi(t,\cdot) \in L^2(\Omega), \forall t \in [0,T] \right\},$$

the weak formulation is to find $\overrightarrow{\varphi} \in H^1(0,T;[H^1(\Omega)]^2)$ and $p \in L^2(0,T;L^2(\Omega))$ such that

$$\int_\Omega \overrightarrow{\varphi}_t \cdot \overrightarrow{v}_h dxdy + \int_\Omega 2\nu D\overrightarrow{\varphi} : D\overrightarrow{v}_h dxdy - \int_\Omega p\nabla \cdot \overrightarrow{v}_h dxdy = \int_\Omega \overrightarrow{f} \cdot \overrightarrow{v}_h dxdy,$$
$$\forall v \in \left[H^1(\Omega)\right]^2,$$
$$-\int_\Omega q\nabla \cdot \overrightarrow{\varphi} dxdy = 0, \quad \forall q \in L^2(\Omega).$$

4.2. FINITE ELEMENT DISCRETIZATION AND ITERATIVE ALGORITHM

Assume $X_h \subset [H^1(\Omega)]^2$ and $Q_h \subset L^2(\Omega)$ are two finite element spaces based on a grid with grid size h. We assume that X_h and Q_h consist of the first order or higher order of piecewise polynomials and satisfy the inf-sup condition [12,13]:

$$\inf_{0 \neq q \in Q_h} \sup_{0 \neq \overrightarrow{v} \in X_h} \frac{b(\overrightarrow{v}, q)}{\|\overrightarrow{v}\|_1 \|q\|_0} > \beta, \tag{29}$$

where $\beta > 0$ is a constant independent of the mesh size h and $b(\overrightarrow{v}, q) = \int_\Omega q\nabla \cdot \overrightarrow{v} dxdy$. This condition is needed to ensure that the spatial discretizations of the Stokes system are stable. Then the finite element formulation is to find $\overrightarrow{\varphi}_h \in H^1(0,T;X_h), p_h \in L^2(0,T;Q_h)$ such that

$$\begin{aligned}
&\int_\Omega \frac{d\overrightarrow{\varphi}_h}{dt} \cdot \overrightarrow{v}_h dxdy + \int_\Omega 2\nu D\overrightarrow{\varphi}_h : D\overrightarrow{v}_h dxdy - \int_\Omega p_h \nabla \cdot \overrightarrow{v}_h dxdy \\
&= \int_\Omega \overrightarrow{f} \cdot \overrightarrow{v}_h dxdy, \quad \forall \overrightarrow{v}_h \in X_h \qquad (30) \\
&\int_\Omega q_h \nabla \cdot \overrightarrow{\varphi}_h dxdy = 0, \quad \forall q_h \in Q_h. \qquad (31)
\end{aligned}$$

Assume $X_h = span\{\phi_i\}_{i=1}^{N_1}$, $Q_h = span\{\psi_i\}_{i=1}^{N_2}$ where $\{\phi_i\}_{i=1}^{N_1}$ and $\{\psi_i\}_{i=1}^{N_2}$ are the global finite element basis functions. With

$$\overrightarrow{\varphi_h} = \begin{pmatrix} \varphi_{1h} \\ \varphi_{2h} \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{N_1} \varphi_{1j}\phi_j \\ \sum_{j=1}^{N_1} \varphi_{2j}\phi_j \end{pmatrix}, \quad p_h = \sum_{j=1}^{N_2} p_j \psi_i, \qquad (32)$$

we test equation (30) and (31) by the following three steps. First, we use

$$\overrightarrow{v_h} = \begin{pmatrix} \phi_i \\ 0 \end{pmatrix}, i = 1, ...N_1$$

to test (30) and get

$$\begin{aligned}
&\int_\Omega \frac{d\varphi_{1h}}{dt}\phi_i dxdy + \int_\Omega \nu \left(2\frac{\partial \varphi_{1h}}{\partial x}\frac{\partial \phi_i}{\partial x} + \frac{\partial \varphi_{1h}}{\partial y}\frac{\partial \phi_i}{\partial y} + \frac{\partial \varphi_{2h}}{\partial x}\frac{\partial \phi_i}{\partial y} \right) dxdy \\
&- \int_\Omega p_h \frac{\partial \phi_i}{\partial x} dxdy = \int_\Omega f_1 \phi_i dxdy. \qquad (33)
\end{aligned}$$

By plugging (32) into equation (33), we get

$$\begin{aligned}
&\sum_{j=1}^{N_1} \frac{\varphi_{1j}}{dt} \left[\int_\Omega \phi_j \phi_i dxdy \right] + \sum_{j=1}^{N_1} \varphi_{1j} \left[\int_\Omega \nu \left(2\frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} \right) dxdy \right] \\
&+ \sum_{j=1}^{N_1} \varphi_{2j} \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y} dxdy \right] + \sum_{j=1}^{N_2} p_i \left[- \int_\Omega \psi_j \frac{\partial \phi_i}{\partial x} dxdy \right] = \int_\Omega f_1 \phi_i dxdy, \\
&i = 1, ..., N_1. \qquad (34)
\end{aligned}$$

Second, we use

$$\overrightarrow{v}_h = \begin{pmatrix} 0 \\ \phi_i \end{pmatrix}, i = 1, ..., N_1$$

to test (30) and get

$$\begin{aligned} &\int_\Omega \frac{d\varphi_{2h}}{dt}\phi_i dxdy + \int_\Omega \nu \left(2\frac{\partial \varphi_{2h}}{\partial y}\frac{\partial \phi_i}{\partial y} + \frac{\partial \varphi_{1h}}{\partial y}\frac{\partial \phi_i}{\partial x} + \frac{\partial \varphi_{2h}}{\partial x}\frac{\partial \phi_i}{\partial x}\right) dxdy \\ &- \int_\Omega p_h \frac{\partial \phi_i}{\partial y} dxdyy = \int_\Omega f_2 \phi_i dxdy. \end{aligned} \tag{35}$$

By plugging (32) into equation (35), we get

$$\begin{aligned} &\sum_{j=1}^{N_1} \frac{d\varphi_{2j}}{dt}\left[\int_\Omega \phi_j\phi_i dxdy\right] + \sum_{j=1}^{N_1} \varphi_{1j}\left[\int_\Omega \nu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x} dxdy\right] \\ + &\sum_{j=1}^{N_1} \varphi_{2j}\left[\int_\Omega \nu \left(2\frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x}\right) dxdy\right] + \sum_{j=1}^{N_2} p_j\left[-\int_\Omega \psi_j \frac{\partial \phi_i}{\partial y} dxdy\right] \\ = &\int_\Omega f_2\phi_i dxdy. \end{aligned} \tag{36}$$

Third, we use $q_h = \psi_i$ to test (31) and get

$$\begin{aligned} 0 &= -\int_\Omega \psi_i \nabla \cdot \overrightarrow{\varphi}_h dxdy \\ &= \sum_{j=1}^{N_1} \varphi_{1j}\left[-\int_\Omega \frac{\partial \phi_j}{\partial x}\psi_i dxdy\right] + \sum_{j=1}^{N_2} \varphi_{2j}\left[-\int_\Omega \frac{\partial \phi_j}{\partial y}\psi_i dxdy\right]. \end{aligned} \tag{37}$$

Based on (34), (36), (37), the linear system can be written as

$$\sum_{j=1}^{N_1}\frac{d\varphi_{1j}}{dt}\left[\int_\Omega \phi_j\phi_i dxdy\right]+\sum_{i=1}^{N_1}\varphi_{1j}\left[\int_\Omega \nu\left(2\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\right)dxdy\right]$$
$$+\sum_{i=1}^{N_1}\varphi_{2j}\left[\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}dxdy\right]+\sum_{j=1}^{N_2}p_j\left[-\int_\Omega \psi_j\frac{\partial\phi_i}{\partial x}dxdy\right]$$
$$=\int_\Omega f_1\phi_i dxdy, i=1,...,N_1,$$
$$\sum_{j=1}^{N_1}\frac{d\varphi_{2j}}{dt}\left[\int_\Omega \phi_j\phi_i dxdy\right]+\sum_{j=1}^{N_1}\varphi_{1j}\left[\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}dxdy\right]$$
$$+\sum_{j=1}^{N_1}\varphi_{2j}\left[\int_\Omega \nu\left(2\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}\right)dxdy\right]+\sum_{j=1}^{N_2}p_j\left[-\int_\Omega \psi_j\frac{\partial\phi_i}{\partial y}dxdy\right]$$
$$=\int_\Omega f_2\phi_i dxdy, i=1,...,N_1$$
$$\sum_{j=1}^{N_1}\varphi_{1j}\left[-\int_\Omega \frac{\partial\phi_j}{\partial x}\psi_i dxdy\right]+\sum_{j=1}^{N_2}\varphi_{2j}\left[-\int_\Omega \frac{\partial\phi_j}{\partial y}\psi_i dxdy\right]=0, i=1,...,N2.$$

Therefore, we can define:

$$M=\left[\int_\Omega \phi_j\phi_i dxdy\right]_{i,j=1}^{N1},\quad A_1=\left[\int_\Omega \nu\left(2\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\right)dxdy\right]_{i,j=1}^{N_1},$$
$$A_2=\left[\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}dxdy\right]_{i,j=1}^{N_1},\quad A_3=\left[\int_\Omega \nu\left(2\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}\right)dxdy\right]_{i,j=1}^{N_1},$$
$$B_1=\left[-\int_\Omega \psi_j\frac{\partial\phi_i}{\partial x}dxdy\right]_{i=1,j=1}^{N_1,N_2},\quad B_2=\left[-\int_\Omega \psi_j\frac{\partial\phi_i}{\partial x}dxdy\right]_{i=1,j=1}^{N_1,N_2},$$
$$\overrightarrow{W_1}=[\varphi_{1j}]_{j=1}^{N_1},\quad \overrightarrow{W_2}=[\varphi_{2j}]_{j=1}^{N_1},\quad \overrightarrow{W_3}=[p_j]_{j=1}^{N_2},$$
$$\overrightarrow{F_1}=\left[\int_\Omega f_1\phi_i dxdy\right]_{i=1}^{N_1},\quad \overrightarrow{F_2}=\left[\int_\Omega f_2\phi_i dxdy\right]_{i=1}^{N_2},\quad \overrightarrow{F_3}=[0]_{N_2\times 1},$$

where $\overrightarrow{W_i}$ and $\overrightarrow{F_i}$ are column vectors. Then the discretized linear system can be rewritten as:

$$M\frac{d\overrightarrow{W_1}}{dt}+A_1\overrightarrow{W_1}+A_2\overrightarrow{W_2}+B_1\overrightarrow{W_3}=\overrightarrow{F_1}$$
$$M\frac{d\overrightarrow{W_2}}{dt}+A_2^T\overrightarrow{W_1}+A_3\overrightarrow{W_2}+B_2\overrightarrow{W_3}=\overrightarrow{F_2}$$
$$B_1^T\overrightarrow{W_1}+B_2^T\overrightarrow{W_2}=\overrightarrow{F_3}$$

Define:

$$\mathbf{M_s} = \begin{pmatrix} M & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & 0 \end{pmatrix}, \mathbf{A_s} = \begin{pmatrix} A_1 & A_2 & B_1 \\ A_2^T & A_3 & B_2 \\ B_1^T & B_2^T & 0 \end{pmatrix}, \overrightarrow{W_s} = \begin{pmatrix} \overrightarrow{W_1} \\ \overrightarrow{W_2} \\ \overrightarrow{W_3} \end{pmatrix},$$

$$\overrightarrow{W_v} = \begin{pmatrix} \overrightarrow{W_1} \\ \overrightarrow{W_2} \end{pmatrix}, \overrightarrow{F_s} = \begin{pmatrix} \overrightarrow{F_1} \\ \overrightarrow{F_2} \\ \overrightarrow{F_3} \end{pmatrix}, \overrightarrow{p} = \overrightarrow{W_3}$$

where M_s is the mass matrix, A_s is the stiffness matrix, $\overrightarrow{W_s}$ is the unknown vector, and $\overrightarrow{F_s}$ is the load vector. Then we can rewrite the system as:

$$M_s \frac{d\overrightarrow{W_s}}{dt} + A_s \overrightarrow{W_s} = \overrightarrow{F_s}. \tag{38}$$

Similarly, we can obtain the following adjoint matrix system:

$$-M_s \frac{d\overrightarrow{W_s^*}}{dt} + A_s \overrightarrow{W_s^*} = \overrightarrow{\tilde{F}_s}. \tag{39}$$

Note that the adjoint matrix $A_s^* = A_s$ because the matrix A_s is a symmetric real matrix.

In this section, we apply the iterative algorithm in the following form:

$$\begin{aligned}
&M_s \frac{\overrightarrow{W_s}^{k+1(j)} - \overrightarrow{W_s}^{k(j)}}{\Delta t} + \theta A_s^{k+1} \overrightarrow{W_s}^{k+1(j)} + (1-\theta) A_s^k \overrightarrow{W_s}^{k(j)} \\
&= \theta \overrightarrow{F_s}^{k+1(j)} + (1-\theta) \overrightarrow{F_s}^{k(j)}, \quad k = 0, ..., M-1 \\
&\overrightarrow{W_s}^{0(j)} = (\overrightarrow{w}^j, \overrightarrow{p_0})^T \\
&-M_s \frac{\overrightarrow{W_s}^{*k+1(j)} - \overrightarrow{W_s}^{*k(j)}}{\Delta t} + \theta A_s^k \overrightarrow{W_s}^{*k(j)} + (1-\theta) A_s^{k+1} \overrightarrow{W_s}^{*k+1(j)} \\
&= \theta \overrightarrow{\tilde{F}_s}^{k(j)} + (1-\theta) \overrightarrow{\tilde{F}_s}^{k+1(j)}, \quad k = 0, ..., M-1 \\
&\overrightarrow{W_s}^{*M(j)} = 0, \\
&\overrightarrow{w}^{j+1} = \overrightarrow{w}^j + \alpha_{j+1} (\overrightarrow{W_v}^{*0(j)} - \alpha \overrightarrow{w}^j) + \beta_{j+1} (\overrightarrow{w}^j - \overrightarrow{w}^{j-1}).
\end{aligned}$$

Here j is the iteration index, $\overrightarrow{W_v}$ is the velocity component of the solution. Parameter α_{j+1} and β_{j+1} can be obtained similarly by applying the conjugate gradient method recalled in section 2.

4.3. NUMERICAL RESULTS FOR STOKES EQUATION

Similar to Section 3.3, we carry out two numerical experiments to validate the iterative algorithm for Stokes equation. The first one is set up with given analytic solutions so that we can compare the errors between the numerical solutions and the analytic solutions in order to demonstrate the parameter sensitivity, convergence, accuracy, and efficiency of the iterative algorithm. The second one is a more realistic numerical test for approximating the optimal control of the variational data assimilation problem.

4.3.1. Numerical Experiment For Validating The Iterative Algorithm. In the first numerical experiment, we consider the target problem of Stokes equation with a given observation vector function $\overrightarrow{\hat{\varphi}} = (\hat{\varphi_1}, \hat{\varphi_2})^T$ to test the iterative algorithm.

Similar to the way of adding $\overrightarrow{\hat{b}}$ in (25) of Section 3.4.1, we artificially add a vector function $\overrightarrow{F}$ and its discretized formulation to the iterative algorithm, which does not affect the convergence property of the algorithm but provides the convenience to set up the first numerical experiment for the convergence of the iterative solution to the analytic solution given below(not the optimal control). Then we can compute the errors between the numerical solutions and the analytic solutions in order to illustrate the properties of the iterative algorithm.

Set $\Omega = [0,1] \times [0,1]$, $\alpha = 1$,

$$\begin{aligned} \hat{\varphi_1} &= (x^5 - x^4 - x^3 - x^2)(5y^4 - 4y^3 - 3y^2 + 2y)cos(2\pi t), \\ \hat{\varphi_2} &= -(5x^4 - 4x^3 - 3x^2 + 2x)(y^5 - y^4 - y^3 + y^2)cos(2\pi t). \end{aligned}$$

The problem is set up with analytic solution

$$\begin{aligned} \varphi_1 &= (x^5 - x^4 - x^3 - x^2)(5y^4 - 4y^3 - 3y^2 + 2y)cos(2\pi t), \\ \varphi_2 &= -(5x^4 - 4x^3 - 3x^2 + 2x)(y^5 - y^4 - y^3 + y^2)cos(2\pi t), \\ p &= sin(\pi x)sin(\pi y)cos(2\pi t). \end{aligned}$$

Hence

$$\begin{aligned} f_1 &= [\pi cos(\pi x)sin(\pi x) + (60x^2 - 24x - 6)(y^5 - y^4 - y^3 + y^2) \\ &\quad -(x^5 - x^4 - x^3 + x^2)(60y^2 - 24y - 6)]cos(2\pi t) \\ &\quad -2\pi(x^5 - x^4 - x^3 + x^2)(5y^4 - 4y^3 - 3y^2 + 2y)sin(2\pi t), \\ f_2 &= [\pi cos(\pi y)sin(\pi x) + (60x^2 - 24x - 6)(y^5 - y^4 - y^3 + y^2) \\ &\quad +(5x^4 - 4x^3 - 3x^2 + 2x)(20y^3 - 12y^2 - 6y + 2)]cos(2\pi t) \\ &\quad +2\pi(5x^4 - 4x^3 - 3x^2 + 2x)(y^5 - y^4 - y^3 + y^2)sin(2\pi t). \end{aligned}$$

Choose Taylor-Hood finite elements for the spatial discretization with step size h. That is, quadratic finite elements are used for the velocity and linear finite elements are used for the pressure. Furthermore, Crank-Nicolson scheme is used for temporal discretization with step size Δt.

The tolerance to stop the iteration is set to be 10^{-6}. Then we obtain the following results for the iterative algorithm. The first step is to test the effects of the initial guess and the mesh size on the iterative algorithm. Tables 4.1-4.5 provides the numerical errors for the solution $\overrightarrow{\varphi}$ at the initial time in different norms and the number k of iteration steps with respect to different initial vector

$$\overrightarrow{w}^0(x,y) = \begin{pmatrix} x^2y^2 \\ x^2y^2 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 10 \\ 10 \end{pmatrix}, \begin{pmatrix} 100 \\ 100 \end{pmatrix},$$

Crank-Nicolson scheme has second order accuracy for the time discretization. Quadratic finite elements have third order accuracy in L^∞/L^2 norms and second order accuracy in H^1 norm for the spatial discretization. Therefore, when we choose $\Delta t \approx h^{3/2}$, we expect the third order accuracy in L^∞/L^2 norms and second order accuracy in H^1 norm for our numerical solution, which can be clearly observed from Tables 4.1-4.5. Using linear regression, the results in Tables 4.1-4.5 satisfy

$$\begin{aligned} \|\overrightarrow{w_h} - \overrightarrow{w}\|_\infty &= 0.1633h^{2.9643}, \\ \|\overrightarrow{w_h} - \overrightarrow{w}\|_0 &= 0.2429h^{2.9690}, \\ \|\overrightarrow{w_h} - \overrightarrow{w}\|_1 &= 0.5656h^{1.9449}. \end{aligned}$$

Furthermore, the small numbers of iteration steps clearly indicate the high effi-

ciency of the iterative algorithm. The numbers of iteration steps also stay almost the same for different initial guess and different step sizes $h(=\Delta t)$, which indicates that the iterative algorithm is not sensitive to the initial guess.

Table 4.1. Numerical results with initial guess equals $(x^2y^2, x^2y^2)^T$

h	Δt	L^∞ error	L^2 error	H^1 error	k
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02	8
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03	8
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03	8
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04	8

Table 4.2. Numerical results with initial guess equals $(-1, -1)^T$

h	Δt	L^∞ error	L^2 error	H^1 error	k
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02	8
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03	8
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03	8
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04	8

Table 4.3. Numerical results with initial guess equals $(1, 1)^T$

h	Δt	L^∞ error	L^2 error	H^1 error	k
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02	7
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03	8
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03	8
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04	8

Table 4.4. Numerical results with initial guess equals $(10, 10)^T$

h	Δt	L^∞ error	L^2 error	H^1 error	k
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02	9
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03	9
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03	9
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04	9

Table 4.5. Numerical results with initial guess equals $(100, 100)^T$

h	Δt	L^∞ error	L^2 error	H^1 error	k
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02	9
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03	9
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03	9
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04	9

The second step is to test the effect of the accuracy of the observational data function on the iterative algorithm. We consider the perturbed observational data functions

$$\begin{aligned}\hat{\varphi_1} &= (x^5 - x^4 - x^3 + x^2)(5y^4 - 4y^3 - 3y^2 + 2y)cos(2\pi t) + r\epsilon, \\ \hat{\varphi_2} &= -(5x^4 - 4x^3 - 3x^2 + 2x)(y^5 - y^4 - y^3 + y^2)cos(2\pi t) + r\epsilon\end{aligned}$$

where r is a random number in $[0, 1]$ and $\epsilon = 10^{-6}, 10^{-4}, 10^{-2}, 1, 10^2$. Then we repeat the same numerical experiment with the iteration number $k = 15$. As expected, we observe from Tables 4.6-4.10 that small perturbations can still provide accuracy numerical results and larger perturbations deteriorate the numerical solutions.

Table 4.6. Numerical results with $\epsilon = 10^{-6}$

h	Δt	L^∞ error	L^2 error	H^1 error
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04

Table 4.7. Numerical results with $\epsilon = 10^{-4}$

h	Δt	L^∞ error	L^2 error	H^1 error
1/4	1/16	2.6871e-03	3.9732e-03	3.8501e-02
1/8	1/32	3.4268e-04	5.0412e-04	9.8409e-03
1/16	1/64	4.3979e-05	6.4528e-05	2.5420e-03
1/32	1/128	5.6490e-06	8.2714e-06	6.7583e-04

Table 4.8. Numerical results with $\epsilon = 10^{-2}$

h	Δt	L^∞ error	L^2 error	H^1 error
1/4	1/16	2.7371e-03	4.0512e-03	3.9317e-02
1/8	1/32	3.5318e-04	5.1273e-04	9.9018e-03
1/16	1/64	4.4126e-05	6.5782e-05	2.5913e-03
1/32	1/128	5.7061e-06	8.3259e-06	6.9147e-04

Table 4.9. Numerical results with $\epsilon = 1$

h	Δt	L^∞ error	L^2 error	H^1 error
1/4	1/16	2.8523e-03	4.1863e-03	4.0782e-02
1/8	1/32	3.6718e-04	5.2319e-04	1.0562e-02
1/16	1/64	4.5179e-05	6.6823e-05	2.7310e-03
1/32	1/128	5.8437e-06	8.4613e-06	7.0715e-04

Table 4.10. Numerical results with $\epsilon = 10^2$

h	Δt	L^∞ error	L^2 error	H^1 error
1/4	1/16	3.5138+e00	5.3621e+00	8.2739e+00
1/8	1/32	3.7461+e00	5.3426e+00	8.4578e+00
1/16	1/64	3.7232+e00	5.6253e+00	8.5937e+00
1/32	1/128	4.2437e+00	6.1247e+00	8.7121e+00

4.3.2. Iterative Algorithm For Approximating The Optimal Control of A More Realistic Problem For Stokes Equation. In the second numerical experiment, we consider the target problem of Stokes equation with given observation vector function $\overrightarrow{\hat{\varphi}} = (\hat{\varphi}_1, \hat{\varphi}_2)^T$ for seeking the optimal control vector $\overrightarrow{w}$. We do not artificially add the vector $\overrightarrow{F}$ so that our iterative solution could converge to the optimal control.

Set $\Omega = [0, 1] \times [-0.25, 0]$,

$$\begin{aligned} \hat{\varphi}_1 &= (x^5 - x^4 - x^3 - x^2)(5y^4 - 4y^3 - 3y^2 + 2y)cos(2\pi t) + 10^{-3}r, \\ \hat{\varphi}_2 &= -(5x^4 - 4x^3 - 3x^2 + 2x)(y^5 - y^4 - y^3 + y^2)cos(2\pi t) + 10^{-3}r \end{aligned}$$

where r is a random number between 0 and 1. The analytic solution, f_1, and f_2 are the same as those in the Section 4.3.1. Here we take the initial vector to be $\overrightarrow{w}$ whose elements are all 1 and set the tolerance to be 10^{-6} with $h = 1/4$ and $\Delta t = 1/16$. Table 4.11 provides the numerical errors in the solution $\overrightarrow{\varphi}$ at the initial time and the number k of the iteration steps for seeking the optimal control

vector $\overrightarrow{w}$ with different parameter α. Recall the cost functional defined in Section 4.1,

$$J(\overrightarrow{w}) = \frac{\alpha}{2}\|\overrightarrow{w}\|^2 + \frac{1}{2}\int_0^T \|\overrightarrow{\hat{\varphi}} - \overrightarrow{\varphi}\|^2 dt$$

where the wight coefficient $\alpha > 0$, $\hat{\varphi}(t)$ is a given function generally defined by the priori observational data, and $\|\cdot\|$ is the norm in a Hilbert space H.

Since α is the weight coefficient of the cost of the control in the cost function, we expect that smaller α can improve the accuracy with an increased cost. This is verified by the decreased errors and increased number of iteration steps in Table 4.11.

Table 4.11. Numerical results for different α

α	L^∞ error	L^2 error	H^1 error	k
1	2.7162e-01	3.5856e-01	8.7990e-01	11
0.5	2.4391e-01	3.3872e-01	8.4736e-01	14
0.2	1.8475e-01	2.7743e-01	7.8216e-01	18
0.1	1.4317e-01	2.3651e-01	6.9637e-01	23
0.01	9.9247e-03	1.5392e-02	1.2736e-01	55
0.001	2.8461e-03	4.1038e-03	4.0318e-02	81

5. CONCLUSIONS

In this thesis, we studied an iterative algorithm [23] with finite elements for the variational data assimilation. This iterative algorithm was applied with the corresponding discretization formulations of the model equations to approximate the optimal control in the variational data assimilation problems. For the three stages at each step of iteration, we first solved the original forward equation and the backward equation with finite elements and finite difference schemes, and then updated the optimal control for the next iteration step with conjugate gradient method. We conducted a group of comprehensive numerical experiments for both the second order parabolic equation and Stokes equation.

We first reviewed the formation of the optimal control problem of variational data assimilation and the corresponding iterative algorithm in [23]. Then we followed [23] to apply the iterative algorithm to the second order parabolic equation in Section 3 for more numerical tests by discretizing the operator formulation into its discretized matrix formulation, and extended the study to Stokes equation in Section 4 based on the corresponding dicretized matrix formulation.

Numerical experiments were carried out for the parameter sensitivity, convergence, accuracy, and efficiency of the algorithm. The numerical results demonstrate the optimal accuracy orders from the numerical errors and the fast convergence from the small number of iteration steps. It is also observed that the numbers of iteration steps stay almost the same for different initial guesses and different step sizes $h(= \Delta t)$. Moreover, as expected, small perturbations to the observational data function can still provide accurate enough numerical results and increasing perturbations deteriorate the numerical solutions. From the numerical experiment for the weight coefficient α in the cost function, we can see that smaller α can improve the accuracy with an increased cost as expected based on the definition of the cost function.

One interesting and promising future work is to extend the fundamental study and numerical experiments in this thesis to more realistic and sophisticated models with different boundary conditions, such as the interface Darcy model and the Stokes-Darcy model for subsurface flow.

BIBLIOGRAPHY

[1] L. Pacific Northwest National and E. United States. Dept. of and S. United States. Dept. of Energy. Office of and I. Technical. (2009), An Assessment of the Commercial Availability of Carbon Dioxide Capture and Storage Technologies as of June 2009. Available: http://www.osti.gov/servlets/purl/967229-7rduSf/.

[2] E. United States. Dept. of. Basic research needs for geosciences: facilitating 21st centuray energy systems. *Bethesda, MDFebruary*, pages 21–23, 2007.

[3] General technical support document for injection and geologic sequestration of carbon dioxide: subparts RR and UU, ed. 2010.

[4] S. L. Barnes. A technique for maximizing details in numerical weather map analysis. *J. Appl. Meteor.*, (3):396–409, 1964.

[5] E. Blayo, E. Cosme, M. Nodet, and A. Vidard. Introduction to data assimilation. 2011.

[6] F. Bouttier and O. Talagrand. Date assimilation concepts and methods. *Meteorological Training Lecture Notes, ECMWF, Shinfield Park, Reading*, 1999.

[7] G. P. Cressman. An Operational Objective Analysis System. *Mon. Wea. Rev*, 87:367–374, 1959.

[8] D. N. Daescu and I. M. Navon. An analysis of a hybrid optimization method for variational data assimilation. *International Journal of Computational Fluid Dynamics*, 17:299–306, 2003.

[9] F.-X. L. Dimet and V. P. Shutyaev. On Newton methods in data assimilation. *Russ. J. Numer. Anal. Math. Modelling*, 15(5):419–434, 2000.

[10] G. Evensen. *Data assimilation : The ensemble Kalman filter*. Springer, Berlin, 2007.

[11] G. Evenson. The Ensemble Kalman Filter: theoretical formulation and practical implementation . *Ocean Dynamics*, 53:343–367, 2003.

[12] V. Girault and P.-A. Raviart. Finite element methods for Navier-Stokes equations. *Springer Series in Computation Mathematics*, 5:376, 1986.

[13] M. D. Gunzburger. *Finite element methods for viscous incompressible flows.* Academic Press Inc, 1989.

[14] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[15] W. J. Layton, F. Schieweck, and I. Yotov. Coupling fluid flow with porous media flow. *SIAM J. Numer. Anal.*, 6(40):2195–2218, 2002.

[16] P. Lynch. The origins of computer weather prediction and climate modeling. *Journal of Computational Physics*, (227):3431–3444, 2008.

[17] G. Marchuk and V. Agoshkov. On solvability and numerical solution of data assimilation problems. *Russ. J. Numer. Anal. Math. Modelling*, 8:1–16, 1993.

[18] G. I. Marchuk and V. Shutyaev. Iteration methods for solving a data assimilation problem. *Russ. J. Numer. Anal. Math. Modelling*, 9:265–279, 1993.

[19] G. I. Marchuk and V. B. Zalesny. A numerical technique for a geophysical data assimilation problem using Pontryagin's principle and splitting -up method. *Russ.J.Numer.Anal.Math.Modelling*, 8:311–326, 1993.

[20] S. Martens, T. Kempka, A. Liebscher, S. Lth, F. Mller, A. Myrttinen, B. Norden, C. Schmidt-Hattenberger, M. Zimmer, and M. Khn. Europe's longest-operating on-shore CO_2 storage site at Ketzin, Germany: a progress report after three years of injection. *Environmental Earth Sciences*, pages 1–12, 2012.

[21] B. Metz and G. I. P. on Climate Change. Working, III, IPCC special report on carbon dioxide capture and storage. *Cambridge: Cambridge University Press for the Intergovernmental Panel on Climate Change*, 2005.

[22] I. M. Navon. Data assimilation for numerical weather prediction : a review.

[23] E. I. Parmuzin and V. Shutyaev. Numerical analysis of iterative methods for solving evolution data assimilation problems. *Russ. J. Numer. Anal. Math. Modelling*, 14:275–289, 1999.

[24] L. Paterson, J. Ennis-King, and S. Sharma. Observations of thermal and pressure transients in carbon dioxide wells. pages 3449–3460, 2010.

[25] L. F. Richardson. *Weather Prediction by Numerical Process.* Cambridge, 1922.

[26] F. A. Rihan, C. G. Collier, and I. Roulstone. Four-dimensional variational data assimilation for Doppler radar wind data. *Journal of Computational and Applied Mathematics*, 176:15–34, 2005.

[27] O. Talagrand. Assimilation of observations, an introduction. *J Meteorol Soc Japan*, 1B(75):191–209, 1997.

[28] C. M. White, B. R. Strazisar, E. J. Granite, J. S. Hoffman, and H. W. Pennline. Separation and Capture of CO_2 from Large Stationary Sources and Sequestration in Geological FormationsCoalbeds and Deep Saline Aquifers. *Journal of the Air Waste Management Association*, 53:645–715, 2003.

[29] G. Zambrano-Narvaez and R. Chalaturnyk. Case study of the cementing phase of an observation well at the Pembina Cardium CO2 monitoring pilot, Alberta, Canada. *International Journal of Greenhouse Gas Control*, 5:841–849, 2011.

[30] F. Zhang, C. Juhlin, C. Cosma, A. Tryggvason, and R. G. Pratt. Cross-well seismic waveform tomography for monitoring CO_2 injection: A case study from the Ketzin Site, Germany. *Geophysical Journal International*, 189:629–646, 2012.

VITA

Xin Shen was born in Mianyang, Sichuan, China. In June 2011, he received his B.S. in Optical Information Science and Technology from Sichuan University, Chengdu, China. He went to work for an year before he became a graduate student in Mathematics from Missouri University of Science and Technology.

As a student, he had an excellent academic performance and participated in some research projects. He also volunteered in some local events and community activities.