

Syracuse University

SURFACE

Dissertations - ALL

SURFACE

June 2018

Implicit Fixed-point Proximity Framework for Optimization Problems and Its Applications

Xiaoxia Liu
Syracuse University

Follow this and additional works at: <https://surface.syr.edu/etd>



Part of the [Physical Sciences and Mathematics Commons](#)

Recommended Citation

Liu, Xiaoxia, "Implicit Fixed-point Proximity Framework for Optimization Problems and Its Applications" (2018). *Dissertations - ALL*. 897.
<https://surface.syr.edu/etd/897>

This Dissertation is brought to you for free and open access by the SURFACE at SURFACE. It has been accepted for inclusion in Dissertations - ALL by an authorized administrator of SURFACE. For more information, please contact surface@syr.edu.

Abstract

A variety of optimization problems especially in the field of image processing are not differentiable in nature. The non-differentiability of the objective functions together with the large dimension of the underlying images makes minimizing the objective function theoretically challenging and numerically difficult. The fixed-point proximity framework that we will systematically study in this dissertation provides a direct and unified methodology for finding solutions to those optimization problems. The framework approaches the models arising from applications straightforwardly by using various fixed point techniques as well as convex analysis tools such as the subdifferential and proximity operator.

With the notion of proximity operator, we can convert those optimization problems into finding fixed points of nonlinear operators. Under the fixed-point proximity framework, these fixed point problems are often solved through iterative schemes in which each iteration can be computed in an explicit form. We further explore this fixed point formulation, and develop implicit iterative schemes for finding fixed points of nonlinear operators associated with the underlying problems, with the goal of relaxing restrictions in the development of solving the fixed point equations. Theoretical analysis is provided for the convergence of implicit algorithms proposed under the framework. The numerical experiments on image reconstruction models demonstrate that the proposed implicit fixed-point proximity algorithms work well in comparison with existing explicit fixed-point proximity algorithms in terms of the consumed computational time and accuracy of the solutions.

Implicit Fixed-point Proximity Framework for Optimization Problems and Its Applications

By

Xiaoxia Liu

B.S., Sun Yat-sen University, 2012

M.S., Syracuse University, 2016

Dissertation

Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Mathematics

Syracuse University

June 2018

Copyright © Xiaoxia Liu 2018

All Rights Reserved

Acknowledgments

First and foremost, I would like to express my deepest gratitude to both of my advisors, Professor Yuesheng Xu and Professor Lixin Shen, for their patient guidance, constant inspiration and generous support in the past six years. They have not only taught me knowledge in convex analysis and fixed point theory, skills in problem posing and problem solving, and techniques in scientific writing and presenting, but they have also enlightened me to fulfill my academic dreams with their expertise, rigorous thinking and enthusiasm. All their guidance will be beneficial to my future career and life. In addition, I greatly appreciate the time that they spent in discussing with me, providing me insightful feedback and correcting my mistakes. I would never have been able to finish my dissertation without their help.

I also sincerely appreciate my committee members, Professors Dan Coman, Eugene Poletsky, Minghao Rostami and Bei Yu for taking time to review my dissertation and providing helpful feedback. I am grateful to the faculty members and staff at Syracuse University, especially Professors Uday Banerjee, Pinyuen Chen, Graham Leuschke, Adam Lutoborski, and Grace Wang for their inspiring courses and help.

I am thankful to all my colleagues and friends for making my life in Syracuse enjoyable. I would like to thank, Feishe Chen, Lei Chen, Stephen Farnham, Rachel Gettinger, James Heffers, Qian Huang, Lei Jin, Muzhi Jin, Hyesu Kim, Si Li, Yuzhen Liu, Wenting Long, Xuefei Ma, Hani Sulieman, Erin Tripp, Agya Wagya and Yihe Yu. Special thanks to my lovely friends, Casey Necheles, Chen Liang, Alice Lim, Mkrtich Ohanyan, Mingyue Wang, Jinxia Xie and Jing Zhou for their warm encouragement and support not only through the

dissertation process but also my entire Ph.D. study.

Finally, I would like to express my sincerest appreciation to my parents Sumei Xie and Chaoxin Liu, and my siblings for their selfless love and faith. Special thanks to my beloved partner Dingqiang Ma for his continued understanding, patience and support over these years.

Contents

Acknowledgements	iv
1 Introduction	1
1.1 Problem Statement	1
1.2 Literature Review	5
1.3 Motivations	7
1.4 Contributions of This Dissertation	8
1.5 Organization of This Dissertation	9
2 Implicit Fixed-point Proximity Framework	10
2.1 Notations and Preliminaries	11
2.1.1 Preliminaries in convex analysis	12
2.1.2 Preliminaries in fixed point theory	15
2.2 Fixed Point Characterization	17
2.3 Fixed-point Proximity Framework	20
2.4 Existing Fixed-point Proximity Algorithms	24
2.5 Implicit Fixed-point Proximity Framework	29
2.5.1 Preliminaries on contractive mappings	30
2.5.2 Implicit fixed-point proximity algorithm	31
2.5.3 Property 1: \mathcal{L}_M is an M -operator	33
2.5.4 Property 2: \mathcal{L}_M can be evaluated efficiently	35

2.5.5	Property 3: \mathcal{L}_M is nonexpansive	36
2.5.6	Possible block structures of M for implicit algorithms	39
2.5.7	Convergence analysis	40
3	Examples of Implicit Fixed-point Proximity Algorithms	48
3.1	Example 1	49
3.1.1	Fixed point characterization	50
3.1.2	Existing fixed-point proximity algorithms	52
3.1.3	Implicit fixed-point proximity algorithm	53
3.1.4	Convergence analysis	55
3.2	Example 2	59
3.2.1	Fixed point characterization	59
3.2.2	Existing fixed-point proximity algorithms	61
3.2.3	Implicit fixed-point proximity algorithms	62
3.2.4	Convergence analysis	66
3.3	Other Examples	76
3.3.1	Fixed point characterization	76
3.3.2	Implicit fixed-point proximity algorithms	78
4	Applications in Image Processing and Numerical Experiments	80
4.1	Applications in Image Processing	80
4.1.1	Image reconstruction models	81
4.1.2	Total variation regularization	81
4.2	Application of Example 1: Total Variation Based Denoising Models	82
4.2.1	Parameter settings	83
4.2.2	ROF model	84
4.2.3	L1-TV model	87
4.2.4	Sensitivity of parameter settings	90

4.3	Application of Example 2: Total Variation Based Deblurring Models	94
4.3.1	Parameter settings	95
4.3.2	L2-TV model	95
4.3.3	L1-TV model	97
5	Conclusion and Future Work	100

Chapter 1

Introduction

1.1 Problem Statement

In this dissertation, the optimization problem of our interest is to minimize a sum of convex functions composed with linear operators

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^N f_i(A_i x), \quad (\text{P0})$$

where \mathbb{R}^n denotes the usual n -dimensional Euclidean space, A_i is an $m_i \times n$ matrix, and $f_i : \mathbb{R}^{m_i} \rightarrow (-\infty, +\infty]$ is proper, lower semi-continuous and convex, $i = 1, 2, \dots, N$.

The optimization problem (P0) is motivated by many applications including image processing and machine learning. Some examples of its applications are shown as follows.

(i) Applications in image processing

In image processing, an observed image z degraded by blurring or/and noise can be modeled as $z = Kx + \eta$, where x represents the unknown image to be recovered, K represents the measurement process, and η represents the additive noise. Image reconstruction models for approximating x are usually formulated as a sum of one data fidelity term and at least one regularization term. The data fidelity term measures,

adapting to the noise type of η , the similarity between the observed image z and the desired image x . The regularization term is included to explore the prior structures of the underlying image. The most popular choice of the regularization term in image processing is the total variation [10, 74] defined as

$$\|\cdot\|_{TV} = \psi \circ D,$$

where D is a first order difference matrix, and ψ is the ℓ_1 norm $\|\cdot\|_1$ or a certain linear combination of the ℓ_2 norm $\|\cdot\|_2$ in \mathbb{R}^2 [32, 51]. Besides the total variation regularization, there are other regularization terms such as the framelet regularization term [7, 8, 47], which is defined as the ℓ_1 norm of the framelet coefficients of the underlying image under a framelet transformation.

In each of the following image reconstruction models, the first term is corresponding to the data fidelity term, the second term (and the third term if any) is corresponding to the regularization term, and $\lambda > 0$ and $\mu > 0$ are model parameters.

- *Rudin-Osher-Fatemi (ROF) Model* [74]: The model is designed to restore an image contaminated by Gaussian noise, and can be written as problem (P0) with $f_1 = \frac{\lambda}{2} \|\cdot - z\|_2^2$, $A_1 = I$, $f_2 = \psi$, $A_2 = D$, i.e.,

$$\min_x \frac{\lambda}{2} \|x - z\|_2^2 + \|x\|_{TV}. \quad (\text{M1})$$

- *L1-TV Denoising Model* [12, 65]: The model is designed to restore an image contaminated by impulsive noise (or called pepper-and-salt noise), and can be written as problem (P0) with $f_1 = \lambda \|\cdot - z\|_1$, $A_1 = I$, $f_2 = \psi$, $A_2 = D$, i.e.,

$$\min_x \lambda \|x - z\|_1 + \|x\|_{TV}. \quad (\text{M2})$$

- *L2-TV Image Restoration Model* [2, 13, 67]: The model is designed to restore a blurry image contaminated by Gaussian noise, and can be written as problem (P0) with $f_1 = \frac{\lambda}{2} \|\cdot - z\|_2^2$, $A_1 = K$, $f_2 = \psi$, $A_2 = D$, i.e.,

$$\min_x \frac{\lambda}{2} \|Kx - z\|_2^2 + \|x\|_{TV}. \quad (\text{M3})$$

- *L1-TV Image Restoration Model* [22, 41]: The model is designed to restore a blurry image contaminated by impulsive noise, and can be written as problem (P0) with $f_1 = \lambda \|\cdot - z\|_1$, $A_1 = K$, $f_2 = \psi$, $A_2 = D$, i.e.,

$$\min_x \lambda \|Kx - z\|_1 + \|x\|_{TV}. \quad (\text{M4})$$

- *Framelet Based Image Reconstruction Models* [6, 34, 54]: The models are similar to the total variation based image reconstruction models mentioned above, except that the total variation regularization term is replaced by the framelet regularization term [7, 8], which can be written as a composition of the ℓ_1 norm $\|\cdot\|_1$ and a linear operator D where D represents a tight frame system generated from framelets [47].
- *MR Image Reconstruction Model* [46, 58, 59, 84]: The model is designed to reconstruct an MR image x that is sparse in the wavelet domain. Let Φ be a given sampling matrix, b be observed measurements, and W be a wavelet transform. The model can be written as problem (P0) with $f_1 = \frac{1}{2} \|\cdot - b\|_2^2$, $A_1 = \Phi$, $f_2 = \psi$, $A_2 = D$, $f_3 = \|\cdot\|_1$, $A_3 = W$, i.e.,

$$\min_x \frac{\lambda}{2} \|\Phi x - b\|_2^2 + \mu \|x\|_{TV} + \|Wx\|_1. \quad (\text{M5})$$

(ii) Applications in machine learning

Various optimization problems arising from machine learning seek to minimize a loss

function possibly along with a regularization term. The loss function describes the expected cost, and the most frequently used regularization term is the ℓ_1 regularization term that enforces sparsity on the desired solution in order to avoid over-fitting [78].

- *The ℓ_1 -regularized Linear Least Squares Problem:* This problem is known as *Basis Pursuit* [9, 21, 29] in compressive sensing and Least Absolute Shrinkage and Selection Operator (*LASSO*) [78] in machine learning and statistics. Basis pursuit is designed to recover a sparse signal x from compressed measurements. Let Φ be a given sampling matrix, b be observed measurements, and λ be a model parameter. The model can be written as problem (P0) with $f_1 = \frac{\lambda}{2} \|\cdot - b\|_2^2, A_1 = \Phi, f_2 = \|\cdot\|_1, A_2 = I$, i.e.,

$$\min_x \frac{\lambda}{2} \|\Phi x - b\|_2^2 + \|x\|_1. \quad (\text{M6})$$

- *The ℓ_1 -regularized Classification Model* [28, 76, 81]: The model is designed for classifying data by using a linear classifier in machine learning. Suppose $w \in \mathbb{R}^n$ is the sparse coefficients of the linear classifier to be computed, $b \in \mathbb{R}$ is the bias of the linear classifier, $\lambda > 0$ is a model parameter, $(x_i, y_i) \in \mathbb{R}^n \times \mathbb{R}$ are the given data points, and $l_i : \mathbb{R} \rightarrow \mathbb{R}$ are loss functions, $i = 1, 2, \dots, N$. The model can be written as problem (P0) with $f_1(w) = \lambda \sum_{i=1}^N l_i(y_i(w^\top x_i + b)), A_1 = I_n, f_2 = \|\cdot\|_1, A_2 = I_n$, i.e.,

$$\min_w \lambda \sum_{i=1}^N l_i(y_i(w^\top x_i + b)) + \|w\|_1.$$

There are several classification models that can be written in this form with different choice of the loss function l_i . For example, support vector machine (SVM) [28, 81] uses the hinge loss function and logistic regression optimization [76] uses the logistic loss function.

- *Consensus Optimization* [48, 64, 83]: The model is designed to minimize the total

cost in a network with N agents. Suppose x is the solution to be computed, and f_i is the cost function of the i -th agent, $i = 1, 2, \dots, N$. The model can be written as problem (P0) with $A_i = I$, i.e.,

$$\min_x \sum_{i=1}^N f_i(x).$$

All the optimization problems mentioned above can be solved by the algorithms developed under the implicit fixed-point proximity framework that we will propose in this dissertation. Depending on the number of convex functions and the number of composite linear operators, the applicable algorithms are different. We will illustrate in detail the fixed-point proximity framework with an emphasis on implicit algorithms in Chapter 2.

1.2 Literature Review

The fixed-point proximity framework for composite optimization problems has been extensively studied in recent years due to its ease of applicability. The framework relies on the notion of proximity operator, which was introduced early in [62, 73] and widely adopted to applications arising from image processing (see, e.g., [4, 5, 27]). Under the fixed-point proximity framework, the solutions of the optimization problem are characterized as fixed points of a mapping defined in terms of proximity operator, thereby allowing for the development of efficient numerical methods via various powerful fixed point iterations.

The first algorithm, developed from the perspective of both proximity operator and fixed point theory, was the fixed point problem algorithm based on proximity operator (FP²O) [60] designed to solve the ROF model (M1) for image denoising. Accordingly, this fixed-point proximity approach has been extended to handle the L1-TV model (M2) for image denoising [17, 50, 61], the basis pursuit model (M6) for compressive sensing [15], the TV-regularized MAP ECT reconstruction model for ECT image reconstruction [49], the exp-model for removing multiplicative noise [56, 57], and other models [53].

Various algorithms have been proposed since then by employing the fixed-point proximity framework along with other techniques in convex analysis and numerical analysis. For example, the multi-step fixed-point proximity algorithm [51, 52] introduced the multi-step scheme into the framework; the primal-dual fixed point algorithm based on proximity operator (PDFP²O) [18–20] combined the framework with the primal dual formulation [11, 69, 82]; the fixed-point proximity Gauss-Seidel algorithm (FPGS) [16] utilized the Gauss-Seidel method and a parameters relaxation technique in addition to the framework.

The fixed-point proximity framework has been demonstrated in the literature to be a powerful tool for composite optimization problems [51]. On one hand, the framework provides a general platform to explore new algorithms for different optimization problems. On the other hand, the framework offers new insights on existing algorithms and puts forward new improvements.

Many existing algorithms can be identified as a fixed-point proximity algorithm and be reinterpreted under the framework, even though they are developed from different perspectives. We classify these algorithms roughly into two categories. The first category of algorithms are developed from the Fenchel-Rockafellar duality theory [25, 72] and have the primal dual formulation [69, 82]. The primal dual method formulates a primal problem and a dual problem, and then updates the primal variable and dual variable alternatively. For example, first order primal dual algorithm (PD) [11, 69, 82], primal-dual hybrid gradient algorithm (PDHG) [33, 87], and contraction-type primal dual algorithm [43] are considered as fixed-point proximity algorithms in the first category. The second category of algorithms are developed from the augmented Lagrangian technique [37, 44, 70, 71]. The augmented Lagrangian method minimizes the augmented Lagrangian function of the equality-constrained optimization problem, and then updates the Lagrange multipliers. For example, augmented Lagrangian method (ALM) [35–37, 44, 70, 71], alternating direction method of multipliers (ADMM) [3, 31, 36], and alternating minimization algorithm (AMA) [80] are considered as fixed-point proximity algorithms in the second category. Some algorithms based on splitting

techniques are shown to be closely related to the algorithms in the second category, including Douglas-Rachford splitting algorithm (DRSA) [26, 31], alternating split Bregman iteration (ASBI) [39], and Bregman operator splitting algorithm (BOS) [86].

1.3 Motivations

Among the algorithms for solving composite optimization problems, various algorithms can be reformulated as fixed-point proximity algorithms and further analyzed under the fixed-point proximity framework. Under this framework, the optimization problems are converted into fixed point problems in relation to proximity operators, and then be solved through iterative schemes. The existing fixed-point proximity algorithms all have an *explicit* iterative scheme so that the algorithms can be computed efficiently. However, we observe that there are some restrictions of those algorithms due to the explicitness. We are motivated to develop fixed-point proximity algorithms with a *fully implicit* scheme, because of the following restrictions of existing explicit fixed-point proximity algorithms.

First, the convergence assumptions of explicit fixed-point proximity algorithms may be relatively strict. For example, the primal dual algorithm (PD) has a relatively restricted selection range for the parameters of the proximity operators, which significantly influences the performance of the algorithm. That is due to the limitations of its underlying algorithm structure which is formed to maintain the explicit expression of the algorithm.

Second, the explicit fixed-point proximity algorithms may only be applicable to limited types of composite optimization problems. For example, the fixed point problem algorithm based on proximity operator (FP²O) and the alternating split Bregman iteration (ASBI) are designed to solve problems with quadratic functions. Additional numerical methods are employed to achieve the explicitness of those algorithms. But those numerical methods require additional assumptions not only on the parameters of the proximity operators but also on the objective function, which restricts the applicable range of the algorithms.

Therefore, we aim to study fixed-point proximity algorithms with a fully implicit scheme, because the implicit schemes allow more flexibility while building the structures of the algorithms, and have a potential to yield an algorithm that outperforms existing explicit algorithms.

1.4 Contributions of This Dissertation

In this dissertation, we establish an implicit fixed-point proximity framework that serves as a guideline for developing implicit iterative algorithms applied to composite optimization problems. We also propose several implicit algorithms under the framework with algorithm structures that are not observed in existing fixed-point proximity algorithms.

The two main contributions of this dissertation are summarized as follows.

- The first one is that we enrich the existing fixed-point proximity framework by analyzing fixed-point proximity algorithms with a fully implicit scheme. The existing framework is designed for developing algorithms with an explicit expression and is not applicable for developing algorithms with a fully implicit expression. Our proposed framework employs fixed point techniques, including contractive mappings, to address the issues that may occur while developing implicit algorithms. Theoretical results are provided to guarantee the convergence of implicit algorithms.
- The second one is that we propose two algorithm structures and develop several implicit fixed-point proximity algorithms for different composite optimization problems. We are not aware of any existing fixed-point proximity algorithms that possessed the proposed algorithm structures. And numerical experiments demonstrate that the implicit algorithms with the proposed algorithm structures outperform the existing fixed-point proximity algorithms in terms of computational time.

1.5 Organization of This Dissertation

This dissertation is organized in the following manner. In Chapter 2, we present the implicit fixed-point proximity framework. We start from fixed-point proximity equations that characterize the solutions of a composite optimization problem, and then build implicit algorithms via contractive mappings with comprehensive theoretical convergence results. In Chapter 3, we propose several implicit algorithms for different optimization problems under the implicit fixed-point proximity framework. For each implicit algorithm, we conduct a convergence analysis with theoretical results. In Chapter 4, we test the proposed implicit proximity algorithms on several image reconstruction models and demonstrate the practical performance of the proposed implicit algorithms over other existing explicit fixed-point proximity algorithms. Finally, some conclusions and future work are presented in Chapter 5.

Chapter 2

Implicit Fixed-point Proximity Framework

In this chapter, we first review the existing fixed-point proximity framework for developing explicit algorithms applied to composite optimization problems, and then propose a framework for developing fixed-point proximity algorithms with fully implicit schemes.

In order to formulate the fixed-point proximity framework in a general setting, we consider the optimization problem in the following general form

$$\min_{x \in \mathbb{R}^n} f(Ax), \tag{P1}$$

where A is an $m \times n$ matrix, and $f : \mathbb{R}^m \rightarrow (-\infty, +\infty]$ is proper, lower semi-continuous, and convex. By defining

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_N \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix},$$

and $f(y) = \sum_{i=1}^N f_i(y_i)$, problem (P0) can be written as problem (P1).

The rest of this chapter is organized in the following manner. In Section 2.1, we present

notations and recall some preliminary results in convex analysis and fixed point theory. Before illustrating the proposed implicit fixed-point proximity framework for problem (P1), we formulate in Section 2.2 fixed-point proximity equations that characterize the solutions of problem (P1), and then present a summary of the existing fixed-point proximity framework in Section 2.3 and a review of one class of existing fixed-point proximity algorithms in Section 2.4. Lastly, we present our main result in Section 2.5. We establish the implicit fixed-point proximity framework, which serves as a guideline for developing implicit algorithms.

2.1 Notations and Preliminaries

Let us introduce some notations and recall some preliminary results in convex analysis and fixed point theory.

For given $x, y \in \mathbb{R}^d$, $\langle x, y \rangle := \sum_{i=1}^d \langle x_i, y_i \rangle$ is the standard inner product and $\|x\|_2 := \sqrt{\langle x, x \rangle}$ is the standard ℓ_2 norm. Let \mathbb{S}_+^d (resp. \mathbb{S}^d) denote the set of symmetric positive definite (resp. semi-definite) matrices of size $d \times d$ and let I_d denote the identity matrix of size $d \times d$. For given $x, y \in \mathbb{R}^d$ and a given $H \in \mathbb{S}_+^d$, $\langle x, y \rangle_H := \langle x, Hy \rangle$ is the H -weighted inner product and $\|x\|_H := \sqrt{\langle x, x \rangle_H}$ is the H -weighted ℓ_2 norm. If H is the identity matrix, then the H -weighted inner product reduces to the standard inner product, and the H -weighted ℓ_2 norm reduces to the standard ℓ_2 norm.

Let $f : \mathbb{R}^d \rightarrow (-\infty, +\infty]$. The domain of f is a set in \mathbb{R}^d defined as $\text{dom } f := \{x \in \mathbb{R}^d : f(x) < +\infty\}$. The function f is proper if $\text{dom } f \neq \emptyset$. The function f is lower semi-continuous at $a \in \mathbb{R}^d$, if $f(a) \leq \underline{\lim}_{x \rightarrow a} f(x)$. The function f is convex if $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$, for all $x, y \in \text{dom } f$, and all $\lambda \in (0, 1)$. The class of proper, lower semi-continuous, and convex functions from \mathbb{R}^d to $(-\infty, +\infty]$ is denoted as $\Gamma_0(\mathbb{R}^d)$.

2.1.1 Preliminaries in convex analysis

Let us present some preliminaries in convex analysis, which will be used in this dissertation.

First, we recall the definitions of several important concepts in convex analysis, including the subdifferential, conjugate, and proximity operator of a convex function.

For $f \in \Gamma_0(\mathbb{R}^d)$, the *subdifferential* of f at $x \in \mathbb{R}^d$ is a set in \mathbb{R}^d defined as

$$\partial f(x) := \{y \in \mathbb{R}^d : f(u) \geq f(x) + \langle y, u - x \rangle \text{ for } \forall u \in \mathbb{R}^d\}. \quad (2.1)$$

The elements of $\partial f(x)$ are called the subgradients of f at x . Furthermore, $\partial f(x)$ is a nonempty compact set for all $x \in \text{dom } f$. In particular, if f is differentiable, then $\partial f(x) = \{\nabla f(x)\}$.

The *conjugate* of f is a mapping from \mathbb{R}^d to $(-\infty, +\infty]$ defined as

$$f^*(y) := \sup_{x \in \mathbb{R}^d} \{\langle x, y \rangle - f(x)\}.$$

If $f \in \Gamma_0(\mathbb{R}^d)$, then the conjugate $f^* \in \Gamma_0(\mathbb{R}^d)$ and $\partial f^*(y)$ is a nonempty compact set for all $y \in \text{dom } f^*$. The subdifferentials of f and f^* have the following relationship

$$y \in \partial f(x) \quad \text{if and only if} \quad x \in \partial f^*(y). \quad (2.2)$$

The *proximity operator* of f with respect to $H \in \mathbb{S}_+^d$ is a mapping from \mathbb{R}^d to \mathbb{R}^d defined as

$$\text{prox}_{f,H}(u) := \underset{x \in \mathbb{R}^d}{\text{argmin}} \left\{ f(x) + \frac{1}{2} \|x - u\|_H^2 \right\}.$$

In particular, if $H = \frac{1}{\alpha} I_d$, $\alpha > 0$, then $\text{prox}_{f,H}(u)$ reduces to the proximity operator with parameter α with respect to the standard ℓ_2 norm, denoted as $\text{prox}_{\alpha f}(u)$.

The proximity operator of f with respect to $H \in \mathbb{S}_+^d$ is closely related to the subdifferential

of f . The equation $x = \text{prox}_{f,H}(u)$ holds if and only if x is the unique solution of the following implicit problem

$$\begin{cases} x = u - H^{-1}y \\ y \in \partial f(x). \end{cases} \quad (2.3)$$

The result above can be reinterpreted as follows by redefining the variable y in (2.3) as $H^{-1}y$,

$$Hy \in \partial f(x) \quad \text{if and only if} \quad x = \text{prox}_{f,H}(x + y). \quad (2.4)$$

The relationship between the proximity operators of f and its conjugate f^* is given by Moreau's identity

$$x = \text{prox}_{f^*,H}(x) + H^{-1}\text{prox}_{f,H^{-1}}(Hx). \quad (2.5)$$

If $f : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ can be written as $f(x) = \sum_{i=1}^N f_i(x_i)$, where $f_i : \mathbb{R}^{d_i} \rightarrow (-\infty, +\infty]$, $x_i \in \mathbb{R}^{d_i}$, $i = 1, \dots, N$, $d = \sum_{i=1}^N d_i$ and

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix},$$

then f is a block separable sum of f_i 's, $i = 1, 2, \dots, N$. The i th block of the proximity operator of f is evaluated by the proximity operator of the i th separable part, that is,

$$(\text{prox}_f(x))_i = \text{prox}_{f_i}(x_i),$$

for $i = 1, 2, \dots, N$.

Second, we present two examples for which we can explicitly compute their subdifferentials, conjugates and proximity operators.

Example 1: (Quadratic Function) We have $\partial(\frac{1}{2}\|\cdot\|_2^2)(x) = \{x\}$, $(\frac{1}{2}\|\cdot\|_2^2)^* = \frac{1}{2}\|\cdot\|_2^2$, and $\text{prox}_{\frac{\alpha}{2}\|\cdot\|_2^2}(x) = \frac{1}{\alpha+1}x$.

Example 2: (The ℓ_2 Norm) Let $\bar{B}(x, r) := \{y \in \mathbb{R}^d : \|x - y\|_2 \leq r\}$ denote a closed ball center at $x \in \mathbb{R}^d$ with radius $r > 0$. Then

$$\partial(\|\cdot\|_2)(x) = \begin{cases} \left\{ \frac{x}{\|x\|_2} \right\}, & \text{if } x \neq 0; \\ \bar{B}(0, 1), & \text{if } x = 0, \end{cases}$$

$$(\|\cdot\|_2)^*(x) = \iota_{\bar{B}(0,1)}(x) = \begin{cases} 0, & \text{if } x \in \bar{B}(0, 1); \\ +\infty, & \text{otherwise,} \end{cases}$$

where $\iota_{\bar{B}(0,1)}$ is the indicator function of $\bar{B}(0, 1)$, and

$$\text{prox}_{\alpha\|\cdot\|_2}(x) = \max(\|x\|_2 - \alpha, 0) \frac{x}{\|x\|_2}.$$

Third, we start with some definitions and then present two important theorems for finding minimizers of a composite optimization problem.

The affine hull of a set $S \subseteq \mathbb{R}^d$ is a set in \mathbb{R}^d defined as

$$\text{aff}(S) := \left\{ \sum_{i=1}^k \alpha_i x_i \in \mathbb{R}^d : k > 0, x_i \in S, \alpha_i \in \mathbb{R}, \sum_{i=1}^k \alpha_i = 1 \right\}.$$

The relative interior of a set $S \subseteq \mathbb{R}^d$ is a set in \mathbb{R}^d of all interior points of S relative to $\text{aff}(S)$ defined as

$$\text{ri}(S) := \{x \in S : \exists \epsilon > 0, B(x, \epsilon) \cap \text{aff}(S) \subseteq S\},$$

where $B(x, \epsilon) := \{y \in \mathbb{R}^d : \|x - y\|_2 < \epsilon\}$ is an open ball centered at $x \in \mathbb{R}^d$ with radius $\epsilon > 0$.

Now we are ready to present the chain rule for the subdifferential and the Fermat's rule,

which will be utilized to characterize the solutions of problem (P1).

The chain rule for the subdifferential of a convex function $f : \mathbb{R}^m \rightarrow (-\infty, +\infty]$ composed with a linear operator $A \in \mathbb{R}^{m \times n}$ is stated as follows: If $f \in \Gamma_0(\mathbb{R}^m)$ and $\text{Range}(A) \cap \text{ri}(\text{dom } f) \neq \emptyset$, then

$$\partial(f \circ A)(x) = A^\top \partial f(Ax),$$

for all $x \in \mathbb{R}^n$.

Fermat's rule characterizes the global minimizers of a proper function $f : \mathbb{R}^d \rightarrow (-\infty, +\infty]$ in terms of the subdifferential of f as follows

$$\text{argmin } f = \{x \in \mathbb{R}^d : 0 \in \partial f(x)\}.$$

2.1.2 Preliminaries in fixed point theory

Let us recall some definitions and helpful theorems in fixed point theory for developing iterative algorithms, including Krasnosel'skiĭ–Mann algorithm that will be used in this dissertation.

Definition 2.1 The set of fixed points of an operator $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is defined as

$$\text{Fix } T := \{x \in \mathbb{R}^d : x = Tx\}.$$

Definition 2.2 An operator $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is

- (i) firmly nonexpansive with respect to $H \in \mathbb{S}_+^d$ if for all $x, y \in \mathbb{R}^d$

$$\|Tx - Ty\|_H^2 \leq \langle Tx - Ty, x - y \rangle_H;$$

(ii) nonexpansive with respect to $H \in \mathbb{S}_+^d$ if for all $x, y \in \mathbb{R}^d$

$$\|Tx - Ty\|_H \leq \|x - y\|_H;$$

(iii) α -averaged with respect to $H \in \mathbb{S}_+^d$, where $\alpha \in (0, 1)$, if there exists a nonexpansive operator $R : \mathbb{R}^d \rightarrow \mathbb{R}^d$ with respect to $H \in \mathbb{S}_+^d$ such that $T = (1 - \alpha)\text{Id} + \alpha R$.

Lemma 2.3 [25] Let $f \in \Gamma_0(\mathbb{R}^d)$ and $H \in \mathbb{S}_+^d$. Then $\text{prox}_{f,H}$ is firmly nonexpansive with respect to H . If $H = \frac{1}{\alpha}I_d$, where $\alpha > 0$, then $\text{prox}_{f,H} = \text{prox}_{\alpha f}$ is firmly nonexpansive with respect to the standard ℓ_2 norm.

Lemma 2.4 [25] Let $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ and $H \in \mathbb{S}_+^d$. Then

(i) T is firmly nonexpansive with respect to H if and only if for all $x, y \in \mathbb{R}^d$

$$\|Tx - Ty\|_H^2 \leq \|x - y\|_H^2 - \|(\text{Id} - T)x - (\text{Id} - T)y\|_H^2;$$

(ii) T is α -averaged with respect to H , $\alpha \in (0, 1)$, if and only if for all $x, y \in \mathbb{R}^d$

$$\|Tx - Ty\|_H^2 \leq \|x - y\|_H^2 - \frac{1 - \alpha}{\alpha} \|(\text{Id} - T)x - (\text{Id} - T)y\|_H^2;$$

(iii) T is firmly nonexpansive with respect to H if and only if T is $\frac{1}{2}$ -averaged with respect to H .

It is clear that a firmly nonexpansive operator is $\frac{1}{2}$ -averaged, and an α -averaged operator is nonexpansive. The nonexpansiveness of an operator T is sufficient to develop an iterative algorithm which generates a sequence converging to a point in $\text{Fix } T$, as demonstrated in the following theorem on Krasnosel'skiĭ–Mann algorithm.

Theorem 2.5 (Krasnosel'skii–Mann Algorithm) Let $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ be a nonexpansive operator with respect to $H \in \mathbb{S}_+^d$. Suppose that $\text{Fix } T \neq \emptyset$. Let $\{\lambda_k\}$ be a sequence in $[0, 1]$ such that $\sum_k \lambda_k(1 - \lambda_k) = +\infty$. Then, for any initial vector $x^0 \in \mathbb{R}^d$, the sequence $\{x^k\}$ generated by

$$x^{k+1} = x^k + \lambda_k (Tx^k - x^k)$$

converges to a point in $\text{Fix } T$.

Proof. The result follows from Theorem 5.14 in [25].

□

2.2 Fixed Point Characterization

With the preliminaries on convex analysis and fixed point theory in Section 2.1, we are ready to characterize the solutions of the optimization problem (P1) as the solutions of a system of fixed point equations in terms of proximity operators.

According to Fermat's rule, a vector $x \in \mathbb{R}^n$ is a solution of problem (P1) if and only if the following inclusion relation holds

$$0 \in \partial(f \circ A)(x). \quad (2.6)$$

By applying equation (2.4) to the inclusion relation (2.6), the solution x of problem (P1) can be characterized as a fixed point of the proximity operator of $f \circ A$ with respect to any $P \in \mathbb{S}_+^n$, that is,

$$x = \text{prox}_{f \circ A, P}(x). \quad (2.7)$$

However, it is rare to have an explicit expression of the proximity operator $\text{prox}_{f \circ A, P}$ for the optimization problems arising from image processing. Hence, it becomes necessary to exploit the composition nature of problem (P1) and to take advantage of the function f if

its proximity operator has a closed formula or is easy to evaluate.

Next, we apply the subdifferential chain rule to the composite function $f \circ A$, and obtain that the subdifferential of $f \circ A$ evaluated at a point x is $A^\top \partial f(Ax)$. Then it follows from equations (2.3) that the fixed point equation (2.7) is equivalent to the following system

$$\begin{cases} x = x - P^{-1}A^\top y \\ y \in \partial f(Ax). \end{cases} \quad (2.8)$$

Note that the second equation in equations (2.8) can be converted into a fixed point equation in terms of the proximity operator of f^* by using equation (2.2) and (2.4). Furthermore, the solutions of problem (P1) can be characterized as the solutions of a system of fixed point equations presented in the following proposition.

Proposition 2.6 Suppose that the set of solutions of problem (P1) is nonempty. If a vector $x \in \mathbb{R}^n$ is a solution of problem (P1), then, for any $P \in \mathbb{S}_+^n$ and $Q \in \mathbb{S}_+^m$, there exists a vector $y \in \mathbb{R}^m$ such that the following system of equations holds

$$\begin{cases} x = x - P^{-1}A^\top y \\ y = \text{prox}_{f^*, Q}(y + Q^{-1}Ax). \end{cases} \quad (2.9)$$

Conversely, if there exist $P \in \mathbb{S}_+^n$ and $Q \in \mathbb{S}_+^m$, $x \in \mathbb{R}^n$, and $y \in \mathbb{R}^m$ satisfying the system of equations (2.9), then x is a solution of problem (P1).

Proof. It follows from Fermat's rule and the subdifferential chain rule that $x \in \mathbb{R}^n$ is a solution of problem (P1) if and only if there exists $y \in \mathbb{R}^m$ such that $y \in \partial f(Ax)$ and $A^\top y = 0$.

Suppose $x \in \mathbb{R}^n$ is a solution of problem (P1). Let $P \in \mathbb{S}_+^n$ and $Q \in \mathbb{S}_+^m$. It follows from equation (2.2) that $Q(Q^{-1}Ax) \in \partial f^*(y)$. Thus, we deduce from equation (2.4) that the system of equations (2.9) holds.

Conversely, suppose that there exist $P \in \mathbb{S}_+^n$ and $Q \in \mathbb{S}_+^m$, $x \in \mathbb{R}^n$, and $y \in \mathbb{R}^m$ satisfying the system of equations (2.9). Then $A^\top y = 0$. Also, it follows from equation (2.2) and equation (2.4) that $y \in \partial f(Ax)$. Thus, x is a solution of problem (P1).

□

Proposition 2.6 demonstrates that solving problem (P1) is equivalent to solving the system of fixed point equations (2.9). In fact, the system of equations (2.9) is not the unique system of fixed point equations that can characterize the solutions. Other variants of equations (2.9) can also be used to characterize the solutions as long as the solutions of fixed point equations (2.9) remain unchanged.

We present two variants of the fixed point equations (2.9). The first variant is generated by substituting one of the fixed point equations into the other equations. The second variant is generated by applying Moreau's identity (2.5) to rewrite the proximity operator if additional information on the objective function is provided.

Example 1: The first equation in equations (2.9) is a fixed point equation with respect to x , so we can replace the variable x in the second equation by the first equation with a parameter \tilde{P}^{-1} different from P^{-1} . The new system of fixed point equations is shown as follows

$$\begin{cases} x = x - P^{-1}A^\top y \\ y = \text{prox}_{f^*,Q}(y + Q^{-1}A(x - \tilde{P}^{-1}A^\top y)). \end{cases} \quad (2.10)$$

Example 2: Suppose that $A = I_n$. It follows from the first equation in equations (2.9) that the variable $y = 0$ and that y can be eliminated from the system. Then the first fixed point equation of x can be rewritten in terms of the proximity operator of f instead of f^* , by substituting the second equation into the variable y in the first equation, choosing $P = Q^{-1}$, and then applying the Moreau's identity (2.5). The new system of fixed point equations is shown as follows

$$x = \text{prox}_{f,Q^{-1}}(x). \quad (2.11)$$

All the fixed point equations mentioned above, including equations (2.9), (2.10) and (2.11), can be written as the following unified and compact fixed point equation

$$w = \text{prox}_{F,R}(Ew), \quad (2.12)$$

where $w \in \mathbb{R}^d$, $F : \mathbb{R}^d \rightarrow (-\infty, +\infty]$, $R \in \mathbb{S}_+^d$ and $E \in \mathbb{R}^{d \times d}$.

Equations (2.9) can be written as the unified fixed point equation (2.12) with $F : \mathbb{R}^{n+m} \rightarrow (-\infty, +\infty]$ defined by $F(w) = f^*(y)$, $R = \text{diag}(P, Q) \in \mathbb{S}_+^{n+m}$,

$$w = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{n+m} \quad \text{and} \quad E = \begin{bmatrix} I_n & -P^{-1}A^\top \\ Q^{-1}A & I_m \end{bmatrix} \in \mathbb{R}^{(n+m) \times (n+m)}.$$

Equations (2.10) can be written as equation (2.12) with the same w , F , and R as equations (2.9), but with a different matrix E defined as follows

$$E = \begin{bmatrix} I_n & -P^{-1}A^\top \\ Q^{-1}A & I_m - Q^{-1}A\tilde{P}^{-1}A^\top \end{bmatrix} \in \mathbb{R}^{(n+m) \times (n+m)}. \quad (2.13)$$

Equations (2.11) can be written as equation (2.12) with $w = x \in \mathbb{R}^n$, $F : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ defined by $F(x) = f(x)$, $R = Q^{-1} \in \mathbb{S}_+^n$ and $E = I_n$.

The unified fixed point equation (2.12) plays a fundamental role in the fixed-point proximity framework. It can represent a variety of fixed point equations that characterize the solutions of problem (P1), and allows us to formulate the fixed-point proximity framework in a general setting.

2.3 Fixed-point Proximity Framework

In this section, we present a general framework for developing algorithms to solve problem (P1). The framework is established based on the unified fixed point equation (2.12) in terms

of proximity operators, so we call this framework a fixed-point proximity framework. We start with equation (2.12) and then reformulate this equation to another equivalent fixed point equation, in order to develop convergent iterative algorithms.

The unified fixed point equation (2.12), which characterizes the solutions of problem (P1), is a fixed point equation of the composite operator $\text{prox}_{F,R} \circ E$. The proximity operator $\text{prox}_{F,R}$ is always firmly nonexpansive with respect to R . However, $\|E\|_2$ in most of the scenarios is strictly great than 1 (see, e.g., [16, 51]). The composite operator $\text{prox}_{F,R} \circ E$ may not be nonexpansive, and the simple iterative algorithm $w^{k+1} = \text{prox}_{F,R}(Ew^k)$ may not converge. Thus, the operator $\text{prox}_{F,R} \circ E$ has to be reformulated into another operator in order to generate a convergent algorithm.

In the fixed-point proximity framework, we split up the matrix E in equation (2.12) into two matrices M and $E - M$ and then study the following implicit iterative algorithm

$$w^{k+1} = \text{prox}_{F,R}(Mw^{k+1} + (E - M)w^k), \quad (2.14)$$

where M is a matrix to be determined later.

In order to have a better understanding of the implicit iterative algorithm (2.14), we introduce a new operator \mathcal{L}_M induced from the implicit iterative scheme (2.14).

Definition 2.7 Let M be a $d \times d$ matrix. If for any vector $u \in \mathbb{R}^d$ the following equation

$$v = \text{prox}_{F,R}(Mv + (E - M)u) \quad (2.15)$$

has a unique solution $v \in \mathbb{R}^d$, then $\mathcal{L}_M : \mathbb{R}^d \rightarrow \mathbb{R}^d : u \mapsto v$, induced from equation (2.15), is called an M -operator associated with the operator $\text{prox}_{F,R} \circ E$.

Depending on the choice of the matrix M , the solution of the implicit equation (2.15) may not exist in general. If it exists, it may not be unique. If we impose the existence and uniqueness on the solution of the implicit equation (2.15) for any given vector u , then the

resulting operator \mathcal{L}_M is well-defined and the fixed point equation (2.12) can be reformulated in terms of the new operator \mathcal{L}_M as follows

$$w = \mathcal{L}_M(w). \quad (2.16)$$

This new fixed point equation preserves the characterization of the solutions of problem (P1), because \mathcal{L}_M and $\text{prox}_{F,R} \circ E$ have the same fixed points.

Proposition 2.8 Suppose \mathcal{L}_M is an M -operator associated with the operator $\text{prox}_{F,R} \circ E$. Then the fixed points of $\text{prox}_{F,R} \circ E$ are the same as the fixed points of \mathcal{L}_M , i.e.,

$$\text{Fix}(\text{prox}_{F,R} \circ E) = \text{Fix}(\mathcal{L}_M).$$

Proof. Let $w \in \mathbb{R}^d$. According to Definition 2.7, $w = \mathcal{L}_M(w)$ if and only if

$$w = \text{prox}_{F,R}(Mw + (E - M)w) = \text{prox}_{F,R}(Ew).$$

Thus, the result immediately follows. \square

As the optimization problem (P1) has been transformed to the fixed point problem (2.16), the powerful tools in fixed point theory mentioned in Section 2.1.2 can be applied to develop iterative algorithms for solving problem (P1). In particular, the result in Theorem 2.5 suggests that an algorithm with a Krasnosel'skiĭ–Mann iterative scheme can efficiently find the fixed points of \mathcal{L}_M , which are also the solutions of problem (P1). The sequence $\{w^k\}$ generated by the algorithm is set as follows

$$w^{k+1} = w^k + \lambda_k(\mathcal{L}_M(w^k) - w^k), \quad (2.17)$$

where $\lambda_k > 0$ and $\mathcal{L}_M(w^k)$ is the unique solution of the following implicit equation

$$w = \text{prox}_{F,R}(Mw + (E - M)w^k). \quad (2.18)$$

We call the algorithm with the above iterative scheme as a **Fixed-point Proximity Algorithm**, since it is developed based on fixed point equations in terms of proximity operators. In this algorithm, the operator \mathcal{L}_M is associated with the choice of the matrix M and has a great impact on the overall performance of the algorithm.

To ensure that the sequence generated by the fixed-point proximity algorithm (2.17) converges to a solution of problem (P1), the operator \mathcal{L}_M should satisfy the following three properties.

- **Property 1:** \mathcal{L}_M is an M -operator;
- **Property 2:** \mathcal{L}_M can be evaluated efficiently;
- **Property 3:** \mathcal{L}_M is nonexpansive.

The first property is to guarantee that \mathcal{L}_M is a well-defined single-valued operator and the algorithm, therefore, is well-defined. The second property is required for practical purposes. The computational efficiency of \mathcal{L}_M at each step influences the computational efficiency of the algorithm. Thus, we have to make sure that \mathcal{L}_M can be evaluated to the error tolerance within an acceptable time period. Otherwise, evaluating the operator \mathcal{L}_M , which is induced from an implicit equation, may be as difficult as solving the original fixed point problem (2.12). The third property is a sufficient condition in Theorem 2.5 that yields the convergence of the algorithm. These three properties serve as a guideline for developing a fixed-point proximity algorithm.

In the following sections, we will first present several existing fixed-point proximity algorithms, then use those algorithms as examples to illustrate how Property 1, Property 2 and Property 3 on \mathcal{L}_M can yield a convergent fixed-point proximity algorithm.

2.4 Existing Fixed-point Proximity Algorithms

Many algorithms in the literature for solving problem (P1) can be identified as fixed-point proximity algorithms, even though they are developed from different perspectives. For example, primal dual algorithm (PD), alternating direction method of multipliers (ADMM), alternating split Bregman iteration (ASBI), and Douglas-Rachford splitting algorithm (DRSA) are fixed-point proximity algorithms, and can be derived from the same fixed-point proximity framework just with different choices of the matrix M .

Next, we review one class of existing fixed-point proximity algorithms, which are designed to solve problem (P1) with

$$A = \begin{bmatrix} I_n \\ B \end{bmatrix}, \quad w = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{n+m},$$

and $f : \mathbb{R}^{n+m} \rightarrow (-\infty, +\infty]$ defined by $f(w) = f_1(x) + f_2(y)$, where $f_1 : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ and $f_2 : \mathbb{R}^m \rightarrow (-\infty, +\infty]$ are proper, lower semi-continuous and convex, and B is an $m \times n$ matrix. The corresponding optimization problem is shown as follows

$$\min_{x \in \mathbb{R}^n} f_1(x) + f_2(Bx). \quad (2.19)$$

The algorithms to be presented can be identified as fixed-point proximity algorithms with the same iterative equation as follows. The function F , and matrices E and R , which are derived from the objective function (2.19), are the same for all the algorithms, while the matrix M is different across algorithms.

$$w^{k+1} = \text{prox}_{F,R}(Mw^{k+1} + (E - M)w^k), \quad (2.20)$$

where $F : \mathbb{R}^{n+m} \rightarrow (-\infty, +\infty]$ defined by $F(w) = f_1(x) + f_2^*(y)$, $R = \text{diag}\left(\frac{1}{\alpha}I_n, \frac{1}{\beta}I_m\right)$,

$\alpha > 0, \beta > 0,$

$$w = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{n+m} \quad \text{and} \quad E = \begin{bmatrix} I_n & -\alpha B^\top \\ \beta B & I_m \end{bmatrix}.$$

Note that F is a block separable sum of two convex functions, so w is a vector of two variables, E , R and M are 2×2 block matrices, and Algorithm (2.20) consists of two iterative equations.

For each of the following fixed-point proximity algorithms, we present its choice of M , and then analyze its corresponding operator \mathcal{L}_M under the fixed-point proximity framework, regarding Property 1, Property 2, and Property 3 discussed in the previous section.

- *Primal Dual Algorithm (PD)* [11,82]: This algorithm is developed from the perspective of the Fenchel-Rockafellar duality theory.

$$\begin{cases} x^{k+1} = \text{prox}_{\alpha f_1}(x^k - \alpha B^\top y^k) \\ y^{k+1} = \text{prox}_{\beta f_2^*}(2\beta B x^{k+1} - \beta B x^k + y^k). \end{cases} \quad (2.21)$$

It can be identified as a fixed-point proximity algorithm of the form as equation (2.20) with

$$M = \begin{bmatrix} 0 & 0 \\ 2\beta B & 0 \end{bmatrix}. \quad (2.22)$$

This algorithm converges if $\alpha\beta\|B\|_2^2 < 1$.

In PD (2.21), x^{k+1} can be computed explicitly, then y^{k+1} can also be computed explicitly by using the newest update x^{k+1} . Thus, the operator \mathcal{L}_M associated with the matrix M defined as (2.22) has an explicit expression and satisfies Property 1 and Property 2. If the convergence assumption $\alpha\beta\|B\|_2^2 < 1$ is satisfied, then \mathcal{L}_M is firmly nonexpansive, which implies Property 3.

- *Fixed Point Algorithm Based on the Proximity Operator for ROF model (FP²O)* [60]:

This algorithm is the first fixed-point proximity algorithm that is developed from the

perspective of fixed point equations in terms of proximity operators.

$$\begin{cases} x^{k+1} = \text{prox}_{f_1}(x^{k+1} - B^\top y^k) \\ y^{k+1} = \text{prox}_{\beta f_2^*}(\beta B x^{k+1} + y^k), \end{cases} \quad (2.23)$$

where $f_1 = \frac{1}{2}\|\cdot - z\|_2^2$.

It can be identified as a fixed-point proximity algorithm of the form as equation (2.20) with $\alpha = 1$ and

$$M = \begin{bmatrix} I & 0 \\ \beta B & 0 \end{bmatrix}. \quad (2.24)$$

This algorithm converges if $\|I_m - \beta B B^\top\|_2 < 1$.

In FP²O (2.23), x^{k+1} is the solution of an implicit equation and y^{k+1} can be computed explicitly by using the newest update x^{k+1} . FP²O is designed to solve the optimization problem (2.19) with $f_1 = \frac{1}{2}\|\cdot - z\|_2^2$, whose proximity operator is $\text{prox}_{f_1}(x) = \frac{1}{2}x + \frac{1}{2}z$. Thus, the first equation has a closed form, that is, $x^{k+1} = z - B^\top y^k$. Then the second equation can further be rewritten as

$$y^{k+1} = \text{prox}_{\beta f_2^*}(\beta B z + (I_m - \beta B B^\top)y^k),$$

and y^{k+1} can be computed explicitly. If the convergence assumption $\|I_m - \beta B B^\top\|_2 \leq 1$ is satisfied, then the operator \mathcal{L}_M associated with the matrix M defined as (2.24) has an explicit expression and satisfies Property 1, Property 2, and Property 3.

- *Alternating Split Bregman Iteration (ASBI)* [39] : This algorithm is developed from the perspective of Bregman splitting. ASBI is equivalent to ADMM [32], while ADMM

is closely related to Douglas-Rachford splitting algorithm (DRSA) [30, 35, 55].

$$\begin{cases} x^{k+1} = \text{prox}_{f_1} ((I_n - \beta B^\top B)x^{k+1} - 2B^\top y^{k+1} + \beta B^\top Bx^k + B^\top y^k) \\ y^{k+1} = \text{prox}_{\beta f_2^*} (\beta Bx^k + y^k), \end{cases} \quad (2.25)$$

where $f_1 = \frac{1}{2} \|\cdot - z\|_2^2$.

It can be identified as a fixed-point proximity algorithm of the form as equation (2.20)

with $\alpha = 1$ and

$$M = \begin{bmatrix} I_n - \beta B^\top B & -2B^\top \\ 0 & 0 \end{bmatrix}. \quad (2.26)$$

The sequence $\{y^k\}$ converges for any $\beta > 0$, but the sequence $\{x^k\}$ may not converge, see [51].

In ASBI (2.25), y^{k+1} can be computed explicitly, but x^{k+1} has to be computed from an implicit equation even after substituting the newest update y^{k+1} . It is mentioned in [39] that this implicit step is evaluated by using Gauss-Seidel method if $f_1 = \frac{1}{2} \|\cdot - z\|_2^2$. Thus, the operator \mathcal{L}_M associated with the matrix M defined as (2.26) has an implicit expression but can be solved explicitly. The operator \mathcal{L}_M satisfies Property 1 and Property 2. However, the convergence assumption $\beta > 0$ cannot guarantee that \mathcal{L}_M is firmly nonexpansive if $B^\top B$ is not a full-rank matrix [51]. Hence, Property 3 may not be achieved for ASBI.

From the review of existing fixed-point proximity algorithms, we notice that the block structure of M is crucial to the operator \mathcal{L}_M , and observe two types of block structures of M . In the first type, M is a strictly block lower or upper triangular matrix, which is observed in PD (2.21). In this case, $\mathcal{L}_M(w^k)$ has a closed form, so the algorithm has an explicit expression and can be computed accurately. In the second type, M has a matrix structure with at least one nonzero diagonal block, which is observed in FP²O (2.23) and ASBI (2.25). In this case, $\mathcal{L}_M(w^k)$ has an implicit expression but can be computed explicitly

provided that additional constraints on the objective function are satisfied.

(i) **Strictly block lower or upper triangular matrix structure**

Two examples of strictly block lower or upper triangular matrix structures on the matrix M are presented as follows

$$M = \begin{bmatrix} & \\ \times & \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \times & \\ \times & \times \end{bmatrix}.$$

If the matrix M has a strictly triangular block structure, then its corresponding operator \mathcal{L}_M automatically satisfies Property 1 and Property 2. Because at least one of the variable components can be computed explicitly, and then it can be utilized to compute other variable components, which results in an explicit operator \mathcal{L}_M .

In PD (2.21), the block structure of M defined as (2.22) can be identified as the first case above; in the proximity algorithm with the Gauss-Seidel scheme (3.13) [16] that will be discussed in Chapter 3, the block structure of M defined as (3.14) can be identified as the second case above. However, from the perspective of algorithm design, only a limited number of algorithms can have such structure of M . Also, in order to maintain such strictly triangular block structure of M , the convergence assumption may be strict.

(ii) **Matrix structure with at least one nonzero diagonal block**

Two examples of matrix structures with at least one nonzero diagonal block on the matrix M are presented as follows

$$M = \begin{bmatrix} \times & \\ \times & \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \times & \times \\ & \end{bmatrix}.$$

In FP²O (2.23), the block structure of M defined as (2.24) can be identified as the first case; in ASBI (2.25), the block structure of M defined as (2.26) can be identified as the second case.

If the matrix M has at least one nonzero diagonal block, then its corresponding operator \mathcal{L}_M has an implicit expression. The operator \mathcal{L}_M can still be computed explicitly and satisfy Property 1 and Property 2, if the diagonal blocks are well selected and the proximity operator of F has a simple form. For example, in both ASBI and FP²O, the optimization problem contains a quadratic function. Then solving the corresponding implicit equation is equivalent to solving a system of linear equations, which can be efficiently computed. However, if the proximity operator of F does not have a simple form, then those algorithms may fail to work.

All in all, the algorithms mentioned above can be computed explicitly and their corresponding operators \mathcal{L}_M satisfy Property 1 and Property 2. However, there are some issues with existing fixed-point proximity algorithms. For instance, some algorithms have relatively strict convergence assumptions due to the block structure of M , while others are applicable only to certain types of optimization problems.

Therefore, the issues occurred in existing fixed-point proximity algorithms mentioned above motivate us to study implicit fixed-point proximity algorithms whose \mathcal{L}_M has a fully implicit expression. Such algorithms can solve a wide range of optimization problems, have reasonable convergence assumptions, and, more importantly, converge to the solution faster than existing fixed-point proximity algorithms.

2.5 Implicit Fixed-point Proximity Framework

In this section, we aim to develop fixed-point proximity algorithms with fully implicit schemes and establish the implicit fixed-point proximity framework for those implicit algorithms. The study on the implicit fixed-point proximity framework is the main contribution

of this dissertation and will be illustrated in detail with sufficient theoretical results.

Recall that fixed-point proximity algorithms share a common iterative scheme (2.17), but differ in \mathcal{L}_M induced from the implicit equation (2.18). Depending on the choice of the matrix M , \mathcal{L}_M may be evaluated explicitly or implicitly.

In the existing fixed-point proximity algorithms, $\mathcal{L}_M(w^k)$ is computed explicitly either directly from a closed form or by some numerical methods. In order to maintain the explicit expression, those existing algorithms suffer from some issues. For example, primal dual algorithm (2.21) has to assume a relatively strict convergence condition and ASBI (2.25) can only solve certain types of optimization problems.

Therefore, we focus on the case when \mathcal{L}_M has a fully implicit expression and $\mathcal{L}_M(w^k)$ is computed implicitly. Because the implicit schemes allow more flexibility while building the structures of the algorithms, and have a potential to yield an algorithm that outperforms existing explicit algorithms in terms of computational time.

Note that we refer the case when \mathcal{L}_M has a fully implicit expression to the case when \mathcal{L}_M has an implicit expression and is not evaluated explicitly. This case should be distinguished from the case when \mathcal{L}_M is written in an implicit expression but can be evaluated explicitly.

In the following, we will establish a novel framework for developing algorithms with fully implicit schemes. To the best of our knowledge, the implicit fixed point proximity framework to be presented is the first framework to address the operator \mathcal{L}_M with a fully implicit expression. And the existing fixed-point proximity framework is designed for developing explicit algorithms and does not address the issues that may occur in implicit algorithms.

2.5.1 Preliminaries on contractive mappings

In order to tackle the issues that we may face while developing implicit algorithms, we employ some powerful tools in fixed point theory. They are contractive mappings and Banach fixed point theorem.

Definition 2.9 A mapping $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is contractive with respect to $H \in \mathbb{S}_+^d$ if there exists a contraction constant $q \in [0, 1)$ such that for all $x, y \in \mathbb{R}^d$

$$\|Tx - Ty\|_H \leq q\|x - y\|_H.$$

It is guaranteed by Banach fixed point theorem that the fixed point of a contractive mapping exists and is unique. Furthermore, this unique fixed point can be achieved via a simple iterative scheme.

Theorem 2.10 (Banach Fixed Point Theorem) [1] Suppose that $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is contractive with respect to $H \in \mathbb{S}_+^d$. Then T has a unique fixed point x^* in \mathbb{R}^d , i.e., $x^* = Tx^*$. Furthermore, for any initial vector $x^0 \in \mathbb{R}^d$, the sequence $\{x^k\}$ generated by

$$x^{k+1} = Tx^k$$

converges to the unique fixed point x^* .

Now, we are ready to develop an implicit fixed-point proximity algorithm by constructing contractive mappings to evaluate the operator \mathcal{L}_M with a fully implicit expression.

2.5.2 Implicit fixed-point proximity algorithm

Under the general fixed-point proximity framework, a fixed-point proximity algorithm has the iterative scheme as equation (2.17). For the operator \mathcal{L}_M not having an explicit expression, $\mathcal{L}_M(w^k)$ is computed iteratively by solving the implicit equation (2.18), that is,

$$w = \text{prox}_{F,R}(Mw + (E - M)w^k). \quad (2.18)$$

The proposed **Implicit Fixed-Point Proximity Algorithm** is presented in Algorithm 1.

Algorithm 1 Implicit Fixed-point Proximity Algorithm for problem (P1)

```

1: Choose  $w^0 \in \mathbb{R}^d$ ,  $\lambda_k \in [0, 1]$ 

2: for  $k$  from 1 to  $K$  do

3:   Compute  $\mathcal{L}_M(w^k)$  via the inner loop:

4:   Set  $l = 0$ , choose  $w_0^{k+1} \in \mathbb{R}^d$ 

5:   repeat

6:      $w_{l+1}^{k+1} = \text{prox}_{F,R}(Mw_l^{k+1} + (E - M)w^k)$  ▷ Inner step

7:      $l \leftarrow l + 1$ 

8:   until stopping criterion is satisfied

9:    $\mathcal{L}_M(w^k) = w_\infty^{k+1}$  ▷  $w_\infty^{k+1}$  is the output from the inner loop

10:   $w^{k+1} = w^k + \lambda_k(\mathcal{L}_M(w^k) - w^k)$ 

11: end for

```

No matter whether \mathcal{L}_M has an explicit expression or an implicit expression, \mathcal{L}_M should satisfy the following three properties, in order to ensure that the corresponding fixed-point proximity algorithm converges.

- **Property 1:** \mathcal{L}_M is an M -operator;
- **Property 2:** \mathcal{L}_M can be evaluated efficiently;
- **Property 3:** \mathcal{L}_M is nonexpansive.

Therefore, Algorithm 1 needs to satisfy the following assumptions so that its corresponding \mathcal{L}_M with a fully implicit expression satisfies Property 1, Property 2, and Property 3.

Assumption (A1.1) $\|M\|_R < 1$;

Assumption (A1.2) $R(E - I)$ is skew-symmetric;

Assumption (A1.3) $R(E - M) \in \mathbb{S}_+^d$.

In the following subsections, we demonstrate that the operator \mathcal{L}_M of Algorithm 1 satisfies Property 1, Property 2, and Property 3, if these three assumptions are satisfied. In particular, Assumption (A1.1) implies that the operator \mathcal{L}_M satisfies Property 1 and Property 2. Assumption (A1.2) and (A1.3) guarantee that \mathcal{L}_M satisfies Property 3. Besides discussing the three properties on \mathcal{L}_M , we also conduct a convergence analysis on Algorithm 1 by taking the errors from inner iterations into consideration.

2.5.3 Property 1: \mathcal{L}_M is an M -operator

Property 1 can ensure that the operator \mathcal{L}_M of Algorithm 1 is well-defined and further guarantee that Algorithm 1 is also well-defined. Thus, it is essential to include Property 1 in the assumptions of Algorithm 1.

The operator \mathcal{L}_M of Algorithm 1 has a fully implicit expression, and \mathcal{L}_M evaluated at w^k , denoted as $\mathcal{L}_M(w^k)$, is induced from the implicit equation (2.18). By Definition 2.7, \mathcal{L}_M being an M -operator associated with $\text{prox}_{F,R} \circ E$ indicates that, for any vector w^k , the solution of the implicit equation (2.18) exists and is unique. To demonstrate that Assumption (A1.1) implies Property 1 on the operator \mathcal{L}_M of Algorithm 1, we need to have a better understanding of the implicit equation (2.18).

First, we recognize the implicit equation (2.18) for a given vector w^k as the fixed point equation $w = \mathcal{T}(w)$, where $\mathcal{T} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is defined as

$$\mathcal{T}(w) := \text{prox}_{F,R}(Mw + (E - M)w^k). \quad (2.27)$$

Thus, \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$ if and only if for any vector w^k the fixed point of \mathcal{T} exists and is unique.

Second, we show that Assumption (A1.1) $\|M\|_R < 1$ implies the existence and uniqueness of the fixed point of \mathcal{T} , and that therefore Property 1 is achieved.

According to Banach fixed point theorem in Theorem 2.10, if \mathcal{T} is a contractive mapping, then the fixed point of \mathcal{T} exists and is unique. And there exists a feasible iterative method to achieve the fixed point.

Algorithm 1 exactly follows this idea. The assumption $\|M\|_R < 1$ ensures that \mathcal{T} is a contractive mapping, and further guarantees Property 1. Also, to solve the fixed point problem of \mathcal{T} , Algorithm 1 conducts inner iterations and generates an inner sequence $\{w_l^{k+1}\}_l$ by the 6th line in Algorithm 1, i.e.,

$$w_{l+1}^{k+1} = \text{prox}_{F,R}(Mw_l^{k+1} + (E - M)w^k). \quad (2.28)$$

This inner step can be viewed as $w_{l+1}^{k+1} = \mathcal{T}(w_l^{k+1})$. Since \mathcal{T} is a contractive mapping, then the inner sequence $\{w_l^{k+1}\}_l$ converges to $\mathcal{L}_M(w^k)$, which is the unique fixed point of \mathcal{T} as well as the unique solution of equation (2.18). Therefore, we obtain the following proposition.

Proposition 2.11 Let M be a $d \times d$ matrix such that $\|M\|_R < 1$. Then \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$. Furthermore, for any vector $w^k \in \mathbb{R}^d$, the sequence $\{w_l^{k+1}\}_l$ generated by equation (2.28), given any initial vector $w_0^{k+1} \in \mathbb{R}^d$, converges to $\mathcal{L}_M(w^k)$.

Proof. Let \mathcal{T} be the operator defined in equation (2.27). Since $\text{prox}_{F,R}$ is firmly nonexpansive with respect to R and $\|M\|_R < 1$, then it follows from Definition 2.9 that \mathcal{T} for any vector $w^k \in \mathbb{R}^d$ defined in (2.27) is contractive with contraction constant $\|M\|_R$ with respect to R . Thus, it follows from Theorem 2.10 that the fixed point of \mathcal{T} is unique and $\{w_l^{k+1}\}_l$, generated by $w_{l+1}^{k+1} = \mathcal{T}(w_l^{k+1})$, i.e., equation (2.28), converges to the unique fixed point of \mathcal{T} . Therefore, \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$ and $\{w_l^{k+1}\}_l$ converges to the unique solution $\mathcal{L}_M(w^k)$. □

In fact, the assumption $\|M\|_R < 1$ can be relaxed, depending on the block structure of

M . The fixed point problem of \mathcal{T} mentioned above may be reformulated to another fixed point problem, where contractive mappings in the inner loops can be constructed with a weaker condition than $\|M\|_R < 1$. We will discuss such assumption case by case in the examples of implicit fixed-point proximity algorithms in Chapter 3.

2.5.4 Property 2: \mathcal{L}_M can be evaluated efficiently

Including Property 2 in the convergence assumptions of Algorithm 1 is of our practical interest, because the performance of Algorithm 1 is related to the computational efficiency of \mathcal{L}_M .

Next, we demonstrate that Algorithm 1 can evaluate the operator \mathcal{L}_M efficiently if Assumption (A1.1) is satisfied.

Algorithm 1 computes $\mathcal{L}_M(w^k)$ in equation (2.17) by inner iterations via contractive mappings. It is shown in Proposition 2.11 that the inner sequence $\{w_l^{k+1}\}_l$, generated by equation (2.28) in the inner loops of Algorithm 1, converges to the unique solution $\mathcal{L}_M(w^k)$. Moreover, we show in the following proposition that $\{w_l^{k+1}\}_l$ converges with a geometric convergence rate.

Proposition 2.12 Let M be a $d \times d$ matrix such that $\|M\|_R = q \in [0, 1)$. Then, for any $w^k \in \mathbb{R}^d$, if the sequence $\{w_l^{k+1}\}_l$ is generated by equation (2.28), given any initial vector $w_0^{k+1} \in \mathbb{R}^d$, then we have

$$\|w_{l+1}^{k+1} - w_\infty^{k+1}\|_R \leq \frac{q^l}{1-q} \|w_1^{k+1} - w_0^{k+1}\|_R,$$

where w_∞^{k+1} is the unique solution of $\mathcal{L}_M(w^k)$.

Proof. $\|M\|_R = q \in [0, 1)$ implies that \mathcal{T} is a contractive mapping with the contraction constant q . Thus, the result immediately follows from [40]. \square

One major concern about implicit algorithms with inner iterations is that the computa-

tional cost may increase dramatically compared to explicit algorithms, in order to maintain the accuracy of the inner solution. However, the inner sequence $\{w_l^{k+1}\}_l$, generated by contractive mappings, has a geometric convergence rate. And the inner sequence converges faster as the contraction constant $\|M\|_R$ gets smaller. Therefore, only few iterations in the inner loop are needed to achieve the inner solution with high precision, and the operator \mathcal{L}_M of Algorithm 1 satisfies Property 2.

2.5.5 Property 3: \mathcal{L}_M is nonexpansive

The nonexpansiveness mentioned in Property 3 is one of the convergence assumptions required in Theorem 2.5 for Krasnosel'skii–Mann algorithm. As fixed-point proximity algorithms generated by equation (2.17) have Krasnosel'skii–Mann iterative schemes, the convergence assumptions of Algorithm 1 should include Property 3.

First, we prove that Assumption (A1.2) and (A1.3) can imply that the operator \mathcal{L}_M of Algorithm 1 is firmly nonexpansive with respect to R and further guarantee that Property 3 is satisfied.

Suppose \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$ as defined in Definition 2.7. Then \mathcal{L}_M , a reformulation of $\text{prox}_{F,R} \circ E$, can preserve the firm nonexpansiveness of $\text{prox}_{F,R}$ if $R(E - M) \in \mathbb{S}_+^d$ and $R(E - I)$ is skew-symmetric.

Lemma 2.13 Let \mathcal{L}_M be an M -operator associated with $\text{prox}_{F,R} \circ E$. Suppose $u_i \in \mathbb{R}^d$, and $v_i = \mathcal{L}_M(u_i)$, $i = 1, 2$. Then

$$\langle v_1 - v_2, R(I - M)(v_1 - v_2) \rangle \leq \langle v_1 - v_2, R(E - M)(u_1 - u_2) \rangle.$$

Proof. By the definition of \mathcal{L}_M in Definition 2.7,

$$v_i = \text{prox}_{F,R}(Mv_i + (E - M)u_i).$$

It follows from Definition 2.2 and Lemma 2.3 that $\text{prox}_{F,R}$ is firmly nonexpansive with respect to R , i.e.,

$$\langle v_1 - v_2, R(v_1 - v_2) \rangle \leq \langle v_1 - v_2, RM(v_1 - v_2) + R(E - M)(u_1 - u_2) \rangle.$$

Thus, after combining $\langle v_1 - v_2, R(v_1 - v_2) \rangle$ on the left-hand side with $\langle v_1 - v_2, RM(v_1 - v_2) \rangle$ on the right-hand side, the result immediately follows. \square

Proposition 2.14 Let \mathcal{L}_M be an M -operator associated with $\text{prox}_{F,R} \circ E$. Suppose $R(E - I) \in \mathbb{R}^{d \times d}$ is a skew-symmetric matrix and

$$R(E - M) \in \mathbb{S}_+^d.$$

Then \mathcal{L}_M is firmly nonexpansive with respect to $R(E - M)$.

Proof. Suppose $u_i \in \mathbb{R}^d$ and $v_i = \mathcal{L}_M(u_i)$, $i = 1, 2$. If $R(E - I)$ is a skew-symmetric matrix, then $\langle v_1 - v_2, R(E - I)(v_1 - v_2) \rangle = 0$. Thus,

$$\begin{aligned} \langle v_1 - v_2, R(I - M)(v_1 - v_2) \rangle &= \langle v_1 - v_2, R(I - M)(v_1 - v_2) \rangle + \langle v_1 - v_2, R(E - I)(v_1 - v_2) \rangle \\ &= \langle v_1 - v_2, R(E - M)(v_1 - v_2) \rangle, \end{aligned}$$

and Lemma 2.13 implies

$$\langle v_1 - v_2, R(E - M)(v_1 - v_2) \rangle \leq \langle v_1 - v_2, R(E - M)(u_1 - u_2) \rangle.$$

If $R(E - M) \in \mathbb{S}_+^d$, then $\langle \cdot, \cdot \rangle_{R(E-M)}$ is an inner product and $\| \cdot \|_{R(E-M)}$ is a norm. Therefore,

$$\|v_1 - v_2\|_{R(E-M)}^2 \leq \langle v_1 - v_2, u_1 - u_2 \rangle_{R(E-M)}$$

and \mathcal{L}_M is firmly nonexpansive with respect to $R(E - M)$.

□

Firm nonexpansiveness implies nonexpansiveness, so Proposition 2.14 demonstrates that Property 3 on the operator \mathcal{L}_M of Algorithm 1 can be achieved if $R(E - M) \in \mathbb{S}_+^d$ and $R(E - I)$ is a skew-symmetric matrix.

The two block matrices $R(E - I)$ and $R(E - M)$ mentioned in the assumptions can be computed via block matrix calculation. It is easy to verify whether $R(E - I)$ is skew-symmetric or not, while it requires extra calculation to verify symmetric positive definite matrices.

Second, we present a lemma for symmetric positive definite matrices, which serves as an efficient tool for verifying the assumption $R(E - M) \in \mathbb{S}_+^d$.

Lemma 2.15 If

$$H = \begin{bmatrix} A & B^\top \\ B & C \end{bmatrix} \in \mathbb{R}^{n+m}$$

is a symmetric matrix, and $C \in \mathbb{S}_+^m$, then $H \in \mathbb{S}_+^{n+m}$ if and only if $A - B^\top C^{-1} B \in \mathbb{S}_+^n$.

Proof.

$$\begin{bmatrix} I_n & -B^\top C^{-1} \\ 0 & I_m \end{bmatrix} \begin{bmatrix} A & B^\top \\ B & C \end{bmatrix} \begin{bmatrix} I_n & 0 \\ -C^{-1} B & I_m \end{bmatrix} = \begin{bmatrix} A - B^\top C^{-1} B & 0 \\ 0 & C \end{bmatrix}$$

Hence, $H \in \mathbb{S}_+^{n+m}$ if and only if $A - B^\top C^{-1} B \in \mathbb{S}_+^n$.

□

If $R(E - M)$ is a 2×2 block matrix, then its symmetric positive definiteness can be verified directly by applying Lemma 2.15. If $R(E - M)$ has more blocks, then it can still be viewed as a 2×2 block matrix and we can apply Lemma 2.15 more than once to verify its symmetric positive definiteness.

Last, we shall mention that Proposition 2.14 may not be applicable to verify Property 3 if the assumptions in Proposition 2.14 cannot be achieved.

There are two assumptions in Proposition 2.14. One is that $R(E - I)$ is skew-symmetric and the other is that $R(E - M) \in \mathbb{S}_+^d$. The matrices E and R are determined by the optimization problem (P1) as well as the fixed point problem (2.12) under the framework. In some variants of the fixed point problem, the corresponding matrix $R(E - I)$ is not skew-symmetric. For example, for the matrix E in (2.13), the corresponding matrix $R(E - I)$ is

$$R(E - I) = \begin{bmatrix} -A^\top & \\ A & -A\tilde{P}^{-1}A^\top \end{bmatrix} \in \mathbb{R}^{(n+m) \times (n+m)}.$$

This matrix is not skew-symmetric unless $\tilde{P}^{-1} = 0$. So we cannot apply Proposition 2.14 to check Property 3 for the case when $\tilde{P}^{-1} \in \mathbb{S}_+^n$.

If Proposition 2.14 is not applicable to verify Property 3, we have to carefully construct the matrix M and prove that its corresponding operator \mathcal{L}_M is nonexpansive by using the definition of nonexpansive operators in Definition 2.2.

All in all, Property 3 can be verified by either applying Proposition 2.14 or utilizing the definition of nonexpansive operators to prove it.

2.5.6 Possible block structures of M for implicit algorithms

Based on the previous results, theoretically, it is possible to develop feasible and efficient implicit fixed-point proximity algorithms if Assumption (A1.1), (A1.2) and (A1.3) are satisfied and there is no restriction on the block structure of M . However, practically, the block structure of M plays an important role in the performance of implicit algorithms. Depending on the block structure of M , Algorithm 1 can be improved and the convergence assumptions can be also be relaxed.

Next, we present two proposed block structures for M whose corresponding operator \mathcal{L}_M can be computed efficiently by inner iterations via contractive mappings.

(i) *Skew diagonal block matrix with zero diagonal blocks*

$$M = \begin{bmatrix} & \times \\ \times & \end{bmatrix}$$

(ii) *Matrix with special structure with zero diagonal blocks*

$$M = \begin{bmatrix} & \times & \\ \times & & \\ \times & \times & \end{bmatrix}$$

We will propose several implicit fixed-point proximity algorithms with the above block structures of M in Chapter 3.

2.5.7 Convergence analysis

In the following, we conduct a convergence analysis on the implicit fixed-point proximity algorithm stated in Algorithm 1. In particular, we prove that the sequence generated by Algorithm 1 converges with a rate of $O(\frac{1}{K})$, where K is the number of outer iterations.

Algorithm 1 is developed from the perspective of the fixed-point proximity equation and is specifically designed for those algorithms in which the operator \mathcal{L}_M has a fully implicit scheme. The operator \mathcal{L}_M is evaluated by performing inner iterations via contractive mappings. By taking into consideration the errors caused by the inner loops, the sequence $\{w^k\}$ generated by Algorithm 1 can be viewed as a sequence generated by the following inexact Krasnosel'skiĭ–Mann iteration

$$w^{k+1} = w^k + \lambda_k (\mathcal{L}_M(w^k) + e_k - w^k), \quad (2.29)$$

where $e_k \in \mathbb{R}^d$ represents the computational errors caused by computing \mathcal{L}_M approximately.

Theorem 2.5 for exact Krasnosel'skiĭ–Mann algorithms is not sufficient to guarantee the convergence of an implicit fixed-point proximity algorithm, and the convergence results for existing fixed-point proximity algorithms are also not applicable to implicit algorithms.

Therefore, we utilize the definitions and techniques introduced in [23, 25] to complete the proof for the convergence of Algorithm 1.

First, let us introduce the definition of quasi-Fejér monotonicity and some convergence results in [23]. The quasi-Fejér monotonicity is an extension of Fejér monotonicity, which can address the errors e_k from inner iterations.

Definition 2.16 A sequence $\{x^k\}$ in \mathbb{R}^d is said to be a quasi-Fejér monotone sequence relative to a target set $S \subseteq \mathbb{R}^d$ if there exists $\varepsilon_k \geq 0$ such that $\sum \varepsilon_k < \infty$ and for all $x \in S$,

$$\|x^{k+1} - x\| \leq \|x^k - x\| + \varepsilon_k.$$

Note that this definition of the quasi-Fejér monotonicity refers to the quasi-Fejér monotonicity of Type I in [23]. Also, if $\varepsilon_k = 0$ for all $k \in \mathbb{N}$ then quasi-Fejér monotone reduces to Fejér monotone.

Lemma 2.17 [23] Let $C \in (0, 1]$, let $\{\alpha_k\}$ be a sequence in $(0, +\infty)$, let $\{\beta_k\}$ be a sequence in $(0, +\infty)$, and let $\{\varepsilon_k\}$ be a summable sequence in $(0, +\infty)$ such that

$$\alpha_{k+1} \leq C\alpha_k - \beta_k + \varepsilon_k.$$

Then $\{\alpha_k\}$ converges and $\sum_k |\beta_k| < \infty$.

Theorem 2.18 Suppose $\{x^k\}$ is a quasi-Fejér monotone sequence relative to a nonempty set S in \mathbb{R}^d . Let $\mathfrak{C}\{x^k\}$ denote the set of cluster points of $\{x^k\}$. Then $\{x^k\}$ converges to a point in S if and only if $\mathfrak{C}\{x^k\} \subset S$.

Proof. The result follows from Proposition 3.2 and Theorem 3.8 in [23].

□

If the sequence $\{w^k\}$ generated by Algorithm 1 is quasi-Fejér monotone with respect to the set of solutions of problem (P1), then the convergence of $\{w^k\}$ to a solution of problem (P1)

can be achieved by verifying the sufficient and necessary condition mentioned in Theorem 2.18. That is to verify if every convergent subsequence of $\{w^k\}$ converges to a solution of problem (P1).

Second, let us introduce the demiclosed principle for nonexpansive operators, in order to study the convergence behavior of each convergent subsequence of $\{w^k\}$.

Theorem 2.19 (Demiclosed Principle) [38] Let $T : \mathbb{R}^d \rightarrow \mathbb{R}^d$ be a nonexpansive operator with respect to $H \in \mathbb{S}_+^d$, let $\{x^k\}$ be a sequence in \mathbb{R}^d , and let x be a point in \mathbb{R}^d . Suppose that $\{x^k\}$ converges to x and that $\{x^k - Tx^k\}$ converges to 0. Then $x \in \text{Fix } T$.

Now, with the results above, we are ready to prove the convergence of the sequence $\{w^k\}$ generated by equation (2.29). The proof is presented in three steps. First, we show that $\{w^k\}$ is a quasi-Fejér monotone sequence relative to the solution set, which is $\text{Fix}(\text{prox}_{F,R} \circ E) = \text{Fix}(\mathcal{L}_M)$. Second, we prove that the assumptions of the demiclosed principle are satisfied, and that therefore the limit of $\{w^k\}$ is exactly a fixed point of \mathcal{L}_M . Last, by applying both Theorem 2.18 and Theorem 2.19, the convergence of $\{w^k\}$ is eventually achieved.

Note that the proof of the following theorem is adapted from the proof in [23, 24] so that it is more applicable to implicit fixed-point proximity algorithms. And this proving strategy can be utilized for implicit fixed-point proximity algorithms with uncommon structures.

Theorem 2.20 Assume that $\text{Fix}(\text{prox}_{F,R} \circ E) \neq \emptyset$. Let \mathcal{L}_M be an M -operator associated with $\text{prox}_{F,R} \circ E$ and \mathcal{L}_M is nonexpansive with respect to $H \in \mathbb{S}_+^d$. Let $\{\lambda_k\}$ be a sequence in $[0, 1]$ and $\{e_k\}$ be a sequence in \mathbb{R}^d such that $\sum_k \lambda_k(1 - \lambda_k) = +\infty$ and $\sum_k \lambda_k \|e_k\|_H < \infty$. Suppose the sequence $\{w^k\}$ is generated by equation (2.29) with an initial vector $w^0 \in \mathbb{R}^d$. Then the following statements hold.

- (i) $\{w^k\}$ is a quasi-Fejér monotone sequence relative to $\text{Fix}(\text{prox}_{F,R} \circ E)$;
- (ii) $\{\mathcal{L}_M w^k - w^k\}$ converges to 0;
- (iii) $\{w^k\}$ converges to a point in $\text{Fix}(\text{prox}_{F,R} \circ E)$.

Proof. Let $w^* \in \text{Fix}(\text{prox}_{\Phi, R} \circ E)$, then by Proposition 2.8, $w^* = \text{prox}_{F, R}(Ew^*)$ and $w^* = \mathcal{L}_M(w^*)$.

(i) It follows from the nonexpansive property of \mathcal{L}_M that

$$\begin{aligned} \|w^{k+1} - w^*\|_H &= \|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) + \lambda_k e_k - w^*\|_H \\ &\leq \lambda_k \|\mathcal{L}_M(w^k) - w^*\|_H + \lambda_k \|e_k\|_H + (1 - \lambda_k) \|w^k - w^*\|_H \\ &= \lambda_k \|\mathcal{L}_M(w^k) - \mathcal{L}_M(w^*)\|_H + \lambda_k \|e_k\|_H + (1 - \lambda_k) \|w^k - w^*\|_H \\ &\leq \|w^k - w^*\|_H + \lambda_k \|e_k\|_H. \end{aligned}$$

Hence, $\{w^k\}$ is a quasi-Fejér monotone sequence relative to $\text{Fix}(\text{prox}_{\Phi, R} \circ E)$.

(ii) Since $(1 - \lambda_k)\text{Id} + \lambda_k \mathcal{L}_M$ is λ_k -average, then by Lemma 2.4

$$\|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) - w^*\|_H^2 \leq \|w^k - w^*\|_H^2 - \frac{1 - \lambda_k}{\lambda_k} \|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) - w^k\|_H^2.$$

Let $C = \sup_k \{2 \|w^k - w^*\|_H + \|e_k\|_H\} < +\infty$. Then we have

$$\begin{aligned} \|w^{k+1} - w^*\|_H^2 &= \|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) + \lambda_k e_k - w^*\|_H^2 \\ &\leq (\|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) - w^*\|_H + \lambda_k \|e_k\|_H)^2 \\ &\leq \|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) - w^*\|_H^2 + C \|e_k\|_H \\ &\leq \|w^k - w^*\|_H^2 - \frac{1 - \lambda_k}{\lambda_k} \|(1 - \lambda_k)w^k + \lambda_k \mathcal{L}_M(w^k) - w^k\|_H^2 + C \|e_k\|_H \\ &= \|w^k - w^*\|_H^2 - \lambda_k (1 - \lambda_k) \|\mathcal{L}_M(w^k) - w^k\|_H^2 + C \|e_k\|_H. \end{aligned}$$

It follows from Lemma 2.17 that $\sum_k \lambda_k (1 - \lambda_k) \|\mathcal{L}_M(w^k) - w^k\|_H^2 < +\infty$. Since $\sum_k \lambda_k (1 -$

$\lambda_k) = +\infty$, then $\liminf_{k \rightarrow \infty} \|\mathcal{L}_M(w^k) - w^k\|_H^2 = 0$. We have

$$\begin{aligned}
\|\mathcal{L}_M(w^{k+1}) - w^{k+1}\|_H &= \|\mathcal{L}_M(w^{k+1}) - (1 - \lambda_k)w^k - \lambda_k\mathcal{L}_M(w^k) - \lambda_k e_k\|_H \\
&\leq \|\mathcal{L}_M(w^{k+1}) - \mathcal{L}_M(w^k)\|_H + (1 - \lambda_k)\|\mathcal{L}_M(w^k) - w^k\|_H + \lambda_k\|e_k\|_H \\
&\leq \|w^{k+1} - w^k\|_H + (1 - \lambda_k)\|\mathcal{L}_M(w^k) - w^k\|_H + \lambda_k\|e_k\|_H \\
&= \|\lambda_k(\mathcal{L}_M(w^k) - w^k) + \lambda_k e_k\|_H + (1 - \lambda_k)\|\mathcal{L}_M(w^k) - w^k\|_H + \lambda_k\|e_k\|_H \\
&\leq \|\mathcal{L}_M(w^k) - w^k\|_H + 2\lambda_k\|e_k\|_H.
\end{aligned}$$

Since $\sum_k 2\lambda_k\|e_k\|_H < +\infty$, then, by Lemma 2.17, $\|\mathcal{L}_M(w^k) - w^k\|_H$ converges to 0. Hence, $\{\mathcal{L}_M(w^k) - w^k\}$ converges to 0.

(iii) Suppose $\{w^{k_i}\}$ is a convergent subsequence of $\{w^k\}$, then $\{\mathcal{L}_M(w^{k_i}) - w^{k_i}\}_i$ converges to 0. By the demiclosed principle in Theorem 2.19, $\{w^{k_i}\}$ converges to a fixed point in $\text{Fix}(\text{prox}_{F,R} \circ E)$. So $\mathfrak{C}\{w^k\} \subset \text{Fix}(\text{prox}_{F,R} \circ E)$. Hence, it follows from Theorem 2.18 that $\{w^k\}$ converges to a fixed point in $\text{Fix}(\text{prox}_{F,R} \circ E)$.

□

If \mathcal{L}_M is firmly nonexpansive with respect to H , which is a stronger condition than the nonexpansive assumption mentioned in Theorem 2.20, then the parameter λ_k can be chosen in $[0, 2]$ instead of $[0, 1]$.

Corollary 2.21 Assume that $\text{Fix}(\text{prox}_{F,R} \circ E) \neq \emptyset$. Let \mathcal{L}_M be an M -operator associated with $\text{prox}_{F,R} \circ E$ and \mathcal{L}_M is firmly nonexpansive with respect to $H \in \mathbb{S}_+^d$. Let $\{\lambda_k\}$ be a sequence in $[0, 2]$ and $\{e_k\}$ be a sequence in \mathbb{R}^d such that $\sum_k \lambda_k(2 - \lambda_k) = +\infty$ and $\sum_k \lambda_k\|e_k\|_H < \infty$. Then sequence $\{w^k\}$ generated by equation (2.29) with an initial vector $w^0 \in \mathbb{R}^d$ converges to a point in $\text{Fix}(\text{prox}_{F,R} \circ E)$.

Proof. Since \mathcal{L}_M is firmly nonexpansive with respect to H , then by Lemma 2.4 there exists

a nonexpansive operator R with respect to H such that $\mathcal{L}_M = \frac{1}{2}\text{Id} + \frac{1}{2}R$. Thus,

$$w^{k+1} = w^k + \frac{\lambda_k}{2} (R(w^k) + 2\varepsilon_k - w^k).$$

Then the result immediately follows from Theorem 2.20. □

The convergence results mentioned above guarantee that the sequence generated by Algorithm 1 converges if \mathcal{L}_M satisfies three properties discussed in the previous subsections, and the errors caused by inner iterations are summable.

Next, we study the theoretical convergence rate of Algorithm 1 in the sense of the partial primal-dual gap [82] without taking inner errors into consideration. The partial primal-dual gap of the average of the sequence $\{w^1, w^2, \dots, w^K\}$, generated by Algorithm 1, vanishes with a rate of $O(\frac{1}{K})$, where K is the number of outer iterations.

Proposition 2.22 Let \mathcal{L}_M be an M -operator associated with $\text{prox}_{F,R} \circ E$. Suppose $R(E - I) \in \mathbb{R}^{d \times d}$ is a skew-symmetric matrix and $R(E - M) \in \mathbb{S}_+^d$. Let $w^k \in \mathbb{R}^d$, $k = 0, \dots, K - 1$ be a sequence generated by $w^{k+1} = \mathcal{L}_M(w^k)$. Then, for any vector $w \in \mathbb{R}^d$, the partial primal-dual gap

$$G(\bar{w}^K, w) \leq \frac{1}{2K} \|w - w^0\|_{R(E-M)}^2,$$

where $G(w_1, w_2) = F(w_1) - F(w_2) + \langle w_1, R(E - I)w_2 \rangle$, and $\bar{w}^K = \frac{1}{K} \sum_{k=1}^K w^k$.

Proof. Since $w^{k+1} = \text{prox}_{F,R}(Mw^{k+1} + (E - M)w^k)$, then by equation (2.4)

$$R(E - M)w^k + R(M - I)w^{k+1} \in \partial F(w^{k+1}).$$

By the definition of subdifferential in equation (2.1),

$$\begin{aligned}
F(w) &\geq F(w^{k+1}) + \langle R(E - M)w^k + R(M - I)w^{k+1}, w - w^{k+1} \rangle \\
&= F(w^{k+1}) + \langle R(E - M)w^k - R(E - M)w^{k+1} + R(E - I)w^{k+1}, w - w^{k+1} \rangle \\
&= F(w^{k+1}) + \langle R(E - M)(w^k - w^{k+1}), w - w^{k+1} \rangle + \langle R(E - I)w^{k+1}, w - w^{k+1} \rangle.
\end{aligned}$$

Since $R(E - I)$ is a skew-symmetric matrix, then we have

$$\langle R(E - I)w^{k+1}, w - w^{k+1} \rangle = \langle R(E - I)w^{k+1}, w \rangle = \langle w^{k+1}, R(E - I)w \rangle$$

and

$$F(w) = F(w^{k+1}) + \langle w^{k+1}, R(E - I)w \rangle + \langle w - w^{k+1}, R(E - M)(w^k - w^{k+1}) \rangle.$$

Hence, the partial primal-dual gap

$$\begin{aligned}
G(w^{k+1}, w) &= F(w^{k+1}) - F(w) + \langle w^{k+1}, R(E - I)w \rangle \\
&\leq \langle w - w^{k+1}, R(E - M)(w^{k+1} - w^k) \rangle \\
&= \frac{1}{2} \|w - w^k\|_{R(E-M)}^2 - \frac{1}{2} \|w - w^{k+1}\|_{R(E-M)}^2 - \frac{1}{2} \|w^{k+1} - w^k\|_{R(E-M)}^2.
\end{aligned}$$

Summing both sides from $k = 0, 1, \dots, K - 1$, we have

$$\begin{aligned}
\sum_{k=0}^{K-1} G(w^{k+1}, w) &\leq \frac{1}{2} \|w - w^0\|_{R(E-M)}^2 - \frac{1}{2} \|w - w^K\|_{R(E-M)}^2 - \frac{1}{2} \sum_{k=0}^{K-1} \|w^{k+1} - w^k\|_{R(E-M)}^2 \\
&\leq \frac{1}{2} \|w - w^0\|_{R(E-M)}^2,
\end{aligned}$$

since $\|w - w^K\|_{R(E-M)}^2$ and $\|w^{k+1} - w^k\|_{R(E-M)}^2$ are non-negative.

By the convexity of $G(\cdot, w)$, we have

$$G(\bar{w}^K, w) \leq \frac{1}{2K} \|w - w^0\|_{R(E-M)}^2.$$

□

Chapter 3

Examples of Implicit Fixed-point Proximity Algorithms

In this chapter, we propose to develop several implicit fixed-point proximity algorithms for different optimization problems. Those algorithms are established under the implicit fixed-point proximity framework introduced in Chapter 2. A detailed convergence analysis will be conducted for each proposed implicit algorithm.

Before presenting these implicit fixed-point proximity algorithms, we introduce the general procedure for constructing implicit algorithms for composite optimization problems that can be identified as problem (P1).

First, we formulate the composite optimization problem as problem (P1) and then characterize its solutions as the solutions of a system of fixed point equations in the form of the unified fixed point equation (2.12).

Second, to form an implicit operator \mathcal{L}_M defined in Definition 2.7, we propose a block structure for the matrix M and carefully construct the block matrix elements. In particular, the resulting \mathcal{L}_M can be computed iteratively by contractive mappings.

Third, we obtain an implicit fixed-point proximity algorithm, which generates a sequence

$\{w^k\}$ by the following equation

$$w^{k+1} = w^k + \lambda_k (\mathcal{L}_M(w^k) - w^k), \quad (2.17)$$

where $\lambda_k > 0$ and $\mathcal{L}_M(w^k)$ is the unique solution of the following implicit equation

$$w = \text{prox}_{F,R}(Mw + (E - M)w^k). \quad (2.18)$$

Last, we study the assumptions on algorithm parameters in order to guarantee the proposed \mathcal{L}_M satisfies Property 1, Property 2 and Property 3 under the implicit fixed-point proximity framework. Then a convergence analysis of the proposed implicit fixed-point proximity algorithm is conducted with theoretical results.

Now, we utilize this procedure to develop several implicit fixed-point proximity algorithms. Each algorithm is designed for a different optimization problem with different numbers of convex functions and linear operators.

3.1 Example 1

The first example is to minimize a separable sum of two convex functions, of which one convex function is composited with a linear operator.

$$\min_{x \in \mathbb{R}^n} f_1(x) + f_2(Bx), \quad (\text{P2})$$

where B is an $m \times n$ matrix, and $f_1 : \mathbb{R}^n \rightarrow (-\infty, +\infty]$, and $f_2 : \mathbb{R}^m \rightarrow (-\infty, +\infty]$ are proper, lower semi-continuous, and convex.

Among the optimization problems presented in Chapter 1, the following problems have the form of problem (P2): ROF model, L1-TV denoising model, framelet-based denoising models, and ℓ_1 -regularized linear least squares problem.

3.1.1 Fixed point characterization

Problem (P2) is a special case of problem (P1) with $f : \mathbb{R}^{n+m} \rightarrow (-\infty, +\infty]$ defined by $f(y) = f_1(u) + f_2(v)$, where

$$y = \begin{bmatrix} u \\ v \end{bmatrix} \in \mathbb{R}^{n+m} \quad \text{and} \quad A = \begin{bmatrix} I_n \\ B \end{bmatrix}. \quad (3.1)$$

By applying Proposition 2.6, the solutions of problem (P2) can be characterized as the solutions of the system of fixed point equations (2.9). The second equation in equations (2.9) involves $\text{prox}_{f^*, Q}$, which is the proximity operator of f^* . Note that $f^*(y) = f_1^*(u) + f_2^*(v)$ is a block separable sum of two convex functions. So if we set $Q = \text{diag}(Q_1, Q_2)$, $Q_1 \in \mathbb{S}_+^n$, and $Q_2 \in \mathbb{S}_+^m$, then it follows from the properties of proximity operator for block separable functions that the equation with $\text{prox}_{f^*, Q}$ consists of two equations. One equation is with respect to the variable u , and the other equation is with respect to the variable v . Thus, the system of fixed point equations (2.9) is corresponding to the following system

$$\begin{cases} x = x - P^{-1}(u + B^\top v) \\ u = \text{prox}_{f_1^*, Q_1}(u + Q_1^{-1}x) \\ v = \text{prox}_{f_2^*, Q_2}(v + Q_2^{-1}Bx). \end{cases} \quad (3.2)$$

Although equations (3.2) can characterize the solutions of problem (P2), this system contains a redundant variable due to the fact that one of the matrix blocks of A defined as (3.1) is the identity matrix. The redundant variable may increase the complexity in the procedure later for developing implicit algorithms. So we eliminate the redundant variable from the system and obtain an equivalent variant of equations (3.2) with fewer fixed point equations.

The equivalent variant of equations (3.2) that consists of only two fixed point equations is presented in the following proposition. We combine the first and second equation in

equations (3.2) into one equation in terms of only the variable x , by using equation (2.5) to convert the proximity operator of f_1^* into the proximity operator of f_1 . Then the variable u in the system can be eliminated from the system and the resulting system of fixed point equations is equivalent to equations (3.2).

Proposition 3.1 Suppose that the set of solutions of problem (P2) is nonempty. If a vector $x \in \mathbb{R}^n$ is a solution of problem (P2), then for any $\alpha > 0$ and $\beta > 0$, there exists a vector $y \in \mathbb{R}^m$ such that the following system of equations holds

$$\begin{cases} x = \text{prox}_{\alpha f_1}(x - \alpha B^\top y) \\ y = \text{prox}_{\beta f_2^*}(y + \beta Bx). \end{cases} \quad (3.3)$$

Conversely, if there exist $\alpha > 0$, $\beta > 0$, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ satisfying the system of equations (3.3), then x is a solution of problem (P2).

Proof. According to Proposition 2.6, the solutions of problem (P2) are characterized as equations (3.2). Then, it follows from the first equation in equations (3.2) that $u = -B^\top v$. By substituting this equation into the variable u in the second equation in equations (3.2), we have

$$-B^\top v = \text{prox}_{f_1^*, Q_1}(-B^\top v + Q_1^{-1}x).$$

This equation can be further rewritten as follows, by using equation (2.5) to convert the proximity operator of f_1^* into the proximity operator of f_1 .

$$\begin{aligned} -B^\top v &= -B^\top v + Q_1^{-1}x - Q_1^{-1}\text{prox}_{f_1, Q_1^{-1}}(Q_1(-B^\top v + Q_1^{-1}x)) \\ x &= \text{prox}_{f_1, Q_1^{-1}}(-Q_1 B^\top v + x). \end{aligned}$$

Then the variable u is eliminated from equations (3.2) and the first two equations in equations

(3.2) are combined into one equation. We obtain the system of equations (3.3) by redefining $v = y$, and choosing $Q_1 = \alpha I_n$ and $Q_2 = \frac{1}{\beta} I_m$, where $\alpha > 0$ and $\beta > 0$.

□

Under the implicit fixed-point proximity framework, the system of fixed point equations (3.3) that characterizes the solutions of problem (P2) is written as the unified fixed point equation (2.12) with $F : \mathbb{R}^{n+m} \rightarrow (-\infty, +\infty]$ defined by $F(w) = f_1(x) + f_2^*(y)$, $R = \text{diag}\left(\frac{1}{\alpha} I_n, \frac{1}{\beta} I_m\right)$, $\alpha > 0, \beta > 0$,

$$w = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{n+m} \quad \text{and} \quad E = \begin{bmatrix} I_n & -\alpha B^\top \\ \beta B & I_m \end{bmatrix} \in \mathbb{R}^{(n+m) \times (n+m)}. \quad (3.4)$$

Then a fixed-point proximity algorithm, developed from this unified fixed point equation, generates a sequence $\{w^k\}$ that follows equation (2.17) with function F and matrices E and R defined as above.

3.1.2 Existing fixed-point proximity algorithms

There are three existing fixed-point proximity algorithms for solving problem (P2) introduced in Section 2.4. They are primal dual algorithm (2.21), FP²O (2.23), and ASBI (2.25). Among those algorithms, there are two block structures of the matrix M observed. PD has the strictly block lower triangular matrix structure of M as shown in equation (2.22) and its \mathcal{L}_M has a closed form. Both of FP²O and ASBI have the matrix structure of M with one nonzero diagonal block as shown in equation (2.24) and (2.26) respectively, and their \mathcal{L}_M have implicit expressions but can be computed explicitly.

The existing fixed-point algorithms that have the block structures mentioned above are explicit algorithms. They either have a relatively strict convergence assumption or can be applied to a limited number of optimization problems. So in the next section, we aim to propose a novel block structure of M and develop a fixed-point proximity algorithm with a

fully implicit \mathcal{L}_M .

3.1.3 Implicit fixed-point proximity algorithm

In the following, we first propose a block structure for the matrix M to develop an implicit fixed-point proximity algorithm for problem (P2) via contractive mappings, and then study the convergence assumptions for the proposed implicit algorithm.

First, we start with a block structure for the matrix M , which has skew diagonal blocks defined as follows

$$M = \begin{bmatrix} 0 & M_1 \\ M_2 & 0 \end{bmatrix}, \quad (3.5)$$

where $M_1 \in \mathbb{R}^{n \times m}$ and $M_2 \in \mathbb{R}^{m \times n}$.

Due to the skew diagonal block structure of M , \mathcal{L}_M has an implicit expression and equation (2.18) for solving $w = \mathcal{L}_M(w^k)$ under the implicit fixed-point proximity framework is corresponding to the following system of fixed point equations

$$\begin{cases} x = \text{prox}_{\alpha f_1}(M_1 y + x^k + (-\alpha B^\top - M_1)y^k) \\ y = \text{prox}_{\beta f_2^*}(M_2 x + (\beta B - M_2)x^k + y^k). \end{cases} \quad (3.6)$$

Second, we construct a contractive mapping to solve this system (3.6). The skew diagonal block structure of M provides a convenient way to construct a contractive mapping for solving equations (3.6). In particular, equations (3.6) can be recognized as a fixed point problem in terms of only one unknown variable. For example, by substituting the second equation into the y variable in the first equation, we obtain the following equation in terms of only one variable x

$$x = \text{prox}_{\alpha f_1}(M_1 \text{prox}_{\beta f_2^*}(M_2 x + (\beta B - M_2)x^k + y^k) + x^k + (-\alpha B^\top - M_1)y^k).$$

This implicit equation, for any given vector x^k and y^k , can be viewed as a fixed point equation

$x = \mathcal{T}(x)$ where $\mathcal{T} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is defined as

$$\mathcal{T}(x) = \text{prox}_{\alpha f_1}(M_1 \text{prox}_{\beta f_2^*}(M_2 x + (\beta B - M_2)x^k + y^k) + x^k + (-\alpha B^\top - M_1)y^k). \quad (3.7)$$

If $\|M_1\|_2 \|M_2\|_2 < 1$, then the operator \mathcal{T} is a contractive mapping and there exists an iterative sequence that converges to the unique solution of the fixed point problem.

Third, the implicit fixed-point proximity algorithm that has the matrix M with the skew diagonal block structure defined as (3.5) is presented as follows.

Algorithm 2 Implicit Fixed-point Proximity Algorithm for problem (P2)

- 1: Choose , $x^0 \in \mathbb{R}^n$, $y^0 \in \mathbb{R}^m$, $\lambda_k \in [0, 2]$
 - 2: **for** k from 1 to K **do**
 - 3: Compute $\mathcal{L}_M(w^k)$ via the inner loop:
 - 4: Set $l = 0$, choose $x_0^{k+1} \in \mathbb{R}^n$
 - 5: **repeat**
 - 6: $y_{l+1}^{k+1} = \text{prox}_{\beta f_2^*}(M_2 x_l^{k+1} + (\beta B - M_2)x^k + y^k)$ ▷ Inner step 1
 - 7: $x_{l+1}^{k+1} = \text{prox}_{\alpha f_1}(M_1 y_{l+1}^{k+1} + x^k + (-\alpha B^\top - M_1)y^k)$ ▷ Inner step 2
 - 8: $l \leftarrow l + 1$
 - 9: **until** stopping criterion is satisfied
 - 10: $\mathcal{L}_M(w^k) = w_\infty^{k+1}$ ▷ w_∞^{k+1} is the output from the inner loop
 - 11: $w^{k+1} = w^k + \lambda_k(\mathcal{L}_M(w^k) - w^k)$
 - 12: **end for**
-

In order to ensure that the proposed \mathcal{L}_M with a fully implicit expression satisfies Property 1, Property 2 and Property 3 under the implicit fixed-point proximity framework, there are three convergence assumptions for Algorithm 2.

Assumption (A2.1) $\|M_1\|_2\|M_2\|_2 < 1$;

Assumption (A2.2) $\frac{1}{\beta}M_2 - \frac{1}{\alpha}M_1^\top = 2B$;

Assumption (A2.3) $I_m - \alpha\beta\left(B - \frac{1}{\beta}M_2\right)\left(-B^\top - \frac{1}{\alpha}M_1\right) \in \mathbb{S}_+^m$.

3.1.4 Convergence analysis

In the following, we illustrate that Assumption (A2.1), (A2.2) and (A2.3) can guarantee Property 1, Property 2 and Property 3 on the operator \mathcal{L}_M with the choice of M defined as (3.5), and then conduct a convergence analysis on Algorithm 2.

(i) Property 1 and Property 2

Assumption (A2.1) yields that the operator \mathcal{T} defined as (3.7) is a contractive mapping, so its corresponding \mathcal{L}_M satisfies Property 1 and Property 2 as shown in the following proposition.

Proposition 3.2 Let M be an $(n+m) \times (n+m)$ matrix defined as (3.5). Suppose that $\|M_1\|_2\|M_2\|_2 < 1$. Then \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$. Furthermore, for any vector $w^k = \begin{bmatrix} x^k^\top & y^k^\top \end{bmatrix}^\top \in \mathbb{R}^{n+m}$, the sequence $\{w_l^{k+1}\}_l$ generated by the 6th-7th line in Algorithm 2, that is,

$$\begin{cases} y_{l+1}^{k+1} = \text{prox}_{\beta f_2^*}(M_2 x_l^{k+1} + (\beta B - M_2)x^k + y^k) \\ x_{l+1}^{k+1} = \text{prox}_{\alpha f_1}(M_1 y_{l+1}^{k+1} + x^k + (-\alpha B^\top - M_1)y^k) \end{cases} \quad (3.8)$$

converges to $\mathcal{L}_M(w^k)$, given any initial vector $x_0^{k+1} \in \mathbb{R}^n$.

Proof. Let \mathcal{T} be the operator defined as (3.7). Suppose that $\|M_1\|_2\|M_2\|_2 < 1$. Since $\text{prox}_{\beta f_2^*}$ and $\text{prox}_{\alpha f_1}$ are firmly nonexpansive with respect to the standard ℓ_2 norm, then \mathcal{T} is contractive for any vector $w^k \in \mathbb{R}^{n+m}$. Thus, it follows from Banach fixed point theorem in

Theorem 2.10 that the fixed point of \mathcal{T} is unique and that \mathcal{L}_M is an M -operator. Moreover, the sequence $\{w_l^{k+1}\}_l$, generated by equations (3.8), converges to the unique solution $\mathcal{L}_M(w^k)$. \square

Assumption (A2.1) $\|M_1\|_2\|M_2\|_2 < 1$ for Algorithm 2 is different from Assumption (A1.1) $\|M\|_R < 1$ for Algorithm 1, which is an implicit fixed-point proximity algorithm designed in a general setting. Here, the norm $\|\cdot\|_R$ can be replaced by the standard ℓ_2 norm, because both $\text{prox}_{\alpha f_1}$ and $\text{prox}_{\beta f_2^*}$ are firmly nonexpansive with respect to the standard ℓ_2 norm. Also, the assumption $\|M\|_2 = \max(\|M_1\|_2, \|M_2\|_2) < 1$ can be replaced by a weaker assumption $\|M_1\|_2\|M_2\|_2 < 1$, thanks to the skew diagonal block structure of M .

(ii) Property 3

Assumption (A2.2) and (A2.3) in Algorithm 2 guarantee that \mathcal{L}_M satisfies Property 3. According to Proposition 2.14, \mathcal{L}_M is firmly nonexpansive with respect to $R(E - M)$ if $R(E - I)$ is skew symmetric and $R(E - M) \in \mathbb{S}_+^{n+m}$.

In Algorithm 2, the matrix E is defined as (3.4) and the matrix M is defined as (3.5). Then the matrix $R(E - I)$ and $R(E - M)$ can be computed as follows

$$R(E - I) = \begin{bmatrix} 0 & -B^\top \\ B & 0 \end{bmatrix} \quad \text{and} \quad R(E - M) = \begin{bmatrix} \frac{1}{\alpha}I_n & -B^\top - \frac{1}{\alpha}M_1 \\ B - \frac{1}{\beta}M_2 & \frac{1}{\beta}I_m \end{bmatrix}.$$

It is clear that $R(E - I)$ is skew symmetric. And by using the following lemma, $R(E - M) \in \mathbb{S}_+^{n+m}$ can also be verified, provided that Assumption (A2.2) and Assumption (A2.3) are satisfied.

Lemma 3.3 Let M be defined as (3.5). Then the matrix $R(E - M) \in \mathbb{S}_+^{n+m}$ if and only if $\frac{1}{\beta}M_2 - \frac{1}{\alpha}M_1^\top = 2B$, and $I_m - \alpha\beta \left(B - \frac{1}{\beta}M_2\right) \left(-B^\top - \frac{1}{\alpha}M_1\right) \in \mathbb{S}_+^m$.

Proof. The result immediately follows from Lemma 2.15.

□

In summary, for the matrix M with skew diagonal blocks defined as (3.5), Assumption (A2.1), (A2.2) and (A2.3) in Algorithm 2 can imply that its corresponding operator \mathcal{L}_M satisfies three essential properties required under the fixed-point proximity framework.

(iii) Convergence theorem

Next, we conduct a convergence analysis for Algorithm 2 with a specific choice of M defined as follows

$$M = \begin{bmatrix} 0 & (-1 - \theta)\alpha B^\top \\ (1 - \theta)\beta B & 0 \end{bmatrix}, \quad (3.9)$$

where $\theta \in \mathbb{R}$. Then the corresponding $R(E - M)$ automatically satisfies Assumption (A2.2) shown as follows

$$R(E - M) = \begin{bmatrix} \frac{1}{\alpha}I_n & \theta B^\top \\ \theta B & \frac{1}{\beta}I_m \end{bmatrix}.$$

If $|\theta| = 1$, then M is a strictly block lower or upper triangular matrix and \mathcal{L}_M has an explicit expression, which results in an explicit fixed-point proximity algorithm. In particular, when $\theta = -1$, the fixed-point proximity algorithm is exactly primal dual algorithm (2.21) and it converges if $\alpha\beta\|B\|_2^2 < 1$.

If $|\theta| \neq 1$, then \mathcal{L}_M has an implicit expression, which results in an implicit fixed-point proximity algorithm as Algorithm 2. The three convergence assumptions under the implicit fixed-point proximity framework can be reinterpreted. Assumption (A2.1) is corresponding to $\alpha\beta|1 - \theta^2|\|B\|_2^2 < 1$, which ensures that inner iterations (3.8) are generated by contractive mappings. Assumption (A2.3) is corresponding to $\alpha\beta\theta^2\|B\|_2^2 < 1$, which guarantees that $R(E - M) \in \mathbb{S}_+^{n+m}$.

The convergence assumptions of the proposed implicit fixed-point proximity algorithm with the choice of M defined as (3.9) with $|\theta| \neq 1$ can be summarized in the following theorem.

Theorem 3.4 Assume that $\text{Fix}(\text{prox}_{F,R} \circ E) \neq \emptyset$. Let M be defined as (3.9) with $|\theta| \neq 1$ and let $\lambda_k \in [0, 2]$ such that $\sum_k \lambda_k(2 - \lambda_k) = +\infty$. Suppose that the following conditions hold

$$(a) \quad \alpha\beta|1 - \theta^2|\|B\|_2^2 < 1;$$

$$(b) \quad \alpha\beta\theta^2\|B\|_2^2 < 1.$$

Then the sequence $\{w^k\}$, generated by Algorithm 2, converges to a solution of problem (P2) if the errors from inner iterations are summable.

Proof. Assumption (A2.2) is automatically satisfied. The condition (a) implies Assumption (A2.1), and the condition (b) implies Assumption (A2.3). Thus, the operator \mathcal{L}_M , corresponding to the matrix M defined as (3.9) with $|\theta| \neq 1$, has Property 1, Property 2 and Property 3 under the implicit fixed-point proximity framework. Therefore, the result follows from Theorem 2.20.

□

Compared to the explicit fixed-point proximity algorithm with $|\theta| = 1$, the proposed implicit fixed-point proximity algorithm with $|\theta| \neq 1$ allows a wider selection range of the parameters α and β , which are the parameters of the proximity operators. For example, the convergence assumption is $\alpha\beta\|B\|_2^2 < 1$ in the case when $|\theta| = 1$, while the convergence assumption is $\alpha\beta\|B\|_2^2 < 2$ in the case when $|\theta| = \frac{1}{\sqrt{2}} \neq 1$. In Chapter 4, we will discuss the sensitivity of the parameter settings for the proposed implicit algorithm and demonstrate that the proposed implicit algorithm outperforms existing explicit fixed-point proximity algorithms in terms of computational time.

3.2 Example 2

The second example is to minimize a separable sum of two convex functions composited with linear operators

$$\min_{x \in \mathbb{R}^n} f_1(A_1x) + f_2(A_2x), \quad (\text{P3})$$

where A_i is an $m_i \times n$ matrix, and $f_i : \mathbb{R}^{m_i} \rightarrow (-\infty, +\infty]$ is proper, lower semi-continuous, and convex, $i = 1, 2$.

Among the optimization problems mentioned in Chapter 1, the following problems have the form of problem (P3): L2-TV image restoration model, L1-TV image restoration model, and framelet-based deblurring models.

3.2.1 Fixed point characterization

Problem (P3) is a special case of problem (P1) with $f : \mathbb{R}^{m_1+m_2} \rightarrow (-\infty, +\infty]$ defined by $f(y) = f_1(u) + f_2(v)$, where

$$y = \begin{bmatrix} u \\ v \end{bmatrix} \in \mathbb{R}^{m_1+m_2} \quad \text{and} \quad A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}.$$

By applying Proposition 2.6, the solutions of problem (P3) can be characterized as the solutions of the system of fixed point equations (2.9) with $y = \begin{bmatrix} u^\top & v^\top \end{bmatrix}^\top \in \mathbb{R}^{m_1+m_2}$, $P \in \mathbb{S}_+^n$, $Q = \text{diag}(Q_1, Q_2)$, $Q_1 \in \mathbb{S}_+^{m_1}$, and $Q_2 \in \mathbb{S}_+^{m_2}$. That is the following system of equations

$$\begin{cases} x = x - P^{-1}(A_1^\top u + A_2^\top v) \\ u = \text{prox}_{f_1^*, Q_1}(u + Q_1^{-1} A_1 x) \\ v = \text{prox}_{f_2^*, Q_2}(v + Q_2^{-1} A_2 x). \end{cases} \quad (3.10)$$

The system of fixed point equations (3.10) is suitable for developing parallel algorithms, because the second equation does not involve the variable v and the third equation does not

involve the variable u . Then u and v can be computed simultaneously without using each other's newest update.

However, u and v , in fact, are related to each other and connected through the variable x in the first equation. So we derive an equivalent variant of equations (3.10) by substituting the first equation with a different parameter $\tilde{P}^{-1} \in \mathbb{S}^n$ into the variable x in the second and third equations. This variant reveals all the underlying relations between the variables and has more flexibility for developing sequential algorithms.

Proposition 3.5 Suppose that the set of solutions of problem (P3) is nonempty. If a vector $x \in \mathbb{R}^n$ is a solution of problem (P3), then, for any $\alpha_1 > 0$, $\alpha_2 > 0$, $\gamma > 0$ and $\beta \geq 0$, there exists vectors $u \in \mathbb{R}^{m_1}$ and $v \in \mathbb{R}^{m_2}$ such that the following equations hold

$$\begin{cases} u = \text{prox}_{\alpha_1 f_1^*}(u + \alpha_1 A_1(x - \beta(A_1^\top u + A_2^\top v))) \\ v = \text{prox}_{\alpha_2 f_2^*}(v + \alpha_2 A_2(x - \beta(A_1^\top u + A_2^\top v))) \\ x = x - \gamma(A_1^\top u + A_2^\top v). \end{cases} \quad (3.11)$$

Conversely, if there exist $\alpha_1 > 0$, $\alpha_2 > 0$, $\gamma > 0$, $\beta \geq 0$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^{m_1}$ and $v \in \mathbb{R}^{m_2}$ satisfying the system of equations (3.11), then x is a solution of problem (P3).

Proof. According to Proposition 2.6, the solutions of problem (P3) are characterized as equations (3.10). The following equation is the first equation in equations (3.10) with a new parameter $\tilde{P}^{-1} = \beta I_n$, $\beta \geq 0$,

$$x = x - \beta(A_1^\top u + A_2^\top v).$$

We substitute this equation into the variable x in the second and third equations in equations (3.10), and then obtain the system of equations (3.11) by interchanging equations and choosing $P = \frac{1}{\gamma}I_n$, $Q_1 = \frac{1}{\alpha_1}I_{m_1}$, and $Q_2 = \frac{1}{\alpha_2}I_{m_2}$, where $\gamma > 0$, $\alpha_1 > 0$, and $\alpha_2 > 0$.

□

Under the implicit fixed-point proximity framework, the system of fixed point equations (3.11) which characterizes the solutions of problem (P3) is written as the unified fixed point equation (2.12) with $F : \mathbb{R}^{m_1+m_2+n} \rightarrow \mathbb{R}$ defined by $F(w) = f_1^*(u) + f_2^*(v)$, $R = \text{diag} \left(\frac{1}{\alpha_1} I_{m_1}, \frac{1}{\alpha_2} I_{m_2}, \frac{1}{\gamma} I_n \right)$, $\alpha_1 > 0, \alpha_2 > 0, \gamma > 0, \beta \geq 0$,

$$w = \begin{bmatrix} u \\ v \\ x \end{bmatrix} \in \mathbb{R}^{m_1+m_2+n}$$

and

$$E = \begin{bmatrix} I_{m_1} - \alpha_1 \beta A_1 A_1^\top & -\alpha_1 \beta A_1 A_2^\top & \alpha_1 A_1 \\ -\alpha_2 \beta A_2 A_1^\top & I_{m_2} - \alpha_2 \beta A_2 A_2^\top & \alpha_2 A_2 \\ -\gamma A_1^\top & -\gamma A_2^\top & I_n \end{bmatrix} \in \mathbb{R}^{(m_1+m_2+n) \times (m_1+m_2+n)}. \quad (3.12)$$

Then a fixed-point proximity algorithm, developed from this unified fixed point equation, generates a sequence $\{w^k\}$ that follows equation (2.17) with function F and matrices E and R defined as above. Apparently, it is more challenging to develop implicit fixed-point proximity algorithms for problem (P3) compared to problem (P2).

3.2.2 Existing fixed-point proximity algorithms

We present one existing fixed-point proximity algorithm for solving problem (P3).

- *Fixed-point Proximity Gauss-Seidel Method (FPGS)* [16]: This algorithm is developed from the perspective of the fixed-point proximity framework and the Gauss-Seidel method.

$$\begin{cases} u^{k+1} = \text{prox}_{\alpha_1 f_1^*}(u^k + \alpha_1 A_1(x^k - \beta(A_1^\top u^k + A_2^\top v^k))) \\ v^{k+1} = \text{prox}_{\alpha_2 f_2^*}(v^k + \alpha_2 A_2(x^k - \beta(A_1^\top u^{k+1} + A_2^\top v^k))) \\ x^{k+1} = x^k - \gamma(A_1^\top u^{k+1} + A_2^\top v^{k+1}). \end{cases} \quad (3.13)$$

This algorithm can be identified as a fixed point proximity algorithm generated by $w^{k+1} = \text{prox}_{F,R}(Mw^{k+1} + (E - M)w^k)$, where the function F , and matrices R and E are defined as (3.12), and the matrix M is chosen as follows

$$M = \begin{bmatrix} 0 & 0 & 0 \\ -\alpha_2\beta A_2 A_1^\top & 0 & 0 \\ -2\gamma A_1^\top & -2\gamma A_2^\top & 0 \end{bmatrix}. \quad (3.14)$$

The algorithm converges if $\alpha_1\beta\|A_1\|_2^2 < 1$, $\alpha_2\beta\|A_2\|_2^2 < 1$, $\gamma \in (0, \beta]$, and $\beta > 0$.

In FPGS (3.13), the matrix M is a strictly block lower triangular matrix, and u^{k+1} , v^{k+1} and x^{k+1} can be computed explicitly in a sequential order, so \mathcal{L}_M has an explicit expression and satisfies Property 1 and Property 2. It is proved in [16] that the sequence $\{w^k\}$ generated by FPGS converges, even though the authors did not verify Property 3.

3.2.3 Implicit fixed-point proximity algorithms

In the following, we first propose a block structure for the matrix M to develop an implicit fixed-point proximity algorithm for problem (P3) via contractive mappings, and then study the convergence assumptions for the proposed implicit algorithm.

First, to develop an implicit fixed-point proximity algorithm for problem (P3), we construct a matrix M with a novel block structure defined as follows

$$M = \begin{bmatrix} 0 & -\alpha_1\theta A_1 A_2^\top & 0 \\ -\alpha_2\theta A_2 A_1^\top & 0 & 0 \\ -\rho A_1^\top & -\rho A_2^\top & 0 \end{bmatrix}, \quad (3.15)$$

where $\theta > 0$ and $\rho > 0$.

Under the implicit fixed-point proximity framework, the operator \mathcal{L}_M corresponding to

this matrix M has an implicit expression and $w = \mathcal{L}_M(w^k)$ is the unique solution of the following system of fixed point equations

$$\begin{cases} u = \text{prox}_{\alpha_1 f_1^*}(u^k + \alpha_1 A_1(x^k - \theta A_2^\top(v - v^k) - \beta(A_1^\top u^k + A_2^\top v^k))) \\ v = \text{prox}_{\alpha_2 f_2^*}(v^k + \alpha_2 A_2(x^k - \theta A_1^\top(u - u^k) - \beta(A_1^\top u^k + A_2^\top v^k))) \\ x = x^k - \rho(A_1^\top u + A_2^\top v) + (\rho - \gamma)(A_1^\top u^k + A_2^\top v^k). \end{cases} \quad (3.16)$$

We mainly study the following two cases with different parameters.

The first case is when $\beta = 0$ and $\rho = 2\gamma$, then equations (3.16) become the following equations

$$\begin{cases} u = \text{prox}_{\alpha_1 f_1^*}(u^k + \alpha_1 A_1(x^k - \theta A_2^\top(v - v^k))) \\ v = \text{prox}_{\alpha_2 f_2^*}(v^k + \alpha_2 A_2(x^k - \theta A_1^\top(u - u^k))) \\ x = x^k - 2\gamma(A_1^\top u + A_2^\top v) + \gamma(A_1^\top u^k + A_2^\top v^k). \end{cases} \quad (3.17)$$

The second case is when $\beta > 0$, $\theta = \beta$ and $\rho = \gamma$, then equations (3.16) become the following equations

$$\begin{cases} u = \text{prox}_{\alpha_1 f_1^*}(u^k + \alpha_1 A_1(x^k - \beta(A_1^\top u^k + A_2^\top v))) \\ v = \text{prox}_{\alpha_2 f_2^*}(v^k + \alpha_2 A_2(x^k - \beta(A_1^\top u + A_2^\top v^k))) \\ x = x^k - \gamma(A_1^\top u + A_2^\top v). \end{cases} \quad (3.18)$$

Next, we construct a contractive mapping for solving equations (3.16). In equations (3.16), the variable x can be computed explicitly after the newest updates of u and v , but the variable u and v have to be compute iteratively. So we formulate a contractive mapping from the first two equations to solve u and v .

By substituting the first equation into the variable u in the second equation, we obtain

the following fixed point equation in terms of only one variable v

$$v = \text{prox}_{\alpha_2 f_2^*}(-\alpha_2 \theta A_2 A_1^\top \text{prox}_{\alpha_1 f_1^*}(-\alpha_1 \theta A_1 A_2^\top v + \tilde{u}^k) + \tilde{v}^k),$$

where

$$\begin{aligned} \tilde{u}^k &= u^k + \alpha_1 A_1 (x^k - \beta A_1^\top u^k - (\beta - \theta) A_2^\top v^k), \\ \tilde{v}^k &= v^k + \alpha_2 A_2 (x^k - \beta A_2^\top v^k - (\beta - \theta) A_1^\top u^k). \end{aligned}$$

This implicit equation, for any given vector x^k , u^k and v^k , can be viewed as a fixed point equation $v = \mathcal{T}(v)$ where $\mathcal{T} : \mathbb{R}^{m_2} \rightarrow \mathbb{R}^{m_2}$ is defined as

$$\mathcal{T}(v) = \text{prox}_{\alpha_2 f_2^*}(-\alpha_2 \theta A_2 A_1^\top \text{prox}_{\alpha_1 f_1^*}(-\alpha_1 \theta A_1 A_2^\top v + \tilde{u}^k) + \tilde{v}^k). \quad (3.19)$$

If $\alpha_1 \alpha_2 \theta^2 \|A_1\|_2^2 \|A_2\|_2^2 < 1$, then the operator \mathcal{T} is a contractive mapping and there exists an iterative sequence that converges to the unique solution of the fixed point problem.

Finally, the implicit fixed-point proximity algorithm with the matrix M defined as (3.15) is presented as follows.

Algorithm 3 Implicit Fixed-point Proximity Algorithm for problem (P3)

```

1: Choose  $w^0 \in \mathbb{R}^{m_1+m_2+n}$ ,  $\lambda_k \in [0, 1]$ 

2: for  $k$  from 1 to  $K$  do

3:   Compute  $\mathcal{L}_M(w^k)$  via the inner loop:

4:   Set  $l = 0$ , choose  $v_0^{k+1} \in \mathbb{R}^{m_2}$ 

5:   repeat ▷ Inner steps

6:      $u_{l+1}^{k+1} = \text{prox}_{\alpha_1 f_1^*}(u^k + \alpha_1 A_1(x^k - \theta A_2^\top(v_l^{k+1} - v^k) - \beta(A_1^\top u^k + A_2^\top v^k)))$ 

7:      $v_{l+1}^{k+1} = \text{prox}_{\alpha_2 f_2^*}(v^k + \alpha_2 A_2(x^k - \theta A_1^\top(u_{l+1}^{k+1} - u^k) - \beta(A_1^\top u^k + A_2^\top v^k)))$ 

8:      $l \leftarrow l + 1$ 

9:   until stopping criterion is satisfied

10:   $x_\infty^{k+1} = x^k - \rho(A_1^\top u_\infty^{k+1} + A_2^\top v_\infty^{k+1}) + (\rho - \gamma)(A_1^\top u^k + A_2^\top v^k)$ 
▷  $u_\infty^{k+1}$  and  $v_\infty^{k+1}$  are the outputs from the inner loop

11:   $\mathcal{L}_M(w^k) = w_\infty^{k+1}$ 

12:   $w^{k+1} = w^k + \lambda_k(\mathcal{L}_M(w^k) - w^k)$ 

13: end for

```

In order to ensure that the proposed \mathcal{L}_M with a fully implicit expression satisfies Property 1 and Property 2 under the implicit fixed-point proximity framework, Algorithm 2 needs to satisfy the following assumption.

Assumption (A3.1) $\alpha_1 \alpha_2 \theta^2 \|A_1\|_2^2 \|A_2\|_2^2 < 1$.

In order to ensure that \mathcal{L}_M satisfies Property 3, Algorithm 3 needs to satisfy one of the following assumptions.

Assumption (A3.2) $\beta = 0$, $\rho = 2\gamma$, and $\gamma \left\| \begin{bmatrix} \sqrt{\alpha_1} A_1 & \sqrt{\alpha_2} A_2 \end{bmatrix} \right\|_2^2 < 1$;

Assumption (A3.3) $\beta > 0$, $\rho = \gamma$, $\theta = \beta$, $\alpha_1\beta\|A_1\|_2^2 < 1$, $\alpha_2\beta\|A_2\|_2^2 < 1$, and $\gamma \in (0, 2\beta]$.

3.2.4 Convergence analysis

In the following, we illustrate that Assumption (A3.1) can guarantee that the operator \mathcal{L}_M with the choice of M defined as (3.15) satisfies Property 1 and Property 2, and that Assumption (A3.2) or Assumption (A3.3) can guarantee that \mathcal{L}_M satisfies Property 3, resulting in the convergence of Algorithm 3.

(i) Property 1 and Property 2

Assumption (A3.1) yields that the operator \mathcal{T} defined as (3.19) is a contractive mapping, so the operator \mathcal{L}_M corresponding to Algorithm 3 satisfies Property 1 and Property 2 as shown in the following proposition.

Proposition 3.6 Let M be a matrix defined as (3.15). Suppose that $\alpha_1\alpha_2\theta^2\|A_1\|_2^2\|A_2\|_2^2 < 1$. Then \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$. Furthermore, for any $w^k = \begin{bmatrix} u^k^\top & v^k^\top & x^k^\top \end{bmatrix}^\top \in \mathbb{R}^{m_1+m_2+n}$, the sequence $\{w_l^{k+1}\}_l$, generated by the 6th-7th line and the 10th line in Algorithm 3, that is

$$\begin{cases} u_{l+1}^{k+1} = \text{prox}_{\alpha_1 f_1^*}(u^k + \alpha_1 A_1(x^k - \theta A_2^\top(v_l^{k+1} - v^k) - \beta(A_1^\top u^k + A_2^\top v^k))) \\ v_{l+1}^{k+1} = \text{prox}_{\alpha_2 f_2^*}(v^k + \alpha_2 A_2(x^k - \theta A_1^\top(u_{l+1}^{k+1} - u^k) - \beta(A_1^\top u^k + A_2^\top v^k))) \\ x_{l+1}^{k+1} = x^k - \rho(A_1^\top u_{l+1}^{k+1} + A_2^\top v_{l+1}^{k+1}) + (\rho - \gamma)(A_1^\top u^k + A_2^\top v^k), \end{cases} \quad (3.20)$$

converges to $\mathcal{L}_M(w^k)$, given any initial vector $v_0^{k+1} \in \mathbb{R}^{m_2}$.

Proof. Let \mathcal{T} be the operator defined as (3.19). Suppose that $\alpha_1\alpha_2\theta^2\|A_1\|_2^2\|A_2\|_2^2 < 1$. Since $\text{prox}_{\alpha_1 f_1^*}$ and $\text{prox}_{\alpha_2 f_2^*}$ are firmly nonexpansive with respect to the standard ℓ_2 norm, then the operator \mathcal{T} is contractive for any vector u^k , v^k and x^k . Thus, it follows from Theorem 2.10 that the fixed point of \mathcal{T} is unique and that \mathcal{L}_M is an M -operator. Moreover, the sequence

$\{w_l^{k+1}\}_l$, generated by equations (3.20), converges to the unique solution $\mathcal{L}_M(w^k)$. \square

(ii) Property 3 when $\beta = 0$

In the case when $\beta = 0$, Property 3 can be verified by applying Proposition 2.14. The two assumptions required in Proposition 2.14 are $R(E - I)$ is skew-symmetric and $R(E - M) \in \mathbb{S}_+^{m_1+m_2+n}$.

If $\beta = 0$, then the matrix $R(E - I)$ is skew-symmetric as shown below

$$R(E - I) = \begin{bmatrix} & A_1 \\ & A_2 \\ -A_1^\top & -A_2^\top \end{bmatrix}.$$

Lemma 3.7 Let M be defined as (3.15) with $\beta = 0$. Then the matrix

$$R(E - M) = \begin{bmatrix} \frac{1}{\alpha_1}I & & A_1 \\ & \frac{1}{\alpha_2}I & A_2 \\ (\frac{\rho}{\gamma} - 1)A_1^\top & (\frac{\rho}{\gamma} - 1)A_2^\top & \frac{1}{\gamma}I \end{bmatrix} \in \mathbb{S}_+^{m_1+m_2+n}$$

if and only if $\rho = 2\gamma$ and $\gamma \left\| \begin{bmatrix} \sqrt{\alpha_1}A_1 & \sqrt{\alpha_2}A_2 \end{bmatrix} \right\|_2^2 < 1$.

Proof. It follows from Lemma 2.15 that $R(E - M) \in \mathbb{S}_+^{m_1+m_2+n}$ if and only if $\rho = 2\gamma$ and $\frac{1}{\gamma}I_n - \alpha_1 A_1^\top A_1 - \alpha_2 A_2^\top A_2 \in \mathbb{S}_+^n$. \square

It follows from the results above and Proposition 2.14 that Assumption (A3.2) implies that the operator \mathcal{L}_M is firmly nonexpansive with respect to $R(E - M)$ and satisfies Property 3. As a result, we have the following convergence result for Algorithm 3 with M defined as (3.15) for the case when $\beta = 0$.

Theorem 3.8 Assume that $\text{Fix}(\text{prox}_{F,R} \circ E) \neq \emptyset$. Let M be defined as (3.15), and let

$\lambda_k \in [0, 2]$ such that $\sum_k \lambda_k(2 - \lambda_k) = +\infty$. Then the sequence $\{w^k\}$, generated by Algorithm 3, converges to a solution of problem (P3) if Assumption (A3.1) and (A3.2) are satisfied, and the errors from inner iterations are summable.

Proof. The result immediately follows from Proposition 3.6, Lemma 3.7, Proposition 2.14 and Theorem 2.20. □

(iii) Property 3 when $\beta > 0$

In the case when $\beta > 0$, Property 3 cannot be verified by Proposition 2.14, because $R(E - I)$ is no longer skew-symmetric as shown below

$$R(E - I) = \begin{bmatrix} -\beta A_1 A_1^\top & -\beta A_1 A_2^\top & A_1 \\ -\beta A_2 A_1^\top & -\beta A_2 A_2^\top & A_2 \\ -A_1^\top & -A_2^\top & \end{bmatrix}.$$

So we have to use the definition of nonexpansive operators in Definition 2.2 to show Property 3, which is that \mathcal{L}_M is nonexpansive.

In the following, we prove that Assumption (A3.3) implies that \mathcal{L}_M is nonexpansive.

First, we assume $\beta > 0$ and $\rho = \gamma$ as in Assumption (A3.3). Let

$$w_i = \begin{bmatrix} u_i \\ v_i \\ x_i \end{bmatrix} \in \mathbb{R}^{m_1+m_2+n} \quad \text{and} \quad w_i^* = \begin{bmatrix} u_i^* \\ v_i^* \\ x_i^* \end{bmatrix} \in \mathbb{R}^{m_1+m_2+n}, \quad (3.21)$$

$i = 1, 2$, such that $w_2 = \mathcal{L}_M(w_1)$ and $w_2^* = \mathcal{L}_M(w_1^*)$.

By applying Lemma 2.13, we obtain the following inequality

$$\langle w_2 - w_2^*, R(I - M)(w_2 - w_2^*) \rangle \leq \langle w_2 - w_2^*, R(E - M)(w_1 - w_1^*) \rangle, \quad (3.22)$$

where

$$R(I - M) = \begin{bmatrix} \frac{1}{\alpha_1} I_{m_1} & \theta A_1 A_2^\top & 0 \\ \theta A_2 A_1^\top & \frac{1}{\alpha_2} I_{m_2} & 0 \\ A_1^\top & A_2^\top & \frac{1}{\gamma} I_n \end{bmatrix}$$

and

$$R(E - M) = \begin{bmatrix} \frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top & -(\beta - \theta) A_1 A_2^\top & A_1 \\ -(\beta - \theta) A_2 A_1^\top & \frac{1}{\alpha_2} I_{m_2} - \beta A_2 A_2^\top & A_2 \\ 0 & 0 & \frac{1}{\gamma} I_n \end{bmatrix}.$$

Due to the fact that $R(E - I)$ is not skew-symmetric, it is impossible to reformulate the matrix $R(I - M)$ on the left-hand side of inequality (3.22) to be the same as the matrix $R(E - M)$ on the right-hand side. Then Proposition 2.14 is not applicable to the case with $\beta > 0$ and the operator \mathcal{L}_M in this case may not be firmly nonexpansive. However, we can prove a weaker property than the firmly nonexpansive property, which is that the operator \mathcal{L}_M is nonexpansive. The nonexpansive property is exactly Property 3 required under the implicit fixed-point proximity framework, and it is sufficient to yield the convergence of the proposed algorithm.

Next, we rewrite the inequality (3.22) by clearing all the off-diagonal matrix blocks in $R(I - M)$ and $R(E - M)$ gradually with the help of the following relations.

Lemma 3.9 Suppose that M is defined as (3.15) with $\beta > 0$ and $\rho = \gamma$, and that \mathcal{L}_M is an M -operator with $\text{prox}_{F,R} \circ E$. Let w_i and w_i^* be defined as (3.21), $i = 1, 2$, such that $w_2 = \mathcal{L}_M(w_1)$ and $w_2^* = \mathcal{L}_M(w_1^*)$. Then

$$x_1 - x_1^* = x_2 - x_2^* + \gamma(A_1^\top(u_2 - u_2^*) + A_2^\top(v_2 - v_2^*)). \quad (3.23)$$

Proof. It is derived from the third equation of equations (3.16) by setting $\rho = \gamma$ that

$$x_2 = x_1 - \gamma(A_1^\top u_2 + A_2^\top v_2)$$

and

$$x_2^* = x_1^* - \gamma(A_1^\top u_2^* + A_2^\top v_2^*).$$

Then equation (3.23) is immediately achieved by subtracting the second equation from the first equation. □

To rewrite the inequality (3.22), we start with clearing the matrix blocks with A_1 or A_2 in $R(E - M)$ on the right-hand side and the matrix blocks with A_1^\top or A_2^\top in $R(I - M)$ on the left-hand side. Then we obtain a new inequality as shown in the following lemma.

Lemma 3.10 Suppose that M is defined as (3.15) with $\beta > 0$ and $\rho = \gamma$, and that \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$. Let w_i and w_i^* be defined as (3.21), $i = 1, 2$, such that $w_2 = \mathcal{L}_M(w_1)$ and $w_2^* = \mathcal{L}_M(w_1^*)$. Then we have

$$\langle w_2 - w_2^*, G_1(w_2 - w_2^*) \rangle \leq \langle w_2 - w_2^*, G_2(w_1 - w_1^*) \rangle, \quad (3.24)$$

where

$$G_1 = \begin{bmatrix} \frac{1}{\alpha_1} I_{m_2} - \gamma A_1 A_1^\top & (\theta - \gamma) A_1 A_2^\top & 0 \\ (\theta - \gamma) A_2 A_1^\top & \frac{1}{\alpha_2} I_{m_2} - \gamma A_2 A_2^\top & 0 \\ 0 & 0 & \frac{1}{\gamma} I_n \end{bmatrix}$$

and

$$G_2 = \begin{bmatrix} \frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top & -(\beta - \theta) A_1 A_2^\top & 0 \\ -(\beta - \theta) A_2 A_1^\top & \frac{1}{\alpha_2} I_{m_2} - \beta A_2 A_2^\top & 0 \\ 0 & 0 & \frac{1}{\gamma} I_n \end{bmatrix}.$$

Proof. According to the right-hand side of the inequality (3.22), the inner product associated with the matrix blocks with A_1 or A_2 in $R(E - M)$ is

$$\begin{aligned} & \langle w_2 - w_2^*, [A_1^\top \ A_2^\top \ 0]^\top [0 \ 0 \ I_n] (w_1 - w_1^*) \rangle \\ &= \langle u_2 - u_2^*, A_1 (x_1 - x_1^*) \rangle + \langle v_2 - v_2^*, A_2 (x_1 - x_1^*) \rangle \\ &= \langle A_1^\top (u_2 - u_2^*) + A_2^\top (v_2 - v_2^*), x_1 - x_1^* \rangle. \end{aligned}$$

By substituting equation (3.23) into the inner product, we have

$$\begin{aligned} & \langle A_1^\top (u_2 - u_2^*) + A_2^\top (v_2 - v_2^*), x_1 - x_1^* \rangle \\ &= \langle A_1^\top (u_2 - u_2^*) + A_2^\top (v_2 - v_2^*), x_2 - x_2^* \rangle \\ & \quad + \gamma \langle A_1^\top (u_2 - u_2^*) + A_2^\top (v_2 - v_2^*), A_1^\top (u_2 - u_2^*) + A_2^\top (v_2 - v_2^*) \rangle \\ &= \langle w_2 - w_2^*, [A_1^\top \ A_2^\top \ 0]^\top [0 \ 0 \ I_n] (w_2 - w_2^*) \rangle \\ & \quad + \gamma \langle w_2 - w_2^*, [A_1^\top \ A_2^\top \ 0]^\top [A_1^\top \ A_2^\top \ 0] (w_2 - w_2^*) \rangle. \end{aligned}$$

Then the inequality can be written as inequality (3.24) with

$$R(E - M) - [A_1^\top \ A_2^\top \ 0]^\top [0 \ 0 \ I_n] = G_2$$

and

$$R(I - M) - [A_1^\top \ A_2^\top \ 0]^\top [0 \ 0 \ I_n] - \gamma [A_1^\top \ A_2^\top \ 0]^\top [A_1^\top \ A_2^\top \ 0] = G_1.$$

□

We continue to clear the off-diagonal matrix blocks of G_1 and G_2 by using the following lemmas.

Lemma 3.11 Let $u \in \mathbb{R}^{m_1}$, $v \in \mathbb{R}^{m_2}$, $A_1 \in \mathbb{R}^{m_1 \times n}$ and $A_2 \in \mathbb{R}^{m_2 \times n}$. Then

$$2\langle A_2^\top v, A_1^\top u \rangle = \|A_1^\top u + A_2^\top v\|_2^2 - \|A_1^\top u\|_2^2 - \|A_2^\top v\|_2^2.$$

Lemma 3.12 Let $w \in \mathbb{R}^d$, $w' \in \mathbb{R}^d$, and $A \in \mathbb{S}_+^d$. Then

$$2\langle w, Aw' \rangle = \langle w, Aw \rangle + \langle w', Aw' \rangle - \langle w - w', A(w - w') \rangle.$$

The resulting inequality with block diagonal matrices is shown in the following proposition. We denote $\text{diag}(D_1, \dots, D_n)$ as a block diagonal matrix whose diagonal blocks starting in the upper left corner are square matrices D_1, \dots, D_n .

Proposition 3.13 Suppose that M is defined as (3.15) with $\beta > 0$ and $\rho = \gamma$, and that \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$. Let w_i and w_i^* be defined as (3.21), $i = 1, 2$, such that $w_2 = \mathcal{L}_M(w_1)$ and $w_2^* = \mathcal{L}_M(w_1^*)$. If $\frac{1}{\alpha_1}I_{m_1} - \beta A_1 A_1^\top \in \mathbb{S}_+^{m_1}$ and $\frac{1}{\alpha_2}I_{m_2} - \beta A_2 A_2^\top \in \mathbb{S}_+^{m_2}$, then we have

$$\begin{aligned} \langle w_2 - w_2^*, H_1(w_2 - w_2^*) \rangle &\leq \langle w_1 - w_1^*, H_1(w_1 - w_1^*) \rangle \\ &\quad - \langle (v_2 - v_2^*) - (v_1 - v_1^*), H_2((v_2 - v_2^*) - (v_1 - v_1^*)) \rangle \\ &\quad - (\beta - \theta) \|A_1^\top(u_1 - u_1^*) + A_2^\top(v_2 - v_2^*)\|_2^2 \\ &\quad - (\beta - \theta) \|A_1^\top(u_2 - u_2^*) + A_2^\top(v_1 - v_1^*)\|_2^2 \\ &\quad - (2\theta - \gamma) \|A_1^\top(u_2 - u_2^*) + A_2^\top(v_2 - v_2^*)\|_2^2, \end{aligned} \tag{3.25}$$

where

$$H_1 = \text{diag} \left(\frac{1}{\alpha_1} I_{m_1} - \theta A_1 A_1^\top, \frac{1}{\alpha_2} I_{m_2} - \theta A_2 A_2^\top, \frac{1}{\gamma} I_n \right)$$

and

$$H_2 = \text{diag} \left(\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top, \frac{1}{\alpha_2} I_{m_2} - \beta A_2 A_2^\top, 0 \right).$$

Proof. We aim to reformulate the following inequality into the inequality (3.25)

$$2\langle w_2 - w_2^*, G_1(w_2 - w_2^*) \rangle \leq 2\langle w_2 - w_2^*, G_2(w_1 - w_1^*) \rangle. \quad (3.26)$$

First, we consider the inner product on the right-hand side of the inequality (3.26). It can be written as a sum of inner products associated with the matrix blocks in G_2

$$\begin{aligned} & 2\langle w_2 - w_2^*, G_2(w_1 - w_1^*) \rangle \\ &= 2\langle u_2 - u_2^*, (\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top)(u_1 - u_1^*) \rangle + 2\langle v_2 - v_2^*, (\frac{1}{\alpha_2} I_{m_2} - \beta A_2 A_2^\top)(v_1 - v_1^*) \rangle \\ &\quad - 2(\beta - \theta)\langle u_2 - u_2^*, A_1 A_2^\top(v_1 - v_1^*) \rangle - 2(\beta - \theta)\langle v_2 - v_2^*, A_2 A_1^\top(u_1 - u_1^*) \rangle \\ &\quad + \frac{2}{\gamma}\langle x_2 - x_2^*, x_1 - x_1^* \rangle. \end{aligned}$$

Next, we rewrite each of the inner products. By applying Lemma 3.12, if $\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top \in \mathbb{S}_+^{m_1}$ and $\frac{1}{\alpha_2} I_{m_2} - \beta A_2 A_2^\top \in \mathbb{S}_+^{m_2}$, then we have

$$\begin{aligned} & 2\langle u_2 - u_2^*, (\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top)(u_1 - u_1^*) \rangle \\ &= \langle u_2 - u_2^*, (\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top)(u_2 - u_2^*) \rangle + \langle u_1 - u_1^*, (\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top)(u_1 - u_1^*) \rangle \\ &\quad - \langle (u_2 - u_2^*) - (u_1 - u_1^*), (\frac{1}{\alpha_1} I_{m_1} - \beta A_1 A_1^\top)[(u_2 - u_2^*) - (u_1 - u_1^*)] \rangle \end{aligned}$$

and

$$\begin{aligned}
& 2\langle v_2 - v_2^*, (\frac{1}{\alpha_2}I_{m_2} - \beta A_2 A_2^\top)(v_1 - v_1^*) \rangle \\
&= \langle v_2 - v_2^*, (\frac{1}{\alpha_2}I_{m_2} - \beta A_2 A_2^\top)(v_2 - v_2^*) \rangle + \langle v_1 - v_1^*, (\frac{1}{\alpha_2}I_{m_2} - \beta A_2 A_2^\top)(v_1 - v_1^*) \rangle \\
&\quad - \langle (v_2 - v_2^*) - (v_1 - v_1^*), (\frac{1}{\alpha_2}I_{m_2} - \beta A_2 A_2^\top)[(v_2 - v_2^*) - (v_1 - v_1^*)] \rangle.
\end{aligned}$$

By applying Lemma 3.11, we have

$$\begin{aligned}
& -2(\beta - \theta)\langle u_2 - u_2^*, A_1 A_2^\top(v_1 - v_1^*) \rangle \\
&= (\beta - \theta)\|A_1^\top(u_2 - u_2^*)\|_2^2 + (\beta - \theta)\|A_2^\top(v_1 - v_1^*)\|_2^2 - (\beta - \theta)\|A_1^\top(u_2 - u_2^*) + A_2^\top(v_1 - v_1^*)\|_2^2
\end{aligned}$$

and

$$\begin{aligned}
& -2(\beta - \theta)\langle v_2 - v_2^*, A_2 A_1^\top(u_1 - u_1^*) \rangle \\
&= (\beta - \theta)\|A_1^\top(u_1 - u_1^*)\|_2^2 + (\beta - \theta)\|A_2^\top(v_2 - v_2^*)\|_2^2 - (\beta - \theta)\|A_1^\top(u_1 - u_1^*) + A_2^\top(v_2 - v_2^*)\|_2^2.
\end{aligned}$$

By applying Lemma 3.12 and then equation (3.23), we have

$$\begin{aligned}
& \frac{2}{\gamma}\langle x_2 - x_2^*, x_1 - x_1^* \rangle \\
&= \frac{1}{\gamma}\|x_2 - x_2^*\|_2^2 + \frac{1}{\gamma}\|x_1 - x_1^*\|_2^2 - \frac{1}{\gamma}\|(x_2 - x_2^*) - (x_1 - x_1^*)\|_2^2 \\
&= \frac{1}{\gamma}\|x_2 - x_2^*\|_2^2 + \frac{1}{\gamma}\|x_1 - x_1^*\|_2^2 - \gamma\|A_1^\top(u_2 - u_2^*) + A_2^\top(v_2 - v_2^*)\|_2^2.
\end{aligned}$$

Second, we consider the inner product on the left-hand side of the inequality (3.26). It

can be written as a sum of inner products associated with the matrix blocks in G_1 .

$$\begin{aligned}
& 2\langle w_2 - w_2^*, G_1(w_2 - w_2^*) \rangle \\
&= 2\langle u_2 - u_2^*, (\frac{1}{\alpha_1}I_{m_1} - \gamma A_1 A_1^\top)(u_2 - u_2^*) \rangle + 2\langle v_2 - v_2^*, (\frac{1}{\alpha_2}I_{m_2} - \gamma A_2 A_2^\top)(v_2 - v_2^*) \rangle \\
&\quad + 4(\theta - \gamma)\langle u_2 - u_2^*, A_1 A_2^\top(v_2 - v_2^*) \rangle + \frac{2}{\gamma}\|x_2 - x_2^*\|_2^2.
\end{aligned}$$

By applying Lemma 3.11, we have

$$\begin{aligned}
& 4(\theta - \gamma)\langle u_2 - u_2^*, A_1 A_2^\top(v_2 - v_2^*) \rangle \\
&= 2(\theta - \gamma)\|A_1^\top(u_2 - u_2^*) + A_2^\top(v_2 - v_2^*)\|_2^2 - 2(\theta - \gamma)\|A_1^\top(u_2 - u_2^*)\|_2^2 - 2(\theta - \gamma)\|A_2^\top(v_2 - v_2^*)\|_2^2.
\end{aligned}$$

Lastly, we combine all the inner products to reformulate the inequality, and obtain equation (3.25). □

Our goal is to ensure that \mathcal{L}_M is nonexpansive. And this can be achieved by setting $H_1 \in \mathbb{S}_+^{m_1+m_2+n}$, $H_2 \in \mathbb{S}^{m_1+m_2+n}$, $\beta - \theta \geq 0$ and $2\theta - \gamma \geq 0$, in addition to the assumptions in Proposition 3.13. In summery, the constraints are $\alpha_1\theta\|A_1\|_2^2 < 1$, $\alpha_2\theta\|A_2\|_2^2 < 1$, $\alpha_1\beta\|A_1\|_2^2 < 1$, $\alpha_2\beta\|A_2\|_2^2 < 1$, $\gamma \in (0, 2\theta]$ and $\theta \leq \beta$. In particular, Assumption (A3.3) can imply those constraints and yield that \mathcal{L}_M is nonexpansive.

Corollary 3.14 Suppose that M is defined as (3.15) and that \mathcal{L}_M is an M -operator associated with $\text{prox}_{F,R} \circ E$. If Assumption (A3.3) is satisfied, then \mathcal{L}_M is nonexpansive with respect to $H_3 \in \mathbb{S}_+^{m_1+m_2+n}$ where

$$H_3 = \text{Diag} \left(\frac{1}{\alpha_1}I_{m_1} - \beta A_1 A_1^\top, \frac{1}{\alpha_2}I_{m_2} - \beta A_2 A_2^\top, \frac{1}{\gamma}I_n \right).$$

Property 3 of the operator \mathcal{L}_M is verified by Corollary 3.14, and then the convergence of Algorithm 3 for the case when $\beta > 0$ is guaranteed as shown in the following theorem.

Theorem 3.15 Assume that $\text{Fix}(\text{prox}_{F,R} \circ E) \neq \emptyset$. Let M be defined as (3.15), and let $\lambda_k \in [0, 1]$ such that $\sum_k \lambda_k(1 - \lambda_k) = +\infty$. Then the sequence $\{w^k\}$, generated by Algorithm 3, converges to a solution of problem (P3) if Assumption (A3.1) and (A3.3) are satisfied, and the errors from inner iterations are summable.

Proof. The result immediately follows from Proposition 3.6, Corollary 3.14 and Theorem 2.20. □

3.3 Other Examples

After exploring the optimization problems with two terms, we are interesting in developing implicit algorithms for optimization problems with three or more terms such as the following models,

$$\min_{x \in \mathbb{R}^n} f_1(x) + g_1(A_1x) + g_2(A_2x) \quad (\text{P4})$$

and

$$\min_{x \in \mathbb{R}^n} f_1(x) + f_2(x) + g_1(A_1x), \quad (\text{P5})$$

where A_i is an $m_i \times n$ matrix, and $f_i : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ and $g_i : \mathbb{R}^{m_i} \rightarrow (-\infty, +\infty]$ is proper, lower semi-continuous, and convex, $i = 1, 2$.

The optimization problems with three or more terms have many applications in image reconstruction and statistics. For example, the ℓ_1 regularized fused lasso [79] and the MR image reconstruction model (M5) mentioned in Chapter 1, if Φ is invertible, have the form of problem (P4); the CT reconstruction model [53] has the form of problem (P5).

3.3.1 Fixed point characterization

Both problem (P4) and problem (P5) can be viewed as a special case of problem (P2) and their solutions can be characterized as the solutions of a system of three fixed point

equations.

In the following, we will use problem (P4) as an example to illustrate the procedure of developing implicit algorithms. This procedure can be adapted accordingly for problem (P5).

By applying Proposition 3.1, the solutions of problem (P4) can be characterized as the solutions of the following system of fixed point equations.

$$\begin{cases} u = \text{prox}_{\alpha_1 g_1^*}(u + \alpha_1 A_1 x) \\ v = \text{prox}_{\alpha_2 g_2^*}(v + \alpha_2 A_2 x) \\ x = \text{prox}_{\beta f_1}(x - \beta(A_1^\top u + A_2^\top v)), \end{cases} \quad (3.27)$$

where $\alpha_1 > 0$, $\alpha_2 > 0$ and $\beta > 0$.

Note that for problem (P4) we do not consider another equivalent variant of equations (3.27). Because the third equation with the proximity operator of f_1 may not be linear. If we substitute the third equation with x into other equations, then the resulting equivalent variant cannot be written as the unified fixed point equation (2.12), and therefore we cannot continue to develop fixed-point proximity algorithms.

Instead, we consider the system of fixed point equations (3.27) directly. Under the implicit fixed-point proximity framework, the system is written as the unified fixed point equation (2.12) with $F : \mathbb{R}^{m_1+m_2+n} \rightarrow \mathbb{R}$ defined by $F(w) = g_1^*(u) + g_2^*(v) + f_1(x)$, $R = \text{diag}\left(\frac{1}{\alpha_1}I_{m_1}, \frac{1}{\alpha_2}I_{m_2}, \frac{1}{\beta}I_n\right)$, $\alpha_1 > 0, \alpha_2 > 0, \beta > 0$, and

$$w = \begin{bmatrix} u \\ v \\ x \end{bmatrix} \in \mathbb{R}^{m_1+m_2+n} \quad \text{and} \quad E = \begin{bmatrix} I_{m_1} & & \alpha_1 A_1 \\ & I_{m_2} & \alpha_2 A_2 \\ -\beta A_1^\top & -\beta A_2^\top & I_n \end{bmatrix} \in \mathbb{R}^{(m_1+m_2+n) \times (m_1+m_2+n)}. \quad (3.28)$$

Then a fixed-point proximity algorithm, developed from this unified fixed point equation,

generates a sequence $\{w^k\}$ that follows equation (2.17) with function F and matrices E and R defined as above.

3.3.2 Implicit fixed-point proximity algorithms

To develop implicit fixed-point proximity algorithms for problem (P4), we have two possible block structures for the 3×3 block matrix M .

The first block structure is the structure with skew diagonal blocks defined as follows

$$M_1 = \begin{bmatrix} 0 & 0 & (-1 - \theta)\alpha_1 A_1^\top \\ 0 & 0 & (-1 - \theta)\alpha_2 A_2^\top \\ (1 - \theta)\beta A_1 & (1 - \theta)\beta A_2 & 0 \end{bmatrix} \quad (3.29)$$

and $R(E - M_1)$ is computed as follows

$$R(E - M_1) = \begin{bmatrix} \frac{1}{\alpha_1} I_{m_1} & 0 & \theta A_1^\top \\ 0 & \frac{1}{\alpha_2} I_{m_2} & \theta A_2^\top \\ \theta A_1 & \theta A_2 & \frac{1}{\beta} I_n \end{bmatrix}.$$

The corresponding implicit algorithm is similar to Algorithm 2, and has the following convergence assumptions.

Assumption (A4.1) $\beta |1 - \theta^2| \left\| \begin{bmatrix} A_1 & A_2 \end{bmatrix} \right\|_2 \left\| \begin{bmatrix} \alpha_1 A_1 & \alpha_2 A_2 \end{bmatrix} \right\|_2 < 1$

Assumption (A4.2) $\beta \theta^2 \left\| \begin{bmatrix} \sqrt{\alpha_1} A_1 & \sqrt{\alpha_2} A_2 \end{bmatrix} \right\|_2 < 1$

These two assumptions are corresponding to the three assumptions in Algorithm 2. Assumption (A2.1) in Algorithm 2 can be interpreted as Assumption (A4.1); Assumption (A2.2) is already satisfied; and Assumption (A2.3) can be interpreted as Assumption (A4.2). Thus, the proposed implicit fixed-point proximity algorithm converges if Assumption (A4.1) and (A4.2) are satisfied.

The second block structure defined as follows has the same structure as the matrix M (3.15) in Example 2,

$$M_2 = \begin{bmatrix} 0 & -\alpha_1\theta A_1 A_2^\top & 0 \\ -\alpha_2\theta A_2 A_1^\top & 0 & 0 \\ -2\beta A_1^\top & -2\beta A_2^\top & 0 \end{bmatrix}, \quad (3.30)$$

where $\theta > 0$ and $R(E - M_2)$ is computed as follows

$$R(E - M_2) = \begin{bmatrix} \frac{1}{\alpha_1} I_{m_1} & \theta A_1 A_2^\top & A_1^\top \\ \theta A_2 A_1^\top & \frac{1}{\alpha_2} I_{m_2} & A_2^\top \\ A_1^\top & A_2^\top & \frac{1}{\beta} I_n \end{bmatrix}.$$

The corresponding implicit algorithm is similar to Algorithm 3 when $\beta = 0$, and has the following convergence assumptions.

Assumption (A4.3) $\alpha_1\alpha_2\theta^2\|A_1\|_2^2\|A_2\|_2^2 < 1$

Assumption (A4.4) $\theta = \beta$, $\alpha_1\beta\|A_1\|_2^2 < 1$ and $\alpha_2\beta\|A_1\|_2^2 < 1$

These two assumptions are corresponding to the two assumptions in Algorithm 3 with $\beta = 0$. Assumption (A3.1) in Algorithm 3 can be interpreted as Assumption (A4.3), and Assumption (A3.2) can be interpreted as Assumption (A4.4). Thus, the proposed implicit fixed-point proximity algorithm converges if Assumption (A4.3) and (A4.4) are satisfied.

Note that we only present the procedure of proposing implicit fixed-point proximity algorithms for problem (P4) and will not conduct numerical experiments to demonstrate the performance. Because those algorithms have similar algorithmic structures of the proposed algorithms in Example 1 and Example 2, as well as similar practical performance.

Chapter 4

Applications in Image Processing and Numerical Experiments

In this chapter, we apply the proposed fixed-point proximity algorithms to applications in image processing and test those implicit algorithms on four image reconstruction models. Two total variation based denoising models have the form of problem (P2) and two total variation based deblurring models have the form of problem (P3). The numerical experiment results demonstrate the performance of the proposed implicit fixed-point proximity algorithms in comparison with other existing explicit fixed-point proximity algorithms.

All algorithms were implemented in Matlab 2017a and executed on an Intel Core i5 CPU at 2.9 GHz, 8G RAM, running a 64 Bit Window 10 system.

4.1 Applications in Image Processing

The image reconstruction models aim to recover the underlying image from the observed and possibly degraded image, and those models can be identified as problem (P0), which minimizes a sum of convex functions composed with or without linear operators.

4.1.1 Image reconstruction models

Suppose $z \in \mathbb{R}^m$ represents the observed image degraded by blur or/and noise and $x \in \mathbb{R}^n$ represents the desired image to be constructed. The relationship between the observed image z and the original image x can be modeled as follows

$$z = Kx + \eta,$$

where $K \in \mathbb{R}^{m \times n}$ represents the measurement process, and $\eta \in \mathbb{R}^m$ represents the unknown additive noise. The matrix K varies in different image reconstruction problems. In denoising problems, K is the identity matrix; in deblurring problems, K represents the convolving process with a blurring kernel; and in compressed sensing, K represents the sparse sampling process. For the noise vector η , two types of noise, namely, Gaussian noise and impulsive noise, will be tested.

Depending on the matrix K and the noise η , there are various image construction models designed to recover the image x as introduced at the beginning of Chapter 1. An image construction model usually consists of one data fidelity term and at least one regularization term. In this dissertation, we mainly focus on the image reconstruction models with total variation regularization. Next, we introduce the formulation of the total variation regularization.

4.1.2 Total variation regularization

The total variation semi-norm measures the total variation between nearby pixels in an image and the sharp edges of the image can be preserved by minimizing its total variation.

Suppose that $x \in \mathbb{R}^n$ is an image vector of a square image with size $\sqrt{n} \times \sqrt{n}$. Then the TV semi-norm of x , which measures the total variation of image pixels, is formulated as follows.

$$\|x\|_{TV} = \psi(Dx),$$

where $\psi : \mathbb{R}^{2n} \rightarrow \mathbb{R}$ is defined by

$$\psi(y) = \sum_{i=1}^n \left\| \begin{bmatrix} y_i & y_{n+i} \end{bmatrix}^\top \right\|_2 \quad (4.1)$$

and $D \in \mathbb{R}^{2n \times n}$ is a first order difference matrix defined as follows in terms of the kronecker product operator \otimes

$$D = \begin{bmatrix} I_d \otimes D_d \\ D_d \otimes I_d \end{bmatrix} \quad (4.2)$$

with $d = \sqrt{n}$. Here D_d is a $d \times d$ matrix given in the following

$$D_d = \begin{bmatrix} -1 & 1 & & & \\ & \ddots & \ddots & & \\ & & -1 & 1 & \\ & & & & 0 \end{bmatrix}.$$

We know that $\|D\|_2 \leq \sqrt{8}$.

Now, we are ready to apply the proposed implicit fixed-point proximity algorithms to the total variation based image reconstruction models and demonstrate the performance of those implicit algorithms.

4.2 Application of Example 1: Total Variation Based Denoising Models

In this section, we consider two total variation based denoising models. They are the ROF model defined as model (M1) and the L1-TV denoising model defined as model (M2). Both models have the form of problem (P2) where f_1 is corresponding to the data fidelity term, and $f_2 \circ B = \psi \circ D = \|\cdot\|_{TV}$ is corresponding to the total variation regularization

term.

$$\min_{x \in \mathbb{R}^n} f_1(x) + \psi(Dx).$$

To remove Gaussian noise, we choose $f_1 = \frac{\lambda}{2} \|\cdot - z\|_2^2$, and the resulting optimization problem refers to as the ROF model. To remove impulsive noise, we choose $f_1 = \lambda \|\cdot - z\|_1$, the resulting optimization problem refers to as the L1-TV denoising model.

Before we apply Algorithm 2 to the total variation based denoising models, we present the parameter settings for three algorithms: fixed point algorithm based on the proximity operator for ROF model (FP) as (2.23), primal dual algorithm (PD) as (2.21), and implicit fixed-point proximity algorithm (IM) as Algorithm 2.

4.2.1 Parameter settings

For performance evaluations, we use the following fixed point proximity algorithms and parameter settings.

- FP: fixed point algorithm based on the proximity operator for ROF model as (2.23),

with

$$M = \begin{bmatrix} I & 0 \\ \beta D & 0 \end{bmatrix}, \alpha = 1, \text{ and } \beta \|D\|_2^2 = 1.999.$$

- PD: primal dual algorithm as (2.21), with

$$M = \begin{bmatrix} 0 & 0 \\ 2\beta D & 0 \end{bmatrix}, \alpha = 0.01, \text{ and } \alpha\beta \|D\|_2^2 = 0.999.$$

- IM: implicit fixed-point proximity algorithm as Algorithm 2, with

$$M = \begin{bmatrix} 0 & (-1 - \theta)\alpha D^\top \\ (1 - \theta)\beta D & 0 \end{bmatrix}, \alpha = 0.01, \alpha\beta \|D\|_2^2 = 1.999, \theta = \frac{1}{\sqrt{2}}, \lambda_k = 1, \\ x_0^{k+1} = 0.5x^k + 0.5x^{k-1}, \text{ and 1 iteration in the inner loop.}$$

In particular, we need the proximity operators of f_1 and f_2^* which have closed forms as below.

- $\text{prox}_{\alpha \frac{\lambda}{2} \|\cdot - z\|_2^2}(x) = \frac{\alpha \lambda}{\alpha \lambda + 1} z + \frac{1}{\alpha \lambda + 1} x$
- $\tilde{x} = \text{prox}_{\alpha \lambda \|\cdot - z\|_1}(x)$, where $\tilde{x}_i = z_i + \max \left\{ 0, 1 - \frac{\alpha \lambda}{|x_i - z_i|} \right\} (x_i - z_i)$
- $\tilde{y} = \text{prox}_{\beta \psi^*}(y)$, where $[\tilde{y}_i, \tilde{y}_{n+i}]^\top = \frac{[y_i, y_{n+i}]^\top}{\max\{\|[y_i, y_{n+i}]^\top\|_2, 1\}}$
- $\tilde{y} = \text{prox}_{\beta(\frac{1}{\lambda} \psi)^*}(y)$, where $[\tilde{y}_i, \tilde{y}_{n+i}]^\top = \frac{[y_i, y_{n+i}]^\top}{\max\{\lambda \|[y_i, y_{n+i}]^\top\|_2, 1\}}$

4.2.2 ROF model

The image of “Girl” of size 256×256 in Figure 4.1(a) is used in our test. This image is contaminated by the additive zero-mean Gaussian noise with standard deviation of $\sigma = 0.05$ (see Figure 4.1(b)). The ROF model with regularization parameter of $\lambda = 16$ is used to remove this noise.

In the ROF model, $f_1 = \frac{\lambda}{2} \|\cdot - z\|_2^2$ is a strongly convex function, which implies that the solution of ROF is unique. So we use the following error measure to evaluate the performance of an algorithm

$$\varepsilon = \|x^k - x^*\|_2,$$

where x^* (see Figure 4.1(c)) is the reference solution generated by running primal dual algorithm for 100,000 iterations, and x^k is the k -th iteration from the algorithm.



(a) Original image

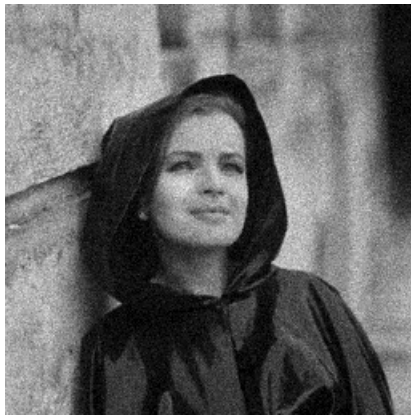
(b) Noisy image with $\sigma = 0.05$ (c) Reference solution with $\lambda = 16$

Figure 4.1: ROF image denoising model

Figure 4.2 depicts the errors of the solutions obtained by three testing algorithms (PD, FP, and IM) against the CPU time consumption. We can see that IM converges faster than PD and FP for the ROF model.

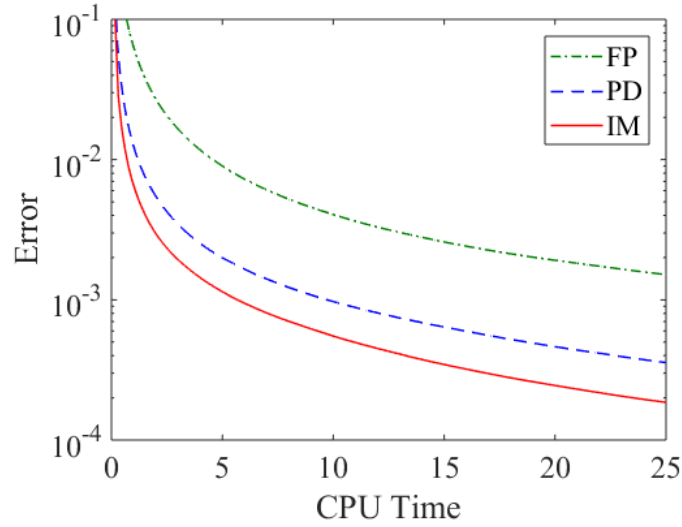


Figure 4.2: Performance evaluations of primal dual algorithm (PD), fixed point algorithm based on the proximity operator for ROF model (FP) and implicit fixed-point proximity algorithm as Algorithm 2 (IM)

To further evaluate our algorithm, four different levels of noise are added to the test image “Girl” and two different error tolerances are tested. In our experiments, the parameter settings remain the same as above. Table 4.1 illustrates the number of iterations and CPU time needed to achieve the error tolerance. “-” indicates the algorithm did not achieve the error tolerance within 5000 iterations. We observe that IM performs better than FD and PD in terms of computational time.

		$\varepsilon = 10^{-2}$	$\varepsilon = 10^{-3}$
$\sigma = 0.05$ $\lambda = 16$	FP	1422 (4.57 s)	-
	PD	457 (1.20 s)	3723 (9.72 s)
	IM	229 (0.70 s)	1865 (5.68 s)
$\sigma = 0.08$ $\lambda = 14$	FP	1351 (4.60 s)	-
	PD	376 (1.00 s)	3166 (8.43 s)
	IM	188 (0.60 s)	1587 (5.08 s)
$\sigma = 0.1$ $\lambda = 10$	FP	2442 (8.18 s)	-
	PD	486 (1.25 s)	4015 (10.31 s)
	IM	240 (0.73 s)	2010 (6.11 s)
$\sigma = 0.15$ $\lambda = 7$	FP	3561 (11.34 s)	-
	PD	500 (1.28 s)	4970 (12.67 s)
	IM	249 (0.77 s)	2464 (7.56 s)

Table 4.1: Comparison between three fixed point proximity algorithms

4.2.3 L1-TV model

The image of “Boat” of size 512×512 in Figure 4.3(a) is used in our test. This image is contaminated by the additive impulsive noise with noise density of $d = 0.2$ (see Figure 4.3(b)). The L1-TV denoising model with regularization parameter of $\lambda = 1.05$ is used to remove this noise.

In the L1-TV model, $f_1 = \lambda \|\cdot - z\|_1$ is not a strongly convex function, which means the solution of L1-TV may not be unique. So we use the following error measure to evaluate the performance of an algorithm

$$\varepsilon = \frac{E(x^k) - E(x^*)}{E(x^*)}, \quad (4.3)$$

where x^* (see Figure 4.3(c)) is the reference solution generated by running primal dual algorithm for 50,000 iterations and $E(\cdot)$ is the objective function (or called energy function) of the L1-TV model.



(a) Original image

(b) Noisy image with $d = 0.2$ (c) Reference solution with $\lambda = 1.05$

Figure 4.3: L1-TV image denoising model

Figure 4.4 depicts the errors of the solutions obtained by PD and IM against the CPU time consumption. We can see that IM converges faster than PD for the L1-TV model. Note that the fixed point algorithm based on the proximity operator for ROF model (FP) is not applicable to L1-TV model.

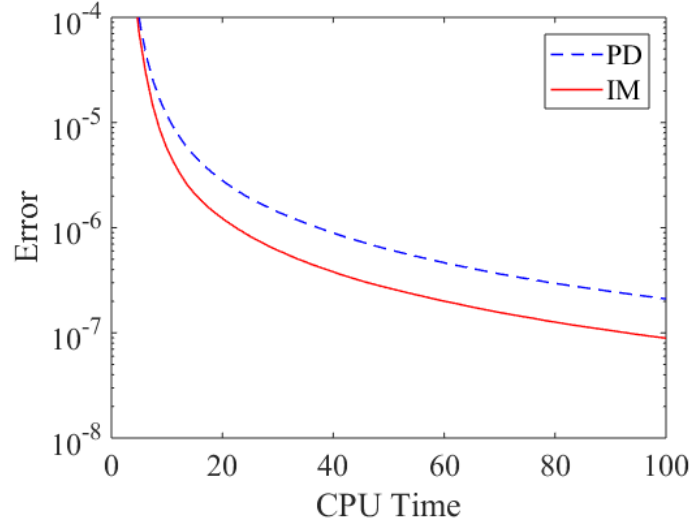


Figure 4.4: Performance evaluations of primal dual algorithm (PD) and implicit fixed-point proximity algorithm as Algorithm 2 (IM)

To further evaluate our algorithm, four different levels of noise are added to the test image “Boat” and two different error tolerances are tested. In our experiments, the parameter settings remain the same as above. Table 4.2 illustrates the number of iterations and CPU time needed to achieve the error tolerance. Again, we observe that IM performs better than PD in terms of computational time.

		$\varepsilon = 10^{-5}$	$\varepsilon = 10^{-6}$
$\sigma = 0.1$ $\lambda = 1.2$	PD	454 (9.84 s)	1491 (32.31 s)
	IM	307 (7.62 s)	793 (19.69 s)
$\sigma = 0.15$ $\lambda = 1.08$	PD	504 (10.67 s)	1721 (37.09 s)
	IM	338 (8.35 s)	904 (22.34 s)
$\sigma = 0.2$ $\lambda = 1.05$	PD	493 (10.68 s)	1721 (37.27 s)
	IM	333 (8.26 s)	903 (22.44 s)
$\sigma = 0.25$ $\lambda = 1.03$	PD	492 (10.68 s)	1661 (36.02 s)
	IM	352 (8.74 s)	896 (22.26 s)

Table 4.2: Comparison between two fixed-point proximity algorithms

4.2.4 Sensitivity of parameter settings

Algorithm 2 as an implicit fixed-point proximity algorithm is computed in a double-loop fashion. In the inner loop, we perform contractive mappings to compute $\mathcal{L}_M(w^k)$. In the outer loop, we utilize the outcome $\mathcal{L}_M(w^k)$ from the inner loop to compute the next step w^{k+1} . So the performance of Algorithm 2 is related to the performance of the inner loop and the parameters in the outer loop.

Next, we discuss the sensitivity of parameter settings in Algorithm 2 from the aspects of the inner loop and the outer loop.

(i) Parameter settings in the inner loop

As shown in Proposition 2.12, the performance of the inner loop via contractive mappings depends on three factors: the initial vector of the inner sequence, the number of inner iterations and the contraction constant. In particular, the inner sequence in Algorithm 2 should converge fast to the solution $\mathcal{L}_M(w^k)$ if the initial vector x_0^{k+1} is close to the solution, the number of inner iterations is large, and the contraction constant $\|M_1\|_2\|M_2\|_2$ is small.

• The initial vector and the number of inner iterations

We test the sensitivity of the algorithm to the initial vector and the number of inner iterations on the ROF model for the “Girl” image with noise level of $\sigma = 0.05$. In this experiment, we use the same parameter settings as mentioned in Section 4.2.1, except for the initial vector and the number of inner iterations. In order to let the initial vector get closer to the solution, we set x_0^{k+1} to be a combination of x^k and x^{k-1} , i.e., $x_0^{k+1} = \rho_k x^k + (1 - \rho_k) x^{k-1}$, where $\rho_k \in \mathbb{R}$. Then we test the sensitivity of the algorithm to the parameter ρ_k as well as the number of inner iterations.

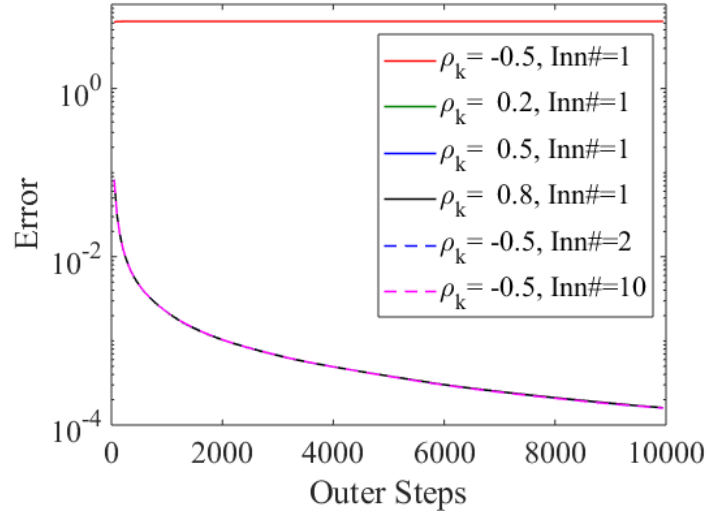


Figure 4.5: Sensitivity of ρ_k and the number of iterations (Inn#) in the inner loop

Note that there are only two curve patterns in Figure 4.5. The upper red curve represents the case when $\rho_k = -0.5$ and the number of inner iteration is 1, while the lower curve with overlapping colors represents all other cases.

It is demonstrated in Figure 4.5 that the performance of Algorithm 2 is not sensitive to the initial vector and the number of iterations in the inner loop as long as they are properly chosen. If we fix the number of inner iterations to be 1, then the error plots for different ρ_k coincide as long as $\rho_k \in (0, 1)$. If we pick the initial vector factor as $\rho_k = -0.5 \notin (0, 1)$, then the performance of the inner loop can still be guaranteed by increasing the number of inner iterations and there is no significant difference between choosing the number of inner iterations to be 2 or 10.

Therefore, it is not necessary to solve each inner loop entirely to a numerical precision. If we choose a proper initial vector factor, $\rho_k \in (0, 1)$, then only one or two inner iterations are sufficient to achieve solutions with the desired accuracy.

We also obtain similar results for the ROF model applied to the “Girl” image with different noise levels and the L1-TV model applied to the “Boat” image with different noise levels.

• **The contraction constant**

In Algorithm 2 for image denoising problems, the contraction constant $\|M_1\|_2\|M_2\|_2$ is equal to $\alpha\beta|1 - \theta^2|\|D\|_2^2$, which is associated with the choice of θ in the matrix M defined as (3.5) and the parameters of proximity operators, α and β . According to Proposition 2.22, if the contraction constant is small, then the inner sequence converges fast to the solution. However, in the parameter settings for denoising models, we choose a relatively large contraction constant $\|M_1\|_2\|M_2\|_2 = 0.999 < 1$ and the performance of the inner loop as demonstrated in Figure 4.5 is still satisfactory if we choose an appropriate initial vector or increase the number of inner iterations.

(ii) **Parameter settings in the outer loops**

The performance of the outer loop is relied on the operator \mathcal{L}_M , and the performance of the operator \mathcal{L}_M for Algorithm 2 is related to the parameter θ in the matrix M defined as (3.5) and the parameters, α and β , in the proximity operators. According to Theorem 3.4, the parameters α , β and θ should satisfy two conditions: $\alpha\beta\theta^2\|B\|_2^2 < 1$ and $\alpha\beta|1 - \theta^2|\|B\|_2^2 < 1$.

Next, we test the sensitivity of θ , α and β on the ROF model for the “Girl” image with noise level of $\sigma = 0.05$. In this experiment, to avoid any possible effect of the inner loop, we set $\rho_k = 0.5$ and 10 iterations in the inner loop. Also, we fix α to be 0.01, and, for comparison, we choose different θ and β such that $\alpha\beta\theta^2\|B\|_2^2 < 1$ and $\alpha\beta|1 - \theta^2|\|B\|_2^2 < 1$.

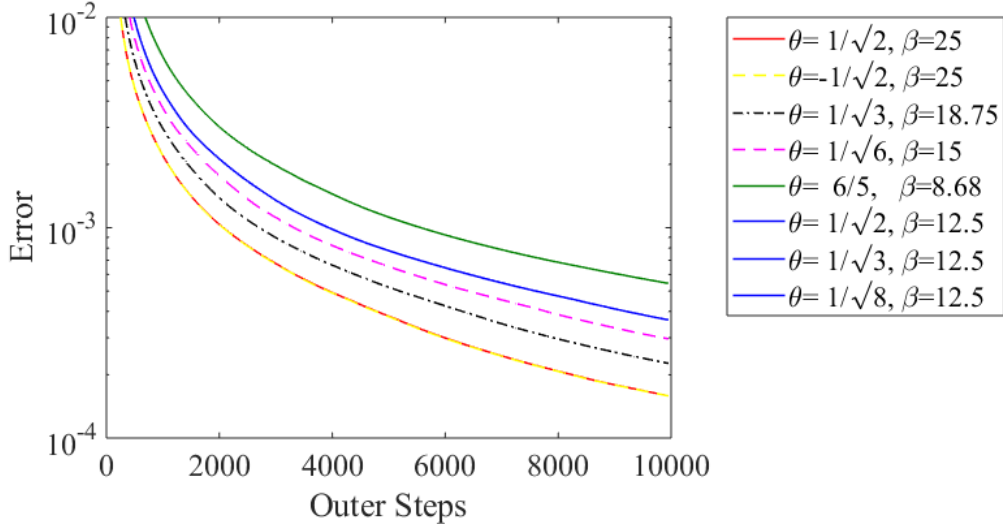


Figure 4.6: Sensitivity of parameters θ and β

Note that the overlapping blue curve represents the cases where $\beta = 12.5$ and θ varies and the overlapping curve with red and yellow colors represents the cases where $|\theta| = \frac{1}{\sqrt{2}}$ and $\beta = 25$.

As shown in Figure 4.6, we have two observations on the sensitivity of the parameters. First, if β is fixed, then the performance of the algorithm is not sensitive to θ . For example, if we fix $\beta = 12.5$, then there is no significant difference between the error plots with different θ . Second, if β varies as θ varies, then the performance improves as β gets larger. Particularly, in case where $|\theta| = \frac{1}{\sqrt{2}}$, β has the largest value and the corresponding algorithm performs the best. In fact, the largest possible value of the product $\alpha\beta$ such that $\alpha\beta\theta^2\|B\|_2^2 < 1$ and $\alpha\beta|1 - \theta^2|\|B\|_2^2 < 1$ is obtain at $|\theta| = \frac{1}{\sqrt{2}}$. The resulting convergence constraint becomes $\alpha\beta\|B\|_2^2 < 2$, which provides the widest selection range for α and β . This is a possible reason why Algorithm 2 outperforms the primal dual algorithm, whose convergence constraint is $\alpha\beta\|B\|_2^2 < 1$.

4.3 Application of Example 2: Total Variation Based Deblurring Models

In this section, we consider two total variation based deblurring models. They are the L2-TV deblurring model defined as model (M3) and the L1-TV deblurring model defined as model (M4). Both models have the form of problem (P3) where $f_1 \circ A_1 = f_1 \circ K$ is corresponding to the data fidelity term, and $f_2 \circ A_2 = \psi \circ D = \|\cdot\|_{TV}$ is corresponding to the total variation regularization term.

$$\min_{x \in \mathbb{R}^n} f_1(Kx) + \psi(Bx),$$

where K is the blurring matrix with $\|K\|_2 = 1$.

In the following experiments, we choose Gaussian blur with space-variant filters [63, 75] instead of space-invariant filters. The convolution with space-invariant filters can be computed by fast Fourier transforms (FFTs), while the convolution with space-variant filters is more difficult to compute and cannot be efficiently simplified in the Fourier domain in general [42, 77].

To remove Gaussian noise, we choose $f_1 = \frac{\lambda}{2} \|\cdot - z\|_2^2$, and the resulting optimization problem refers to as the L2-TV deblurring model. To remove impulsive noise, we choose $f_1 = \lambda \|\cdot - z\|_1$, and the resulting optimization problem refers to as the L1-TV deblurring model.

Before we apply Algorithm 3 to the total variation based deblurring models, we present the parameter settings for three algorithms: fixed-point proximity Gauss-Seidel algorithm (GS) as (3.13), implicit fixed-point proximity algorithm (IM) as Algorithm 3, and alternating direction method of multipliers [32] with conjugate gradient method [66] (ADMMCG).

4.3.1 Parameter settings

For performance evaluation, we use the following fixed point proximity algorithms and parameter settings.

- GS: fixed-point proximity Gauss-Seidel algorithm as (3.13), with

$$M = \begin{bmatrix} 0 & 0 & 0 \\ -\alpha_2\beta A_2 A_1^\top & 0 & 0 \\ -\gamma A_1^\top & -\gamma A_2^\top & 0 \end{bmatrix}, \beta = 0.01, \alpha_1\beta\|K\|_2^2 = 0.999, \alpha_2\beta\|B\|_2^2 = 0.999, \\ \gamma = \beta, \text{ and } \lambda_k \in (0, 1).$$

- IM: implicit fixed-point proximity algorithm as Algorithm 3, with

$$M = \begin{bmatrix} 0 & -\alpha_1\beta A_1 A_2^\top & 0 \\ -\alpha_2\beta A_2 A_1^\top & 0 & 0 \\ -\gamma A_1^\top & -\gamma A_2^\top & 0 \end{bmatrix}, \beta = 0.01, \alpha_1\beta\|K\|_2^2 = 0.999, \alpha_2\beta\|B\|_2^2 = 0.999, \\ \gamma = 2\beta, \lambda_k \in (0, 1), v_0^{k+1} = 0.5v^k + 0.5v^{k-1}, \text{ and 1 iteration in the inner loop.}$$

- ADMMCG: alternating direction method of multipliers with conjugate gradient method, with $\alpha = 10$.

In particular, we need the proximity operators of f_1^* and f_2^* . The proximity operator $\text{prox}_{\beta\psi^*}$ is provided in the previous section and $\text{prox}_{\alpha f_1^*}$ can be computed by using the proximity operator of f_1 as follows

$$\text{prox}_{\alpha f_1^*}(x) = x - \alpha \text{prox}_{\frac{1}{\alpha} f_1}\left(\frac{x}{\alpha}\right).$$

4.3.2 L2-TV model

The image of “Cameraman” of size 256×256 in Figure 4.7(a) is used in our test. This image is degraded by the space-variant Gaussian blur with filter size 17×17 , and standard deviation σ fixed to be 1 in the vertical direction and varying from 0.5 to 17 in the horizontal direction, and then the image is also contaminated by the additive Gaussian noise with

standard deviation of $\sigma = 0.02$ (see Figure 4.7(b)). The L2-TV deblurring model with regularization parameter of $\lambda = 8$ is used to recover the image.

In the L2-TV model, the solution may not be unique if K is not full rank. So we use the error measure defined in equation (4.3), where x^* (see Figure 4.7(c)) is the reference solution generated by running implicit fixed-point proximity algorithm for 10,000 iterations and $E(\cdot)$ is the objective function of the L2-TV model.



(a) Original image



(b) Blurry and noisy image



(c) Reference solution with $\lambda = 8$

Figure 4.7: L2-TV image deblurring model

Figure 4.8 depicts the normalized errors of the objective function values evaluated at the solutions obtained by GS, IM and ADMMCG against the CPU time consumption. We can see that IM converges faster than GS and ADMMCG for the L2-TV model.

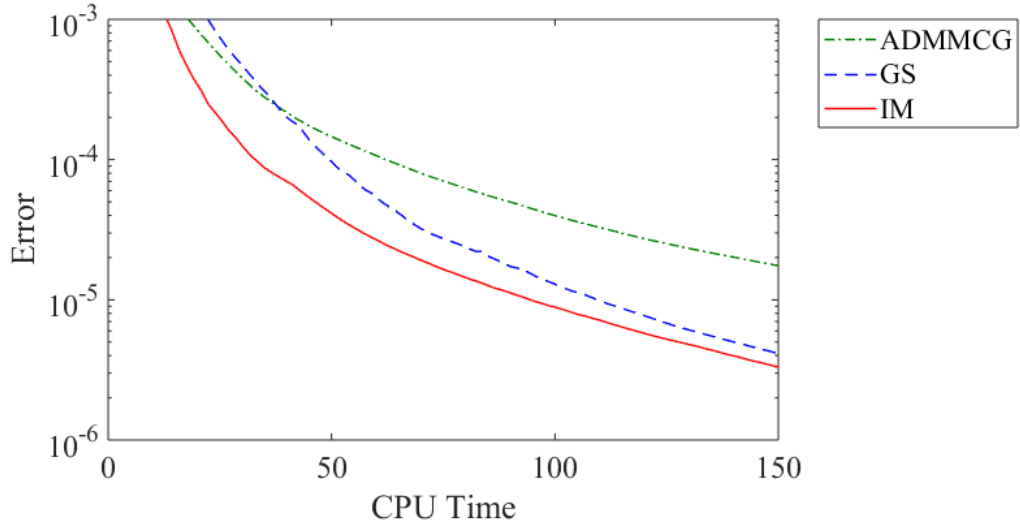


Figure 4.8: Performance evaluations of the fixed-point proximity Gauss-Seidel algorithm (GS), the implicit fixed-point proximity algorithm as Algorithm 3 (IM) and ADMMCG

4.3.3 L1-TV model

The image of “Barbara” of size 512×512 in Figure 4.9(a) is used in our test. This image is degraded by the space-variant Gaussian blur with filter size 21×21 , and standard deviation σ fixed to be 1 in the vertical direction and varying from 0.5 to 20 in the horizontal direction, and then the image is also contaminated by the additive impulsive noise with noise density of $d = 0.3$ (see Figure 4.9(b)). The L1-TV deblurring model with regularization parameter of $\lambda = 3$ is used to recover the image. Again, for performance evaluations, we use the error measure defined in equation (4.3), where x^* (see Figure 4.9(c)) is the reference solution generated by running implicit fixed-point proximity algorithm for 5,000 iterations and $E(\cdot)$ is the objective function of the L1-TV model.



(a) Original image



(b) Blurry and noisy image

(c) Reference solution with $\lambda = 3$

Figure 4.9: L1-TV image deblurring model

Figure 4.10 depicts the normalized errors of the objective function values evaluated at the solutions obtained by GS, IM and ADMMCG against the CPU time consumption. We can see that IM converges faster than GS and ADMMCG for the L1-TV model.

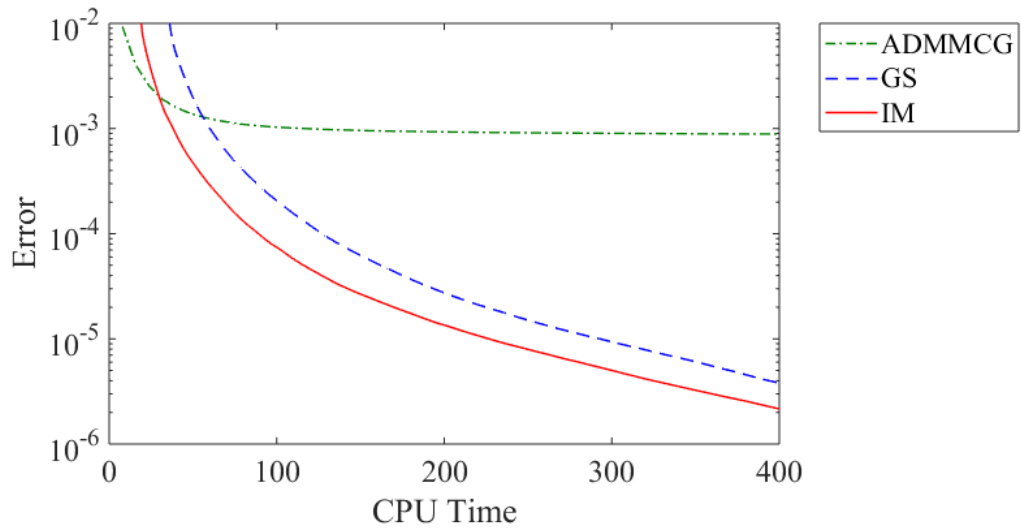


Figure 4.10: Performance evaluations of the fixed-point proximity Gauss-Seidel algorithm (GS), the implicit fixed-point proximity algorithm as Algorithm 3 (IM) and ADMMCG

Chapter 5

Conclusion and Future Work

The fixed point proximity framework is a powerful and efficient tool for developing algorithms for solving composite optimization problems. The framework converts the composite optimization into fixed point problems and develops iterative algorithms to solve the fixed point problems. Our proposed implicit fixed point proximity framework complements the existing framework that is designed for explicit iterative algorithms, by developing implicit iterative algorithms with theoretical results. The numerical experiments demonstrate that the algorithms with a fully implicit scheme do have a potential to overcome the restrictions in explicit algorithms, which encourages us to continue our study on implicit fixed-point proximity framework.

In the future, we would like to improve the implicit fixed-point proximity framework in the following directions.

- *Block structure of the matrix M* : In this dissertation, we proposed two block structures designed for developing implicit fixed-point proximity algorithms. We seek to construct other possible block structures of the matrix M that may yield to an implicit fixed-point proximity algorithm with a favorable convergence speed.
- *Composite optimization problems with three or more terms*: The implicit fixed point proximity framework, studied in this dissertation, is established based on a general com-

posite convex optimization problem and can be adapted to develop implicit fixed point proximity algorithms for optimization problems with different numbers of composite terms. The flexibility in developing implicit algorithms for the composite optimization problems with three or more terms motivates us to construct the matrix M creatively to overcome the complexity in the resulting implicit algorithms.

- *Convergence rate*: It is proved in this dissertation that the convergence rate of a fixed-point proximity algorithm is $O(\frac{1}{K})$, where K is the number of iterations. We shall aim to improve this convergence rate by exploring the underlying properties of the operator \mathcal{L}_M with a fully implicit expression.

Bibliography

- [1] R. P. AGARWAL, M. MEEHAN, AND D. O'REGAN, *Fixed point theory and applications*, vol. 141, Cambridge university press, 2001.
- [2] A. BECK AND M. TEBoulLE, *Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems*, IEEE Transactions on Image Processing, 18 (2009), pp. 2419–2434.
- [3] S. BOYD, N. PARIKH, E. CHU, B. PELEATO, AND J. ECKSTEIN, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends® in Machine Learning, 3 (2011), pp. 1–122.
- [4] J.-F. CAI, R. H. CHAN, L. SHEN, AND Z. SHEN, *Convergence analysis of tight framelet approach for missing data recovery*, Advances in Computational Mathematics, 31 (2009), pp. 87–113.
- [5] J.-F. CAI, R. H. CHAN, L. SHEN, AND Z. SHEN, *Simultaneously inpainting in image and transformed domains*, Numerische Mathematik, 112 (2009), pp. 509–533.
- [6] J.-F. CAI, H. JI, C. LIU, AND Z. SHEN, *Framelet-based blind motion deblurring from a single image*, IEEE Transactions on Image Processing, 21 (2012), pp. 562–572.
- [7] J.-F. CAI, S. OSHER, AND Z. SHEN, *Linearized bregman iterations for frame-based image deblurring*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 226–252.

- [8] J.-F. CAI, S. OSHER, AND Z. SHEN, *Split bregman methods and frame based image restoration*, Multiscale modeling & simulation, 8 (2009), pp. 337–369.
- [9] E. J. CANDÈS, J. ROMBERG, AND T. TAO, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Transactions on information theory, 52 (2006), pp. 489–509.
- [10] A. CHAMBOLLE, V. CASELLES, D. CREMERS, M. NOVAGA, AND T. POCK, *An introduction to total variation for image analysis*, Theoretical foundations and numerical methods for sparse recovery, 9 (2010), p. 227.
- [11] A. CHAMBOLLE AND T. POCK, *On the ergodic convergence rates of a first-order primal-dual algorithm*, Mathematical Programming, (2015), pp. 1–35.
- [12] T. F. CHAN AND S. ESEDOGLU, *Aspects of total variation regularized l1 function approximation*, SIAM Journal on Applied Mathematics, 65 (2005), pp. 1817–1837.
- [13] T. F. CHAN, G. H. GOLUB, AND P. MULET, *A nonlinear primal-dual method for total variation-based image restoration*, SIAM journal on scientific computing, 20 (1999), pp. 1964–1977.
- [14] F. CHEN, *Composite Minimization: Proximity Algorithms and Their Applications*, PhD thesis, Syracuse University, 2015.
- [15] F. CHEN, L. SHEN, B. W. SUTER, AND Y. XU, *Nesterov’s algorithm solving dual formulation for compressed sensing*, Journal of Computational and Applied Mathematics, 260 (2014), pp. 1–17.
- [16] F. CHEN, L. SHEN, B. W. SUTER, AND Y. XU, *Minimizing compositions of functions using proximity algorithms with application in image deblurring*, Frontiers in Applied Mathematics and Statistics, 2 (2016), p. 12.

- [17] F. CHEN, L. SHEN, Y. XU, AND X. ZENG, *The moreau envelope approach for the $l1/tv$ image denoising model*, Inverse Problems and Imaging, 8 (2014), pp. 53–77.
- [18] P. CHEN, J. HUANG, AND X. ZHANG, *A primal–dual fixed point algorithm for convex separable minimization with applications to image restoration*, Inverse Problems, 29 (2013), p. 025011.
- [19] P. CHEN, J. HUANG, AND X. ZHANG, *A primal-dual fixed point algorithm for minimization of the sum of three convex separable functions*, Fixed Point Theory and Applications, 2016 (2016), p. 54.
- [20] P. CHEN, J. HUANG, AND X. ZHANG, *A primal-dual fixed point algorithm for multi-block convex minimization*, arXiv preprint arXiv:1602.00414, (2016).
- [21] S. S. CHEN, D. L. DONOHO, AND M. A. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM review, 43 (2001), pp. 129–159.
- [22] C. CLASON, B. JIN, AND K. KUNISCH, *A duality-based splitting method for $l1-tv$ image restoration with automatic regularization parameter choice*, SIAM Journal on Scientific Computing, 32 (2010), pp. 1484–1505.
- [23] P. L. COMBETTES, *Quasi-fejérian analysis of some optimization algorithms*, Studies in Computational Mathematics, 8 (2001), pp. 115–152.
- [24] P. L. COMBETTES, *Solving monotone inclusions via compositions of nonexpansive averaged operators*, Optimization, 53 (2004), pp. 475–504.
- [25] P. L. COMBETTES AND H. H. BAUSCHKE, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, 2011.
- [26] P. L. COMBETTES AND J.-C. PESQUET, *A douglas–rachford splitting approach to nonsmooth convex variational signal recovery*, IEEE Journal of Selected Topics in Signal Processing, 1 (2007), pp. 564–574.

- [27] P. L. COMBETTES AND V. R. WAJS, *Signal recovery by proximal forward-backward splitting*, Multiscale Modeling & Simulation, 4 (2005), pp. 1168–1200.
- [28] C. CORTES AND V. VAPNIK, *Support-vector networks*, Machine learning, 20 (1995), pp. 273–297.
- [29] D. L. DONOHO, *Compressed sensing*, IEEE Transactions on information theory, 52 (2006), pp. 1289–1306.
- [30] J. DOUGLAS AND H. H. RACHFORD, *On the numerical solution of heat conduction problems in two and three space variables*, Transactions of the American mathematical Society, 82 (1956), pp. 421–439.
- [31] J. ECKSTEIN AND D. P. BERTSEKAS, *On the douglas—rachford splitting method and the proximal point algorithm for maximal monotone operators*, Mathematical Programming, 55 (1992), pp. 293–318.
- [32] E. ESSER, *Applications of lagrangian-based alternating direction methods and connections to split bregman*, CAM report, 9 (2009), p. 31.
- [33] E. ESSER, X. ZHANG, AND T. F. CHAN, *A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 1015–1046.
- [34] H. FANG, L. YAN, H. LIU, AND Y. CHANG, *Blind poissonian images deconvolution with framelet regularization*, Optics letters, 38 (2013), pp. 389–391.
- [35] D. GABAY, *Chapter ix applications of the method of multipliers to variational inequalities*, in Studies in mathematics and its applications, vol. 15, Elsevier, 1983, pp. 299–331.
- [36] D. GABAY AND B. MERCIER, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Institut de recherche d’informatique et d’automatique, 1975.

- [37] R. GLOWINSKI AND P. LE TALLEC, *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*, vol. 9, SIAM, 1989.
- [38] K. GOEBEL AND W. A. KIRK, *Topics in metric fixed point theory*, vol. 28, Cambridge University Press, 1990.
- [39] T. GOLDSTEIN AND S. OSHER, *The split bregman method for l_1 -regularized problems*, SIAM journal on imaging sciences, 2 (2009), pp. 323–343.
- [40] A. GRANAS AND J. DUGUNDJI, *Fixed point theory*, Springer Science & Business Media, 2013.
- [41] X. GUO, F. LI, AND M. K. NG, *A fast l_1 -tv algorithm for image restoration*, SIAM Journal on Scientific Computing, 31 (2009), pp. 2322–2341.
- [42] S. HARMELING, H. MICHAEL, AND B. SCHÖLKOPF, *Space-variant single-image blind deconvolution for removing camera shake*, in Advances in Neural Information Processing Systems, 2010, pp. 829–837.
- [43] B. HE AND X. YUAN, *Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective*, SIAM Journal on Imaging Sciences, 5 (2012), pp. 119–149.
- [44] M. R. HESTENES, *Multiplier and gradient methods*, Journal of optimization theory and applications, 4 (1969), pp. 303–320.
- [45] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, 1990.
- [46] J. HUANG, S. ZHANG, AND D. METAXAS, *Efficient mr image reconstruction for compressed mr imaging*, Medical Image Analysis, 15 (2011), pp. 670–679.
- [47] R.-Q. JIA AND Z. SHEN, *Multiresolution and wavelets*, Proceedings of the Edinburgh Mathematical Society, 37 (1994), pp. 271–300.

- [48] B. JOHANSSON, T. KEVICZKY, M. JOHANSSON, AND K. H. JOHANSSON, *Subgradient methods and consensus algorithms for solving convex optimization problems*, in Decision and Control, 2008. CDC 2008. 47th IEEE Conference on, IEEE, 2008, pp. 4185–4190.
- [49] A. KROL, S. LI, L. SHEN, AND Y. XU, *Preconditioned alternating projection algorithms for maximum a posteriori ect reconstruction*, Inverse problems, 28 (2012), p. 115005.
- [50] Q. LI, C. A. MICCHELLI, L. SHEN, AND Y. XU, *A proximity algorithm accelerated by gauss–seidel iterations for l1/tv denoising models*, Inverse Problems, 28 (2012), p. 095003.
- [51] Q. LI, L. SHEN, Y. XU, AND N. ZHANG, *Multi-step fixed-point proximity algorithms for solving a class of optimization problems arising from image processing*, Advances in Computational Mathematics, 41 (2015), pp. 387–422.
- [52] Q. LI, Y. XU, AND N. ZHANG, *Two-step fixed-point proximity algorithms for multi-block separable convex problems*, Journal of Scientific Computing, 70 (2017), pp. 1204–1228.
- [53] Q. LI AND N. ZHANG, *Fast proximity-gradient algorithms for structured convex optimization problems*, Applied and Computational Harmonic Analysis, 41 (2016), pp. 491–517.
- [54] Y.-R. LI, Z.-Y. CAO, AND Y.-W. WEN, *Color image restoration based on framelet regularization*, in Systems and Informatics (ICSAI), 2012 International Conference on, IEEE, 2012, pp. 1892–1895.
- [55] P.-L. LIONS AND B. MERCIER, *Splitting algorithms for the sum of two nonlinear operators*, SIAM Journal on Numerical Analysis, 16 (1979), pp. 964–979.

- [56] J. LU, L. SHEN, C. XU, AND Y. XU, *Multiplicative noise removal in imaging: An exp-model and its fixed-point proximity algorithm*, Applied and Computational Harmonic Analysis, 41 (2016), pp. 518–539.
- [57] J. LU, L. SHEN, C. XU, AND Y. XU, *Multiplicative noise removal with a sparsity-aware optimization model*, Inverse Problems & Imaging, 11 (2017), pp. 949–974.
- [58] M. LUSTIG, D. DONOHO, AND J. M. PAULY, *Sparse mri: The application of compressed sensing for rapid mr imaging*, Magnetic resonance in medicine, 58 (2007), pp. 1182–1195.
- [59] S. MA, W. YIN, Y. ZHANG, AND A. CHAKRABORTY, *An efficient algorithm for compressed mr imaging using total variation and wavelets*, in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE, 2008, pp. 1–8.
- [60] C. A. MICCHELLI, L. SHEN, AND Y. XU, *Proximity algorithms for image models: denoising*, Inverse Problems, 27 (2011), p. 045009.
- [61] C. A. MICCHELLI, L. SHEN, Y. XU, AND X. ZENG, *Proximity algorithms for the l_1/tv image denoising model*, Advances in Computational Mathematics, 38 (2013), pp. 401–426.
- [62] J.-J. MOREAU, *Proximité et dualité dans un espace hilbertien*, Bull. Soc. Math. France, 93 (1965), pp. 273–299.
- [63] J. G. NAGY AND D. P. O’LEARY, *Restoring images degraded by spatially variant blur*, SIAM Journal on Scientific Computing, 19 (1998), pp. 1063–1082.
- [64] A. NEDIC AND A. OZDAGLAR, *Distributed subgradient methods for multi-agent optimization*, IEEE Transactions on Automatic Control, 54 (2009), pp. 48–61.
- [65] M. NIKOLOVA, *A variational approach to remove outliers and impulse noise*, Journal of Mathematical Imaging and Vision, 20 (2004), pp. 99–120.

- [66] J. NOCEDAL AND S. J. WRIGHT, *Conjugate gradient methods*, Numerical optimization, (2006), pp. 101–134.
- [67] S. OSHER, M. BURGER, D. GOLDFARB, J. XU, AND W. YIN, *An iterative regularization method for total variation-based image restoration*, Multiscale Modeling & Simulation, 4 (2005), pp. 460–489.
- [68] N. PARIKH, S. P. BOYD, ET AL., *Proximal algorithms*, Foundations and Trends in optimization, 1 (2014), pp. 127–239.
- [69] T. POCK AND A. CHAMBOLLE, *Diagonal preconditioning for first order primal-dual algorithms in convex optimization*, in Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 1762–1769.
- [70] M. J. POWELL, *A method for non-linear constraints in minimization problems*, UKAEA, 1967.
- [71] R. T. ROCKAFELLAR, *The multiplier method of hestenes and powell applied to convex programming*, Journal of Optimization Theory and applications, 12 (1973), pp. 555–562.
- [72] R. T. ROCKAFELLAR, *Conjugate duality and optimization*, vol. 16, Siam, 1974.
- [73] R. T. ROCKAFELLAR, *Monotone operators and the proximal point algorithm*, SIAM journal on control and optimization, 14 (1976), pp. 877–898.
- [74] L. I. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Physica D: Nonlinear Phenomena, 60 (1992), pp. 259–268.
- [75] A. A. SAWCHUK, *Space-variant image restoration by coordinate transformations*, JOSA, 64 (1974), pp. 138–144.
- [76] J. SHI, W. YIN, S. OSHER, AND P. SAJDA, *A fast hybrid algorithm for large-scale l_1 -regularized logistic regression*, Journal of Machine Learning Research, 11 (2010), pp. 713–741.

- [77] F. SROUBEK, J. KAMENICKY, AND Y. M. LU, *Decomposition of space-variant blur in image deconvolution*, IEEE Signal Processing Letters, 23 (2016), pp. 346–350.
- [78] R. TIBSHIRANI, *Regression shrinkage and selection via the lasso*, Journal of the Royal Statistical Society. Series B (Methodological), (1996), pp. 267–288.
- [79] R. TIBSHIRANI, M. SAUNDERS, S. ROSSET, J. ZHU, AND K. KNIGHT, *Sparsity and smoothness via the fused lasso*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67 (2005), pp. 91–108.
- [80] P. TSENG, *Applications of a splitting algorithm to decomposition in convex programming and variational inequalities*, SIAM Journal on Control and Optimization, 29 (1991), pp. 119–138.
- [81] V. VAPNIK, *The nature of statistical learning theory*, Springer science & business media, 2013.
- [82] A. CHAMBOLLE AND T. POCK, *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of Mathematical Imaging and Vision, 40 (2011), pp. 120–145.
- [83] L. XIAO, S. BOYD, AND S.-J. KIM, *Distributed average consensus with least-mean-square deviation*, Journal of parallel and distributed computing, 67 (2007), pp. 33–46.
- [84] J. YANG, Y. ZHANG, AND W. YIN, *A fast alternating direction method for tvl1-l2 signal reconstruction from partial fourier data*, IEEE Journal of Selected Topics in Signal Processing, 4 (2010), pp. 288–297.
- [85] C. ZALINESCU, *Convex analysis in general vector spaces*, World scientific, 2002.
- [86] X. ZHANG, M. BURGER, X. BRESSON, AND S. OSHER, *Bregmanized nonlocal regularization for deconvolution and sparse reconstruction*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 253–276.

- [87] M. ZHU AND T. CHAN, *An efficient primal-dual hybrid gradient algorithm for total variation image restoration*, UCLA CAM Report, (2008), pp. 08–34.

Biographical Data

NAME OF AUTHOR: Xiaoxia Liu

PLACE OF BIRTH: Chaozhou, Guangdong, China

DATE OF BIRTH: September 24, 1989

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

Syracuse University, Syracuse, New York, USA

Sun Yat-sen University, Guangzhou, Guangdong, China

DEGREES AWARDED:

B.S. Information and Computational Science, Sun Yat-sen University, 2012

M.S. Mathematics, Syracuse University, 2016