A QUASI-NEWTON SUBSPACE TRUST REGION ALGORITHM FOR NONMONOTONE VARIATIONAL INEQUALITIES AND APPLICATIONS IN ADVERSARIAL LEARNING

ZICHENG QIU

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University

Department of Applied Mathematics

A quasi-Newton subspace trust region algorithm for nonmonotone variational inequalities and applications in adversarial learning

Zicheng Qiu

A thesis submitted in partial fulfilment of the requirements

for the degree of Doctor of Philosophy

December 2024

Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____(Signature)

Qiu Zicheng (Name of student)

Dedicate to my parents.

Abstract

Nonmonotone Variational Inequalities (VIs) are widely applied in the fields of data sciences and machine learning. However, the design of algorithms towards nonmonotone VIs is still a challenge. State-of-the-art algorithms can only solve nonmonotone VIs under some strong assumptions, such as pseudomonotonicity or Minty's condition. The main purpose of this thesis is to study a class of nonmonotone VIs with box constraints, which is equivalent to a system of nonsmooth equations. The thesis is primarily divided into two parts.

In the first part of the thesis, we propose a smoothing Quasi-Newton Subspace Trust Region (QNSTR) algorithm for the least squares problems defined by the smoothing approximation of nonsmooth equations. Based on the structure of the nonmonotone VI, we use an adaptive quasi-Newton formula to approximate the Hessian matrix and solve a low-dimensional strongly convex quadratic program with ellipse constraints in a subspace at each step of the QNSTR algorithm efficiently. Moreover, we study the relationship between solutions of the VI and first order stationary points of the least squares problem, and prove the global convergence of the QNSTR algorithm to a solution of the VI under some mild conditions. We also propose a strategy to update the smoothing parameter and establish its complexity.

In the second part of the thesis, we implement the QNSTR algorithm to solve a box constrained nonconvex-nonconcave minimax optimization problem with application to practical problems. Since the objective function of the optimization problem

has expectation, we apply the Sample Average Approximation (SAA) method to solve the optimization problem. We prove that any accumulation points of the global minimax point, first order stationary point, second order stationary point of the SAA problem is a global minimax point, first order stationary point, second order stationary point of the original problem respectively, as the sample size N tends to be infinity. We formulate the first order optimality condition of the box constrained SAA minimax problem as a nonmonotone VI, and apply the QNSTR to find a first order stationary point of the SAA problem via the nonmonotone VI. We also present numerical results based on the QNSTR algorithm with different subspaces for generative adversarial networks on several practical adversarial learning problems using real data on eyes. The numerical results show that the QNSTR algorithm is efficient and effective for solving large scale minimax optimization problems.

This thesis contains the research results of the following paper, which published during the period of my Ph.D. study.

- Z. Qiu, J. Jiang , X. Chen. A Quasi-Newton Subspace Trust Region Algorithm for Nonmonotone Variational Inequalities in Adversarial Learning over Box Constraints. Journal of Scientific Computing, 101(2): 45, 2024.

Acknowledgements

First and foremost, I would like to express my sincere appreciation to my supervisor, Professor Xiaojun Chen, for her patient guidance and encouraging participation in the supervision of my Ph.D. study. Her professional suggestions and insightful ideas helped me tide over the difficulties in my research. Without her generous support, I cannot finish this Ph.D. thesis successfully. Her extensive knowledge, rigorous academic attitude and enthusiasm for research have inspired me not only in my Ph.D. study, but also in my future life.

In addition, I also would like to express my gratitude to my co-supervisor, Doctor Yun Shi, who always talks to me and provides a lot of guidances for me on my research. He has provided me with inspiration and insightful discussions that have shaped my research topic.

Moveover, I would like to thank my collaborator and senior, Doctor Jie Jiang. He provides a lot of guidances and assitances on my Ph.D study period.

I am very grateful to join the research group of Professor Xiaojun Chen and my special thanks also go to my academic brothers and sisters: Professor Chao Zhang, Professor Congpei An, Professor Wei Bian, Professor Xin Liu, Professor Yanfang Zhang, Professor Hailin Sun, Professor Zaikun Zhang, Professor Xiao Wang, Dr. Yang Zhou, Dr. Lei Yang, Dr. Bo Wen, Dr. Hong Wang, Dr. Zhenhua Peng, Dr. Chao Li, Dr. Fang He, Dr. Xiaozhou Wang, Dr. Jianfeng Luo, Dr. Wei Liu, Dr. Shisen Liu, Dr. Guan Wang, Dr. Lei Wang, Dr. Xingbang Cui, Dr. You Zhao, Dr.

Yong Zhao, Dr. Lin Chen, Dr. Zhihua Zhao, Dr. Jian Guo, Dr. Fan Wu, Dr. Fei Fang, Dr. Kaixin Gao, Miss. Xin Qu, Mr. Shijie Yu, Miss. Yue Wang, Mr. Yifan He, Mr. Xiao Zha, Miss. Lingzi Jin, Miss. Yixuan Zhang, Mr. Zhouxing Luo, Mr. Wentao Ma, Miss. Yiyang Li.

Last but not least, my deepest appreciations and intense love belong to my families: my mother Qian Cheng, my father Fang Qiu and my girl friend Bei Lin for their unconditional understanding, encouragement, trust and support throughout my life. Their love and companionship give me power to keep fearless and indomitable.

Contents

Certificate of Originality **ii**

Abstract **vi**

Acknowledgements **ix**

List of Figures **xiii**

List of Tables **xv**

List of Notation **xvi**

1 Introduction **1**

1.1 Background 1

1.2 Literature review 2

1.3 Summary of contributions 6

1.4 Organization 7

2 A globally convergent Quasi-Newton Subspace Trust Region (QNSTR) algorithm for nonsmooth least squares problems **9**

2.1 Smoothing approximation 10

2.1.1 Closed form of the smoothing approximation and basis properties 10

2.2 Structure of the QNSTR algorithm 15

2.2.1 Quasi-Newton approximation of least squares problem 16

2.3 Convergence analysis 20

2.3.1 Asymptotic convergence of the QNSTR to exact solution of VIs 28

2.3.2 Complexity analysis of the QNSTR to an inexact solution of VIs 37

3 Application of the QNSTR on nonconvex-nonconcave minimax problems 45

3.1 Nonconvex-nonconcave minimax problems 45

3.2 Sample Average Approximation (SAA) and asymptotic convergence . 47

3.3 Numerical experiments 56

3.3.1 Logistic regression min-max problems 57

3.3.2 Two-layer GAN on MNIST data 61

4 Image segmentation by mix generative adversarial networks using the QNSTR 73

4.1 Image segmentation problem 73

4.2 Blood vessel segmentation of Digital Retinal Images (DRI) 76

4.2.1 Segmentation results of DRIVE data 77

4.2.2 Segmentation results of CHASE_DB1 data 78

4.3 Eye image segmentation for optic disc 79

5 Conclusions and future work 89

5.1 Conclusions 89

5.2 Future work 90

Bibliography 92

List of Figures

2.1 Smoothing approximations of $\mathrm{mid}(l, u, q(z))$ with $l = -1$, $u = 1$, $q(z) = z$ for $-2 \le z \le 2$: (a) $h(\cdot, \mu)$ in (2.1.3) with different values of μ; (b) comparison of $h(\cdot, \mu)$ in (2.1.3) and the smoothing functions $h(\cdot, \mu)$ in (2.1.1) and (2.1.2) with $\mu = 0.3$. 13

3.1 Results for the smoothing function $s(\cdot, \mu)$: (a) $s(\cdot, \mu)$ in (3.3.2) with different values of μ; (b) comparison of $s(\cdot, \mu)$ in (3.3.2) and SCAD [1], MCP [2] with $\mu = 0.2$. 58

3.2 The convergence of res_N as N grows (Left: the range of res_N with different N; Right: the boxplot of res_N with different N) 64

3.3 The convergence of res_μ as μ decreases (Left: the range of res_μ with different μ; Right: the boxplot of res_μ with different μ). 65

3.4 Smoothing approximation error of $F_{N,\mu}(z)$ to $F_N(z)$ for $N = 1000, 2000, 10000$. 66

3.5 The residual $\|F_{N,\mu}(z_k)\|$ with $V_k^{z,H}$ (left: $N_G^1 = N_D^1 = 64$; right: $N_G^1 = N_D^1 = 128$) 67

3.6 The residual $\|F_{N,\mu}(z_k)\|$ with different choice of V_k (left: $N_G^1 = N_D^1 = 64$; right: $N_G^1 = N_D^1 = 128$) 68

3.7 Comparison results between the QNSTR algorithm for V_k^z, V_k^F, V_k^g, $V_k^{z,H}$ and γ-alternate Adam (left: $N_G^1 = N_D^1 = 64$; right $N_G^1 = N_D^1 = 128$) 69

3.8 Comparison results between the QNSTR algorithm for V_k^z, V_k^F, V_k^g, $V_k^{z,H}$ and PPA (left: $N_G^1 = N_D^1 = 64$; right $N_G^1 = N_D^1 = 128$) 69

3.9 Comparison results between the QNSTR algorithm for V_k^z, V_k^F, V_k^g, $V_k^{z,H}$ and γ-alternate Adam (left: $N_G^1 = N_D^1 = 64$; right $N_G^1 = N_D^1 = 128$) 70

4.1 Results on DRIVE: Row 1. fundus image, Row 2. manual segmentation, Row 3. vessel map generated by GANs with the QNSTR algorithm, Row 4. yellow (correct); red (wrong); green (missing), Row 5. error. 81

4.2 Comparison of alternating Adam and the QNSTR algorithm on DRIVE. Columns from left to right are: 1. manual segmentation, 2. vessel map generated by GANs with alternating Adam, 3. yellow (correct); red (wrong); green (missing) of alternating Adam, 4. vessel map generated by GANs with QNSTR algorithm, 5. yellow (correct); red (wrong); green (missing) of the QNSTR algorithm. 82

4.3 Results on CHASE_DB1: Row 1. fundus image, Row 2. manual segmentation, Row 3. vessel map generated by GANs with QNSTR algorithm, Row 4. yellow (correct); red (wrong); green (missing), Row 5. error. 83

4.4 Comparison of alternating Adam and the QNSTR algorithm on CHASE_DB1. Columns from left to right are: 1. manual segmentation, 2. vessel map generated by GANs with alternating Adam, 3. yellow (correct); red (wrong); green (missing) of alternating Adam, 4. vessel map generated by GANs with QNSTR algorithm, 5. yellow (correct); red (wrong); green (missing) of the QNSTR algorithm. 84

4.5 Results on RIM-ONE: Column 1. fundus image, Column 2. manual segmentation, Column 3. optic dist segmented by GANs with QNSTR algorithm, Column 4. yellow (correct); red (wrong); green (missing), Column 5. Error. 85

4.6 Comparison of alternating Adam and QNSTR algorithm on RIM_ONE_R2. Columns from left to right are: 1. original images, 2. manual segmentation, 3. yellow (correct); red (wrong); green (missing) of QNSTR, 4. yellow (correct); red (wrong); green (missing) of alternating Adam. 86

4.7 The segmentation results along with the ground truth. blue (ground truth); red (results of alternating Adam); green (results of QNSTR algorithm). 87

List of Tables

3.1 Performance of the QNSTR algorithm and alternating Adam with $x_0 = 0.2e$, $y_0 = 0.2e$. 60

3.2 Performance of the QNSTR algorithm and alternating Adam with $x_0 = 0.8e$, $y_0 = 0.8e$. 60

3.3 Number of elements whose absolute value is less or equal to 10^{-5} in stationary point found by the QNSTR algorithm and alternating Adam for (3.3.1) with $x_0 = 0.2e$, $y_0 = 0.2e$. 61

3.4 Number of elements whose absolute value is less or equal to 10^{-5} in stationary point found by the QNSTR algorithm and alternating Adam for (3.3.1) with $x_0 = 0.8e$, $y_0 = 0.8e$. 61

3.5 Means, variances and 95% CIs of res_N with different N 64

3.6 FID scores of different algorithms with $N_D^1 = N_G^1 = 64$ 71

3.7 FID scores of different algorithms with $N_D^1 = N_G^1 = 128$ 72

4.1 Performance of the QNSTR algorithm and alternating Adam for model (4.1.2), and other methods in [3, 4, 5] for model (4.1.1) on DRIVE 78

4.2 Performance of the QNSTR algorithm and alternating Adam for model (4.1.2), and other methods in [3, 4, 5] for model (4.1.1) on CHASE_DB1. 79

List of Notation

$\mathbb{R}$	the set of real numbers
$\mathbb{R}^n$	the set of n-dimensional real vectors
$\mathbb{R}_+$	the nonnegative orthant
$\mathbb{R}_{++}$	the positive orthant
$\mathbb{R}^{m\times n}$	the set of $m \times n$ real matrices
$\mathbb{B}$	the Euclidean closed ball centred at zero with radius one
$\mathcal{B}(x, \delta)$	the Euclidean closed ball centred at x with radius δ
$\mathcal{N}_{\mathcal{Z}}(z)$	the normal cone of $\mathcal{Z}$ at z
$\mathcal{T}_{\mathcal{Z}}(z)$	the tangent cone of $\mathcal{Z}$ at z
$x \perp y$	vectors x and y are perpendicular
$x^\top$	transpose of matrix or vector x
$\|x\|$	the ℓ_2-norm of vector x
$\|A\|$	the Euclidean norm of a matrix A defined as the square root of the maximum eigenvalue of $A^\top A$, i.e., $\sqrt{\lambda_{\max}(A^\top A)}$
A^{-1}	inverse of a nonsingular matrix A
I_n	identity matrix of dimension n
e_i	the i-th column vector of identity matrix with suitable dimension

e	the all one vector
$\mathrm{d}(a, B)$	Euclidean distance function from vector a to set B
$\mathrm{d}(A, B)$	Euclidean deviation distance function from set A to set B, i.e., $\sup_{a \in A} \mathrm{d}(a, B)$
$F'(x; d)$	directional derivative of the vector-valued function F along the direction d
$\partial F(x)$	Clarke generalized Jacobian of the vector-valued function F at x
$\mathrm{VI}(\mathcal{Z}, H)$	the variational inequality problem defined by the feasible set $\mathcal{Z}$ and operator H
$\mathrm{SOL}(\mathcal{Z}, H)$	the solution set of of $\mathrm{VI}(\mathcal{Z}, H)$

Chapter 1

Introduction

The first chapter provides an introduction to the background of variational inequalities (VIs) and reviews relevant results related to current works on nonmonotone variational inequality problems.

1.1 Background

Let $H : \mathbb{R}^n \to \mathbb{R}^n$ be a continuous operator and $\mathcal{Z} \subseteq \mathbb{R}^n$ be a closed convex set. A finite-dimensional variational inequality, denoted as $\mathrm{VI}(\mathcal{Z}, H)$, is to find a z^*, such that

$$\langle H(z^*), z - z^* \rangle \geq 0, \qquad \forall z \in \mathcal{Z}. \tag{1.1.1}$$

Recall that the *tangent cone* to a convex set $\mathcal{Z} \subseteq \mathbb{R}^n$ at $\bar{z} \in \mathcal{Z}$, denoted by $\mathcal{T}_{\mathcal{Z}}(\bar{z})$, is defined as [6, Definition 6.1]

$$\mathcal{T}_{\mathcal{Z}}(\bar{z}) := \left\{ w | \ \frac{z_k - \bar{z}}{t_k} \to w \text{ as } k \to \infty \text{ for some } z_k \overset{\mathcal{Z}}{\to} \bar{z} \text{ and } t_k \downarrow 0 \text{ as } k \to \infty \right\},$$

where $z_k \overset{\mathcal{Z}}{\to} \bar{z}$ means that $\{z_k\} \subseteq \mathcal{Z}$ and $z_k \to \bar{z}$ as $k \to \infty$. The *normal cone* to $\mathcal{Z}$ at $\bar{z}$, denoted by $\mathcal{N}_{\mathcal{Z}}(\bar{z})$, is defined as

$$\mathcal{N}_{\mathcal{Z}}(\bar{z}) := \{v | \ \langle v, w \rangle \leq 0, \forall w \in \mathcal{T}_{\mathcal{Z}}(\bar{z})\} = \{v | \ \langle v, z - \bar{z} \rangle \leq 0, \forall z \in \mathcal{Z}\}.$$

(1.1.1) can be represented as

$$\langle H(z^*), v\rangle \leq 0, \quad \forall v \in \mathcal{T}_{\mathcal{Z}}(z^*), \tag{1.1.2}$$

or

$$H(z^*) \in \mathcal{N}_{\mathcal{Z}}(z^*). \tag{1.1.3}$$

VI$(\mathcal{Z}, H)$ in (1.1.1) has many important applications in many fields including physics, economics, statistics, and engineering. A wide range of constrained optimization problems and equilibrium problems, including Nash equilibrium, economic equilibrium, and traffic equilibrium, can be equivalently reformulated as variational inequalities. This highlights the versatility of VIs as a uniform framework for tackling a diverse array of important optimization and equilibrium problems.

Given the widespread applicability of the VI$(\mathcal{Z}, H)$, it is of significant interest to develop efficient algorithms to solve this problem (1.1.1).

In this thesis, we focus on box constrained variational inequalities, i.e., we assume that in (1.1.1), the feasibility set $\mathcal{Z}$ is defined as follows:

$$\mathcal{Z} := \{z \in \mathbb{R}^n | \quad l \leq z \leq u\}, \tag{1.1.4}$$

where $l \in \mathbb{R}^n$, $u \in \mathbb{R}^n$ and $l < u$. Furthermore, we assume that H is continuously differentiable and ∇H is locally Lipschitz continuous. Under the assumption for the feasible set in (1.1.4), we can guarantee that the solution set of VI$(\mathcal{Z}, H)$, denoted as SOL$(\mathcal{Z}, H)$, is nonempty. This is because H is continuous and $\mathcal{Z}$ in (1.1.4) is compact (See Corollary 2.2.5 in [7]).

1.2 Literature review

The studies of finite-dimensional VIs can be dated back to 1960s and after that, many algorithms for solving VIs have been proposed. An important type of algorithm

for solving VIs is known as projection-type methods. The earliest projection-type methods can date back to (gradient) projection method proposed by Sibony [8], proximal method proposed by Martinet [9], and extra-gradient method proposed by Korpelevich [10]. In 1995, Tseng studied the convergence of projection-type methods of VIs [11]. After that, a number of algorithms such as modified forward-backward method [13], mirror-prox method [14], dual-extrapolation method [15, 16], hybrid proximal extra-gradient method [17], OGDA [18, 19], and extra-point method [20] are proposed. The above approaches are called first-order projection methods. Except for the first-order projection methods, there are also researches for developing high-order projection methods [21, 22, 23, 26]. An important property of VIs for designing algorithms is the (strong) monotonicity, which requires the operator H to be (strongly) monotone. We say an operator H is monotone if

$$\langle H(u) - H(v), u - v\rangle \geq 0, \qquad \forall u, v \in \mathcal{Z}. \tag{1.2.1}$$

Moreover, if there exists some $\mu > 0$ such that

$$\langle H(u) - H(v), u - v\rangle \geq \mu\|u - v\|^2, \qquad \forall u, v \in \mathcal{Z}, \tag{1.2.2}$$

we say H is strongly monotone.

All the aforementioned approaches are designed to tackle monotone problems. However, monotonicity is often too idealistic to accurately model real-world problems. Currently, many practical problems, particularly in the domains of machine learning and artificial intelligence, do not satisfy the monotonicity assumption. This necessitates the development of new algorithms that can effectively handle nonmonotone VIs. Very recently, the development of algorithms for different nonmonotone VIs has garnered significant attention, driven by their applications in machine learning [27, 28, 29, 30, 31, 32]. Most existing algorithms for nonmonotone VIs rely on assumptions that relax the monotonicity requirement. We note that there are sev-

eral conditions that can be imposed on the structure of H, which can help relax the monotonicity assumption.

- Pseudo-monotonicity:

$$\langle H(z), z' - z\rangle \geq 0 \implies \langle H(z'), z' - z\rangle \geq 0, \qquad \forall z, z' \in \mathcal{Z}. \tag{1.2.3}$$

- Quasi-monotonicity:

$$\langle H(z), z' - z\rangle > 0 \implies \langle H(z'), z' - z\rangle \geq 0, \qquad \forall z, z' \in \mathcal{Z}. \tag{1.2.4}$$

- Minty's condition: there exists $z^* \in \mathrm{SOL}(\mathcal{Z}, H)$ such that

$$\langle H(z), z - z^*\rangle \geq 0, \qquad \forall z \in \mathcal{Z}. \tag{1.2.5}$$

Based on the aforementioned assumption, projection-type methods can be extended to tackle variational inequalities with pseudomonotonicity [7, 27, 33], quasi-monotonicity [31], and those that satisfy Minty's condition [28].

To the best of our knowledge, Minty's condition is the weakest condition known to guarantee the convergence of nonmonotone VIs. However, even Minty's condition is quite stringent. We provide a counterexample to demonstrate that Minty's condition does not always hold.

Example 1.1. *Consider the following VI:*

$$0 \in H(z) + \mathcal{N}_{\mathcal{Z}}(z) \tag{b1}$$

with $z \in \mathbb{R}$, $H(z) = \sin(z)$ *and* $\mathcal{Z} = [-6, 6]$. *Obviously,* $\mathrm{SOL}(\mathcal{Z}, H) = \{-6, -\pi, 0, \pi, 6\}$. *However, for any* $z^* \in \{0, \pi, 6\}$, *there exists a* $z \in [-6, -\pi)$ *such that*

$$\langle H(z), z - z^*\rangle = \sin(z)(z - z^*) < -\pi \sin(z) < 0.$$

Therefore, we know none of $z^* \in \{0, \pi, 6\}$ *satisfies Minty's condition in* (1.2.5). *Similarly, we claim that no solution exists on* $\{-6, -\pi\}$. *Thus, we can conclude that Minty's condition does not hold on* (b1).

In fact, in most real-world optimization problems, Minty's condition is difficult to verify, and the convergence properties of current algorithms for nonmonotone VIs may fail to hold. An alternative approach to solve (1.1.1) is to reformulate VI($\mathcal{Z}, H$) into equivalent forms. For example, VI($\mathcal{Z}, H$) with $\mathcal{Z}$ in (1.1.4) can be reformulated as a nonsmooth equations system

$$F(z) := z - \text{mid}(l, u, z - H(z)) = 0, \tag{1.2.6}$$

where "mid" is the middle operator in the componentwise sense, that is

$$\text{mid}(l, u, z - H(z))_i = \begin{cases} l_i, & \text{if} \quad (z - H(z))_i < l_i, \\ u_i, & \text{if} \quad (z - H(z))_i > u_i, \\ (z - H(z))_i, & \text{otherwise}, \end{cases} \quad i = 1, \cdots, n.$$

Moreover, since SOL($\mathcal{Z}, H$) is nonempty, solving the nonsmooth equations system is equivalent to solve the nonsmooth minimization problem as follows:

$$\min_z \ r(z) := \frac{1}{2}\|F(z)\|^2. \tag{1.2.7}$$

Obviously, the global optimizer z^* of r, that is $r(z^*) = 0$ is the necessary and sufficient condition of $z^* \in \text{SOL}(\mathcal{Z}, H)$.

Several algorithms have been developed to solve VI($\mathcal{Z}, H$) via the nonsmooth equation system (1.2.6) or the minimization problem (1.2.7), including quasi-Newton methods [34], semismooth-Newton methods [35], inexact-Newton methods [36], and trust region methods [37]. By incorporating strategies such as line search to select a suitable step size, the global convergence of these methods can be established. However, these approaches typically require solving a high-dimensional linear system of equations at each iteration, which can be computationally challenging for large-scale problems. This limitation motivates us to develop a new method for nonmonotone VIs, particularly tailored for large-scale problems.

In this thesis, we introduce a Quasi-Newton Subspace Trust Region (QNSTR) algorithm to solve the nonmonotone VIs via the least squares problems (1.2.7).

1.3 Summary of contributions

The contributions of this thesis are as follows:

- We propose a smoothing QNSTR algorithm for (1.1.1) via the least squares problem (1.2.7). Leveraging the structure of (1.2.7), we employ an adaptive quasi-Newton formula to approximate the Hessian matrix and solve a low-dimensional strongly convex quadratic program with elliptical constraints in a subspace at each iteration. Additionally, we explore the relationship between the solutions of the VI($\mathcal{Z}, H$) and the least squares problem (1.2.7), demonstrating the global convergence of the QNSTR algorithm to a solution of VI($\mathcal{Z}, H$). Under the assumption that the Jacobian matrix of F at all accumulation points of the sequence generated by QNSTR are nonsingular, we prove that the complexity of the QNSTR to obtain a ϵ-solution of (1.1.1) is $O(\ln(\epsilon^{-2})\epsilon^{-3})$.

- We investigate the connection between VIs and practical optimization problems, with a focus on nonconvex-nonconcave minimax optimization problems with expectation. Initially, we employ the Sample Average Approximation (SAA) method to discretize the problem, examining the relationship between the original minimax problem and its SAA. We then present numerical results obtained from the QNSTR algorithm across various subspaces, which demonstrate its efficiency and effectiveness in solving large-scale minimax optimization problems. Moreover, we implement the QNSTR algorithm and apply it to practical image segmentation problems. The results showcase the promise of our proposed algorithm for a broad range of adversarial learning problems, particularly in scenarios where data is limited.

1.4 Organization

The thesis is organized as follows.

- In Chapter 1, we introduce the background and provide a literature review for the related topic.

- In Chapter 2, we first introduce the smoothing approximation for the nonsmooth (1.2.7) and obtain a closed form of the smoothing approximation of (1.2.7). Based on the differentiable smoothing problem, the QNSTR with quasi-Newton approximation dedicated in the structure of least squares is developed. The global asymptotic convergence property is established under suitable assumptions. Moreover, we propose the update strategy of the smoothing parameter and give the complexity of the QNSTR.

- In Chapter 3, we implement the QNSTR algorithm to address a nonconvex-nonconcave minimax optimization problem. Initially, we apply the SAA method to convert the problem into a discrete summation form, to examine the relationship between the original minimax problem and its SAA representation. Furthermore, we apply the QNSTR on some minimax problems in practice to show its outperformance comparing with other algorithms.

- In Chapter 4, we apply the QNSTR algorithm on some real image segmentation problems. We use three different datasets on eye and apply the QNSTR algorithm on a mix Generative Adversarial Network (GAN), which can be represented as a nonconvex-nonconcave minimax problem. We apply F1-score, Sensitivity, Specificity and Accuracy to judge the quality of the segmentation results. According to the results, we show that the QNSTR is efficient and effective on large scale minimax optimization problems.

- In Chapter 5, we summarize the main results of the thesis and discuss some future work.

Chapter 2

A globally convergent Quasi-Newton Subspace Trust Region (QNSTR) algorithm for nonsmooth least squares problems

In this chapter, we propose the QNSTR algorithm to solve the nonlinear equations (1.2.6) via the nonlinear nonsmooth least squares problem (1.2.7), which has the global convergence property under some suitable assumptions.

Since H is locally Lipschitz continuous, we know F is locally Lipschitz continuous. According to Rademacher's theorem, F is differentiable almost everywhere. The Clarke generalized Jacobian of F at z [40] is defined as

$$\partial F(z) = \text{con}\partial_B F(z), \tag{2.0.1}$$

and

$$\partial_B F(z) = \{v|\ \nabla F(z') \to v, F \text{ is differentiable at } z',\ z' \to z\},$$

where "con" denotes the convex hull. Similarly, the Clarke generalized gradient of r in (1.2.7) is defined as:

$$\partial r(z) = \text{con}\{v|\ \nabla r(z') \to v, r \text{ is differentiable at } z',\ z' \to z\},$$

and a vector z is called a Clarke stationary point of (1.2.7) if $0 \in \partial r(z)$.

Because of the nonsmoothness of F, we first give the smoothing approximation $\tilde{F}$ of F. We will study the relationship between $\tilde{F}$ and F in the next section.

2.1 Smoothing approximation

Definition 2.1. *Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a continuous function. For a smoothing function $\tilde{F} : \mathbb{R}^n \times \mathbb{R}_{++} \to \mathbb{R}^n$, if*

1. *$\tilde{F}(\cdot, \mu)$ is continuously differentiable in $\mathbb{R}^n$ for any fixed $\mu > 0$,*

2. *for any $z \in \mathbb{R}^n$,*

$$\lim_{\bar{z} \to z, \mu \downarrow 0} \tilde{F}(\bar{z}, \mu) = F(z)$$

holds, we say $\tilde{F}(\cdot, \mu)$ is a smoothing approximation of F and μ is the smoothing parameter.

2.1.1 Closed form of the smoothing approximation and basis properties

Let $q(z) = z - H(z)$. The function F is not differentiable at z when $q(z)_i = l_i$ or $q(z)_i = u_i$ for some $1 \leq i \leq n$. To handle the nonsmoothness of $F(z)$, we adopt its smoothing approximation

$$\tilde{F}(z, \mu) := z - h(z, \mu),$$

where $h(z, \mu)$ is a smoothing approximation of the term $\mathrm{mid}(l, u, q(z))$, which is given by (see [38, 39])

$$h(z, \mu)_i := \int_{-\infty}^{\infty} (\mathrm{mid}(l_i, u_i, q(z)_i - \mu t)\, \rho(t) \mathrm{d}t$$

for $i = 1, \cdots, n$, where $\rho : \mathbb{R} \to \mathbb{R}_+$ is a density function with a bounded absolute mean, that is,

$$\int_{-\infty}^{\infty} |t|\, \rho(t) \mathrm{d}t < \infty.$$

In this case, we have, for $i = 1, \cdots, n$, that

$$\begin{aligned}\lim_{\mu\downarrow 0} \nabla_z \tilde{F}(z,\mu)_i &= \lim_{\mu\downarrow 0} \nabla_z (z_i - h(z,\mu)_i) \\ &= \lim_{\mu\downarrow 0} \left(e_i - \left(\int_{(q(z)_i - u_i)/\mu}^{(q(z)_i - l_i)/\mu} \rho(t)\mathrm{d}t \right) (e_i - \nabla H(z)_i) \right) \\ &= \begin{cases} \nabla H(z)_i, & \text{if } q(z)_i \in (l_i, u_i), \\ e_i - (\int_{-\infty}^{0} \rho(t)\mathrm{d}t)(e_i - \nabla H(z)_i), & \text{if } q(z)_i = l_i, \\ e_i - (\int_{0}^{\infty} \rho(t)\mathrm{d}t)(e_i - \nabla H(z)_i), & \text{if } q(z)_i = u_i, \\ e_i, & \text{otherwise,} \end{cases}\end{aligned}$$

where e_i is the i-th column of the $n \times n$ identity matrix.

The smoothing function $h(\cdot, \mu)$ can be expressed explicitly with some specific density functions ρ. To see this, we give the following example.

Example 2.1 ([34]). *1. If* $\rho(t) = \frac{e^{-t}}{(1+e^{-t})^2}$, *then*

$$h(z,\mu)_i = l_i + \mu \log(1 + e^{(q(z)_i - l_i)/\mu}) - \mu \log(1 + e^{(q(z)_i - u_i)/\mu}). \tag{2.1.1}$$

2. If $\rho(t) = \frac{2}{(t^2+4)^{\frac{3}{2}}}$, *then*

$$h(z,\mu)_i = \frac{1}{2}\left(\sqrt{(q(z)_i - l_i)^2 + 4\mu^2} - \sqrt{(u_i - q(z)_i)^2 + 4\mu^2} + u_i + l_i \right). \tag{2.1.2}$$

3. If $\rho(t) = \begin{cases} 1, & |t| \le 0.5, \\ 0, & \textit{otherwise}, \end{cases}$ *then for any* $0 < \mu \le \min_{1\le i\le n}(u_i - l_i)$, *we have*

$$h(z,\mu)_i = \begin{cases} -\frac{1}{2\mu}(u_i - q(z)_i + \frac{\mu}{2})^2 + u_i, & |u_i - q(z)_i| \le \frac{\mu}{2}, \\ \frac{1}{2\mu}(l_i - q(z)_i - \frac{\mu}{2})^2 + l_i, & |l_i - q(z)_i| \le \frac{\mu}{2}, \\ l_i, & q(z)_i < l_i - \frac{\mu}{2}, \\ u_i, & q(z)_i > u_i + \frac{\mu}{2}, \\ q(z)_i, & \textit{otherwise.} \end{cases} \tag{2.1.3}$$

In the remainder of the thesis, we will apply the smoothing form in (2.1.3).

We first give some properties of the smoothing function (2.1.3) for the "mid" function.

Proposition 2.1. *The function $h(\cdot,\mu)$ in (2.1.3) is continuously differentiable for any fixed $\mu \in (0,\hat{\mu})$ with $\hat{\mu} := \min_{1\le i\le n}(1, u_i - l_i)$ and satisfies the following properties:*

(i) $|h(z,\mu)_i - \mathrm{mid}(l,u,q(z))_i| \le \frac{\mu}{8}$ *for any* $z \in \mathrm{R}^n$ *and* $\mu \in (0,\hat{\mu})$;

(ii) $h(z,\mu)_i = \mathrm{mid}(l,u,q(z))_i$ *if* $|q(z)_i - l_i| > \frac{\mu}{2}$ *and* $|q(z)_i - u_i| > \frac{\mu}{2}$;

(iii) $\|\nabla_z h(z,\mu)_i - \nabla_z h(z',\mu)_i\| \le \frac{\|\nabla q(z)_i\|}{\mu}\|q(z)_i - q(z')_i\| + \|\nabla q(z)_i - \nabla q(z')_i\|$ *for any* $z, z' \in \mathbb{R}^n$.

Proof. By the definition of h in (2.1.3), we obtain

$$|h(z,\mu)_i - \mathrm{mid}(l,u,q(z))_i| = \begin{cases} \frac{1}{2\mu}(u_i - q(z)_i + \frac{\mu}{2})^2, & \text{if } u_i < q(z)_i < u_i + \frac{\mu}{2}, \\ \frac{1}{2\mu}(u_i - q(z)_i - \frac{\mu}{2})^2, & \text{if } u_i - \frac{\mu}{2} \le q(z)_i \le u_i, \\ \frac{1}{2\mu}(l_i - q(z)_i + \frac{\mu}{2})^2, & \text{if } l_i < q(z)_i < l_i + \frac{\mu}{2}, \\ \frac{1}{2\mu}(l_i - q(z)_i - \frac{\mu}{2})^2, & \text{if } l_i - \frac{\mu}{2} < q(z)_i < l_i, \\ 0, & \text{otherwise.} \end{cases} \tag{2.1.4}$$

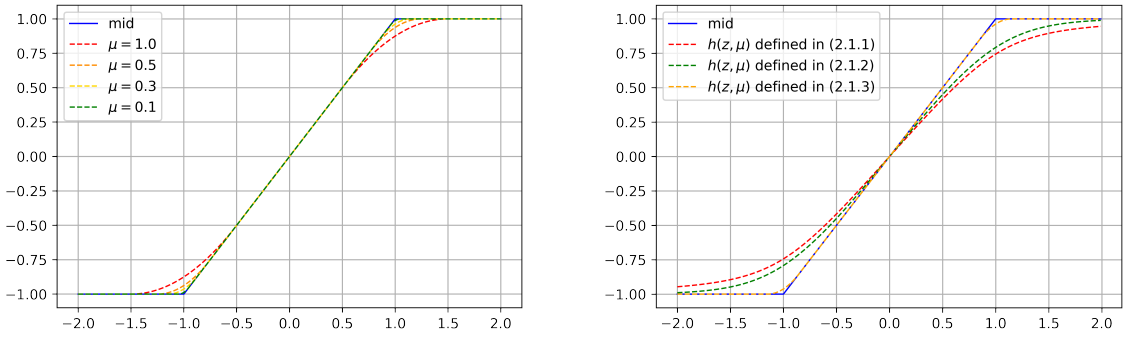
This implies (i) and (ii). By straightforward calculation, we can verify that $h(z,\mu)$ is continuously differentiable with respect to z, and the closed form of $\nabla h(z,\mu)_i$ can be represented as

$$\nabla h(z,\mu)_i = g(q(z),\mu)_i e_i \nabla q(z)_i, \tag{2.1.5}$$

with

$$g(t,\mu)_i = \begin{cases} -\frac{1}{\mu}(t_i - u_i - \frac{\mu}{2}), & \text{if } |u_i - t_i| \le \frac{\mu}{2}, \\ \frac{1}{\mu}(t_i - l_i + \frac{\mu}{2}), & \text{if } |l_i - t_i| \le \frac{\mu}{2}, \\ 1, & \text{if } l_i + \frac{\mu}{2} \le t_i \le u_i - \frac{\mu}{2}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.1.6}$$

Obviously, we know $\|g(t,\mu)_i e_i\| \le 1$ for any $t \in \mathbb{R}^n$. According to Proposition 4.2.2 (c) in [7], we know $g(\cdot,\mu)$ is globally Lipschitz continuous with a Lipschitz constant



(a) (b)

Figure 2.1: Smoothing approximations of $\text{mid}(l, u, q(z))$ with $l = -1$, $u = 1$, $q(z) = z$ for $-2 \le z \le 2$: (a) $h(\cdot, \mu)$ in (2.1.3) with different values of μ; (b) comparison of $h(\cdot, \mu)$ in (2.1.3) and the smoothing functions $h(\cdot, \mu)$ in (2.1.1) and (2.1.2) with $\mu = 0.3$.

$\frac{1}{\mu}$. Then

$$\begin{aligned}\|\nabla_z h(z,\mu)_i - \nabla_z h(z',\mu)_i\| &\le \|g(q(z),\mu)_i e_i \nabla q(z)_i - g(q(z'),\mu)_i e_i \nabla q(z')_i\| \\ &\le \|(g(q(z),\mu)_i - g(q(z'),\mu)_i) e_i \nabla q(z)_i\| + \|g(q(z'),\mu)_i e_i (\nabla q(z)_i - \nabla q(z')_i)\| \\ &\le \frac{1}{\mu}\|\nabla q(z)_i\|\|q(z)_i - q(z')_i\| + \|\nabla q(z)_i - \nabla q(z')_i\|\end{aligned} \tag{2.1.7}$$

for any fixed $\mu \in (0, \hat{\mu})$ and the estimation in (iii) holds. □

By Proposition 2.1 (ii), it holds that for any fixed $z \in \mathbb{R}^n$, if $q(z)_i \ne l_i$ and $q(z)_i \ne u_i$ for $i = 1, \cdots, n$, there exists $\bar{\mu}$, such that $h(z, \mu) = \text{mid}(l, u, q(z))$. This is the main advantage of $h(\cdot, \mu)$ in (2.1.3) compared with the other approximation forms such as (2.1.1) or (2.1.2). Figure 2.1 (a) shows that the smoothing function $h(z, \mu)$ for $\text{mid}(l, u, q(z))$ with different values of μ when $l = -1$, $u = 1$ and $q(z) = z$ for $z \in [-2, 2]$, while Figure 2.1 (b) shows the relationship of "mid" and its smoothing approximation forms defined in (2.1.1), (2.1.2) and (2.1.3).

Comparing with other smoothing approximation forms such as (2.1.1) and (2.1.2), (2.1.3) has better approximation quality towards the nonsmooth "mid" function for

given smoothing parameter μ. Moreover, (2.1.3) can be formulated as a piecewise quadratic form of q, which is easy to calculate in practical computation. If $h(\cdot,\mu)$ is defined in (2.1.3), the closed form of $\tilde{F}(\cdot,\mu)$ can be formulated as

$$\tilde{F}(z,\mu)_i = \begin{cases} \frac{1}{2}(H(z)_i + z_i) + \frac{1}{2\mu}(u_i - q(z)_i)^2 + \frac{\mu}{8} - \frac{u_i}{2}, & \text{if } |u_i - q(z)_i| \le \frac{\mu}{2}, \\ \frac{1}{2}(H(z)_i + z_i) - \frac{1}{2\mu}(l_i - q(z)_i)^2 - \frac{\mu}{8} - \frac{l_i}{2}, & \text{if } |l_i - q(z)_i| \le \frac{\mu}{2}, \\ F(z)_i, & \text{otherwise.} \end{cases} \tag{2.1.8}$$

In what follows, we summarize some useful properties of the smoothing function $\tilde{F}(\cdot,\mu)$.

Lemma 2.1. *Let $\tilde{F}(\cdot,\mu)$ be a smoothing approximation of $F(\cdot)$ defined in (2.1.8). Then for any $\mu \in (0,\hat{\mu})$, $\tilde{F}(\cdot,\mu)$ is continuously differentiable and has the following properties.*

(i) There is a $\kappa = \frac{\sqrt{n}}{8}$ such that for any $z \in \mathbb{R}^n$ and $\mu > 0$,

$$\|\tilde{F}(z,\mu) - F(z)\| \le \kappa\mu. \tag{2.1.9}$$

(ii) For any $z \in \mathbb{R}^n$, we have

$$\lim_{\mu\downarrow 0} \mathrm{d}(\nabla_z \tilde{F}(z,\mu), \partial_C F(z)) = 0,$$

where $\partial_C F(z) = \partial F(z)_1 \times \partial F_N(z)_2 \times \cdots \times \partial F(z)_n$, and $\partial(F(z))_i$ is the Clarke generalized gradient of $F(\cdot)_i$ at z for $i = 1,\ldots,n$. Moreover, there exists a $\bar{\mu} > 0$ such that for any $\mu \in (0,\bar{\mu})$, we have $\nabla\tilde{F}(z,\mu) \in \partial_C F(z)$.

Proof. Since (ii) has been proved in [35], we only need to prove the conclusion in (i). According to (i) in Proposition 2.1, we know

$$\begin{aligned} \|\tilde{F}(z,\mu) - F(z)\|^2 &= \|h(z,\mu) - \mathrm{mid}(l,u,q(z))\|^2 \\ &\le n\|h(z,\mu) - \mathrm{mid}(l,u,q(z))\|_\infty^2 \\ &\le \frac{n}{64}\mu^2, \end{aligned}$$

which implies (i). □

By introducing the smoothing $\tilde{F}(\cdot, \mu)$, we can formulate a smoothing least squares problem as follows

$$\min_z \tilde{r}(z, \mu) := \frac{1}{2}\|\tilde{F}(z, \mu)\|^2. \tag{2.1.10}$$

And the gap between the minimizers of (1.2.7) and (2.1.10) can be controlled by the smoothing parameter μ. Next, we will introduce the algorithm to solve (2.1.10). Before we give the algorithm, we first give two definitions as follows, which are relevant to the results of the proposed algorithm.

Definition 2.2. *[40] A function $r : \mathbb{R}^n \to \mathbb{R}$ is said to be regular at $z \in \mathbb{R}^n$ if for all $v \in \mathbb{R}^n$, its directional derivative exists and*

$$r(z; v) = \lim_{t \downarrow 0} \frac{r(z + tv) - r(z)}{t} = \lim \sup_{\bar{z} \to z, t \downarrow 0} \frac{r(\bar{z} + tv) - r(\bar{z})}{t}. \tag{2.1.11}$$

If r is regular at all $z \in \mathbb{R}^n$, r is said to be regular.

Definition 2.3. *[39] A smoothing function $\tilde{r}$ of $r : \mathbb{R}^n \to \mathbb{R}$ is said to satisfy the gradient consistency at $z \in \mathbb{R}^n$ if*

$$\text{con}\{v |\ \nabla_z \tilde{r}(\bar{z}, \mu) \to v \text{ for } \bar{z} \to z, \mu \downarrow 0\} = \partial r(z). \tag{2.1.12}$$

If $\tilde{r}$ satisfies gradient consistency at all $z \in \mathbb{R}^n$, we say $\tilde{r}$ satisfies gradient consistency.

2.2 Structure of the QNSTR algorithm

Problem (2.1.10) has been extensively studied for several decades, with various methods such as Newton's method, inexact Newton's method, Levenberg-Marquardt method, quasi-Newton's method and trust region method are purposed to solve

(2.1.10) (See [12]). However, in each step of these methods, they need to solve a linear system in full scale. This means that the iteration subproblem will be difficult to solve when the dimension n is sufficiently large. Therefore, we propose the QN-STR for (2.1.10), which can be regarded as the combination of subspace trust region method and quasi-Newton method. The global convergence to its Clarke stationary point is established under suitable assumptions.

2.2.1 Quasi-Newton approximation of least squares problem

In this subsection, we study the structure of the least squares problem. For simplicity, we use $\nabla\tilde{r}(z,\mu)$ and $\nabla\tilde{F}(z,\mu)$ to represent $\nabla_z\tilde{r}(z,\mu)$ and $\nabla_z\tilde{F}(z,\mu)$ respectively, and use F_k, g_k, J_k to denote $\tilde{F}(x_k,\mu_k)$, $\nabla_z\tilde{r}(z_k,\mu_k)$ and $\nabla_z\tilde{F}(z_k,\mu_k)$ respectively. If $\tilde{F}(\cdot,\mu)$ is twice continuously differentiable, the gradient and Hessian of $\tilde{r}(\cdot,\mu)$ can be expressed as:

$$\nabla\tilde{r}(z,\mu) = \nabla\tilde{F}(z,\mu)^\top\tilde{F}(z,\mu), \tag{2.2.1}$$

$$\nabla^2\tilde{r}(z,\mu) = \nabla\tilde{F}(z,\mu)^\top\nabla\tilde{F}(z,\mu) + \sum_{i=1}^{n}\tilde{F}(z,\mu)_i\nabla^2\tilde{F}(z,\mu)_i. \tag{2.2.2}$$

Since $\tilde{F}(\cdot,\mu)$ defined in (2.1.3) is not twice continuously differentiable, the second part in (2.2.2) is unfeasible. In order to give a globally convergent algorithm for problem (1.2.7) without using the second order derivatives, we keep the term $\nabla\tilde{F}(z,\mu)^\top\nabla\tilde{F}(z,\mu)$ in (2.2.2), and approximate $\sum_{i=1}^{n}\tilde{F}(z,\mu)_i\nabla^2\tilde{F}(z,\mu)_i$ by quasi-Newton form. Thus, the Hessian approximation at the k-th iteraion point z_k is given by

$$H_k = J_k^\top J_k + A_k, \tag{2.2.3}$$

where

$$A_{k+1} = \begin{cases} B_{k+1}, & \text{if } \|B_{k+1}\| \le \gamma \text{ and } (v_k^\top s_k)/(s_k^\top s_k) \ge \bar{\epsilon}, \\ \|F_{k+1}\|I_n, & \text{otherwise,} \end{cases} \tag{2.2.4}$$

and

$$B_{k+1} = \begin{cases} B_k - \frac{B_k s_k s_k^\top B_k^\top}{s_k^\top B_k s_k} + \frac{v_k v_k^\top}{s_k^\top v_k}, & \text{if } \|B_{k+1}\| \leq \gamma \text{ and } (v_k^\top s_k)/(s_k^\top s_k) \geq \bar{\epsilon}, \\ B_k, & \text{otherwise} \end{cases} \tag{2.2.5}$$

with $s_k := z_{k+1} - z_k$ and $v_k := (J_{k+1} - J_k)^\top F_{k+1}\|F_{k+1}\|/\|F_k\|$.

Moreover, we choose $\{d_k^1, \cdots, d_k^{L-1}\}$ $(L \geq 2)$ to let $-g_k, d_k^1, \cdots d_k^{L-1}$ be L linearly independent vectors. In each iteration of the QNSTR, we update z_{k+1} as

$$\begin{aligned} z_{k+1} =& z_k + p_k, \\ p_k =& \alpha_k^1(-g_k) + \sum_{i=2}^{L} \alpha_k^i d_k^{i-1}. \end{aligned}$$

This is equivalent to adding a constraint of $p \in \mathcal{L}_k$ on the trust region subproblem such that

$$\begin{aligned} p_k = \arg\min_{p \in \mathcal{L}_k} \ & \tilde{r}(z_k) + g_k^\top p + \frac{1}{2} p^\top H_k p, \\ \text{s.t. } & \|p\| \leq \Delta_k \end{aligned} \tag{2.2.6}$$

with $\mathcal{L}_k = \text{span}\{-g_k, d_k^1, \cdots, d_k^{L-1}\}$. Let $V_k = [-g_k, d_k^1, \cdots, d_k^{L-1}] \in \mathbb{R}^{n \times L}$ and denote

$$G_k := V_k^\top V_k, \qquad c_k := V_k^\top g_k, \qquad Q_k := V_k^\top H_k V_k,$$

we can reformulate (2.2.6) as

$$\begin{aligned} \alpha_k = \arg\min_{\alpha \in \mathbb{R}^L} \quad & m_k(\alpha) := \tilde{r}(z_k) + c_k^\top \alpha + \tfrac{1}{2} \alpha^\top Q_k \alpha, \\ \text{s.t.} \quad & \|V_k \alpha\| \leq \Delta_k. \end{aligned} \tag{2.2.7}$$

In each iteration, QNSTR algorithm only solves a low dimensional strongly convex quadratic subproblem (2.2.7) in L dimensions. Here, the strong convexity can be guaranteed for initializing H_0 as positive definite because in [41], they proved H_{k+1} would be always positive definite if H_k is positive definite.

Focusing on the difficulty of storing and calculating the approximate Hessian matrix H_k for large-scale problems in practice, we announce the tricks as follows to avoid the whole Hessian storage or computation. First, we denote

$$\mathcal{K}_1 := \{k|\ \|B_{k+1}\| \leq \gamma \text{ and } (v_k^\top s_k)/(s_k^\top s_k) \geq \bar{\epsilon}\} \quad \text{and} \quad \mathcal{K}_2 = \mathbb{N} \setminus \mathcal{K}_1,$$

and Q_k can be represented as

$$Q_k = \begin{cases} (J_k V_k)^\top J_k V_k + V_k^\top B_k V_k, & k \in \mathcal{K}_1, \\ (J_k V_k)^\top J_k V_k + \|F_{k+1}\| V_k^\top V_k, & k \in \mathcal{K}_2. \end{cases} \tag{2.2.8}$$

$\|F_{k+1}\| V_k^\top V_k$ can be computed directly. For the term $(J_k V_k)^\top J_k V_k$ in (2.2.8), we compute $J_k V_k$ in a component-wise way

$$J_k g_k \approx \frac{\tilde{F}(z_k + \epsilon g_k) - \tilde{F}(z_k)}{\epsilon}, \quad J_k d_k^i \approx \frac{\tilde{F}(z_k + \epsilon d_k^i) - \tilde{F}(z_k)}{\epsilon}, i = 1, \cdots, L-1.$$

Since the above calculations are independent, they can be computed in parallel efficiently. For the computation of $V_k^\top B_k V_k$, $k \in \mathcal{K}_1$, we know

$$B_k V_k = B_0 V_k + \sum_{k' \in \mathcal{K}_1} \frac{B_{k'} s_{k'} s_{k'}^\top B_{k'}^\top}{s_{k'}^\top B_{k'} s_{k'}} V_k + \sum_{k' \in \mathcal{K}_1} \frac{v_{k'} v_{k'}^\top}{v_{k'}^\top s_{k'}}. \tag{2.2.9}$$

Suppose k is the $(J+1)$-th element in $\mathcal{K}_1$ and we use $k_1 < k_2 < \cdots < k_J$ to represent the J elements in $\mathcal{K}_1$ before k. According to the update rules of A_k in (2.2.4) and (2.2.5), then

$$\begin{aligned} B_k V_k &= B_{k_J} V_k - \frac{B_{k_J} s_{k_J} s_{k_J}^\top B_{k_J}^\top}{s_{k_J}^\top B_{k_J} s_{k_J}} V_k + \frac{v_{k_J} v_{k_J}^\top}{v_{k_J}^\top s_{k_J}} V_k \\ &= B_{k_1} V_k - \sum_{j=1}^{J} \frac{B_{k_j} s_{k_j} s_{k_j}^\top B_{k_j}}{s_{k_j}^\top B_{k_j} s_{k_j}} V_k + \sum_{j=1}^{J} \frac{v_{k_j} v_{k_j}^\top}{v_{k_j}^\top s_{k_j}} V_k \\ &= B_{k_1} V_k - \sum_{j=1}^{J} (a_j^k)^\top \otimes (B_{k_j} s_{k_j}) + \sum_{j=1}^{J} (b_j^k)^\top \otimes v_{k_j}, \end{aligned} \tag{2.2.10}$$

where $(a_j^k)^\top = \frac{(A_{k_j}s_{k_j})^\top V_k}{s_{k_j}^\top A_{k_j}s_{k_j}}$, $(b_j^k)^\top = \frac{v_{k_j}^\top V_k}{v_{k_j}^\top s_{k_j}}$ and $\otimes$ represents the Kronecker product. Moreover, $B_{k_j}s_{k_j}$ in (2.2.10) can be computed by

$$\begin{aligned} B_{k_j}s_{k_j} &= B_{k_{j-1}}s_{k_j} - \frac{B_{k_{j-1}}s_{k_{j-1}}s_{k_{j-1}}^\top B_{k_{j-1}}^\top}{s_{k_{j-1}}^\top B_{k_{j-1}}s_{k_{j-1}}}s_{k_j} + \frac{v_{k_{j-1}}v_{k_{j-1}}^\top}{v_{k_{j-1}}^\top s_{k_{j-1}}}s_{k_j} \\ &= B_{k_1}s_{k_j} - \sum_{i=0}^{j-1}(\frac{s_{k_i}^\top B_{k_i}^\top s_{k_j}}{s_{k_i}^\top B_{k_i}s_{k_i}})B_{k_i}s_{k_i} + \sum_{i=0}^{j-1}(\frac{v_{k_i}^\top s_{k_j}}{v_{k_i}^\top s_{k_i}})v_{k_i}. \end{aligned} \tag{2.2.11}$$

If we initialize $B_0 = I_n$, from update rule in (2.2.5), we know $B_0 = \cdots = B_{k_1} = I_n$. Thus, the computation of quasi-Newton matrix B_kV_k can be replaced by a sequence of vector-vector products involving $\{s_k\}_{k\in\mathcal{K}_1}$, $\{v_k\}_{k\in\mathcal{K}_1}$ and the column vectors of V_k. Consequently, Q_k can be calculated efficiently without computing and storing the full information of H_k. To compute v_k, we can leverage automatic differentiation techniques to compute

$$J_{k+1}^\top F_{k+1} = \nabla_z(\tilde{F}(z,\mu_{k+1})^\top F_{k+1})|_{z=z_{k+1}} \quad \text{and} \quad J_k^\top F_{k+1} = \nabla_z(\tilde{F}(z,\mu_k)^\top F_{k+1})|_{z=z_k}$$

respectively. Moreover, we can make a truncation for the computation and only use the information of last L_1 iterations to update H_k, this is called limit-memory strategy [12]. It is significantly efficient to solve large-scale least squares optimization problems with real data when the dimension n is much larger than L_1. Moreover, if we apply the limit-memory strategy, then obviously we know $\|B_{k+1}\|$ is bounded. Therefore, we can drop the judgment of $\|B_{k+1}\| \le \gamma$ in (2.2.4) and (2.2.5) if we apply limit-memory strategy.

The structure of the QNSTR is given in Algorithm 1.

Algorithm 1 QNSTR Algorithm

Given: $0 < \beta_1 < 1 < \beta_2$, $0 < \eta \leq \zeta_1 < \zeta_2 < 1$, $0 \leq \overline{\Delta} < +\infty$, $0 < \nu \leq 1$, $\tau > 0$, $K_{\max} < +\infty$; and initial value $0 < \mu_0 < \min_{1\leq i\leq n}(1, u_i - l_i)$, $\Delta_0 \in (0, \overline{\Delta})$, $z_0 \in \mathcal{Z}$. Set $k = 0$.

while $k < K_{\max}$ **do**

If $\|g_k\| = 0$, set $z_{k+1} = z_k$ and go to Step 5.

Else go to Step 1.

Step 1. Solve (2.2.7) for α_k.

Step 2. Compute the reduction ratio at iterate k:

$$\rho_k = \frac{\tilde{r}(z_k, \mu_k) - \tilde{r}(z_k + V_k\alpha_k, \mu_k)}{m_k(0) - m_k(\alpha_k)}. \tag{2.2.12}$$

Step 3. Update

$$\Delta_{k+1} = \begin{cases} \min\{\beta_2\Delta_k, \overline{\Delta}\}, & \text{if } \rho_k \geq \zeta_2 \text{ and } \|V_k\alpha_k\| = \Delta_k, \\ \beta_1\Delta_k, & \text{if } \rho_k < \zeta_1, \\ \Delta_k, & \text{otherwise.} \end{cases} \tag{2.2.13}$$

Step 4. Update

$$z_{k+1} = \begin{cases} z_k + V_k\alpha_k, & \text{if } \rho_k > \eta, \\ z_k, & \text{otherwise.} \end{cases} \tag{2.2.14}$$

Step 5. Update

$$\mu_{k+1} = \begin{cases} \nu\mu_k, & \text{if } \|\nabla\tilde{r}(z_{k+1}, \mu_k)\| \leq \tau\mu_k, \\ \mu_k, & \text{otherwise,} \end{cases} \tag{2.2.15}$$

and let $k = k + 1$.

end while

We next show the convergence of the QNSTR in Algorithm 1.

2.3 Convergence analysis

In this section, we are concerned with the convergence and stability of the QNSTR. Next, we show the QNSTR has globally convergent property of nonlinear nonsmooth least squares problems under some mild assumptions. For converience, we let $R = \max_{1\leq i\leq n}(u_i - l_i)$ and denote three sets as follows:

$$\bar{\mathcal{S}} := \left\{ z \in \mathbb{R}^n |\ \mathrm{d}(z, \mathcal{Z}) \leq \sqrt{\frac{n}{2}}(1 + \frac{\sqrt{n}}{4})R \right\}, \tag{2.3.1}$$

$$\mathcal{S}_\mu := \left\{ z \in \mathbb{R}^n |\ \tilde{r}(z, \mu) \leq \frac{n}{2}(R + \frac{\sqrt{n}}{8}\mu)^2 \right\}, \tag{2.3.2}$$

$$\mathcal{S} := \left\{ z \in \mathbb{R}^n |\ r(z) \leq \frac{n}{2}(1 + \frac{\sqrt{n}}{4})^2 R^2 \right\}. \tag{2.3.3}$$

Then for arbitrary $\mu \in (0, \hat{\mu})$ with $\hat{\mu} := \min_{1 \leq i \leq n} (1, u_i - l_i)$, we have the following relationship between the sets.

Lemma 2.2. *Let $\bar{\mathcal{S}}$, $\mathcal{S}_\mu$ and $\mathcal{S}$ be the sets defined in* (2.3.1), (2.3.2) *and* (2.3.3) *respectively, then we have*

$$\mathcal{Z} \subseteq \mathcal{S}_\mu \subseteq \mathcal{S} \subseteq \bar{\mathcal{S}}.$$

Proof. We first show that $\mathcal{Z} \subseteq \mathcal{S}_\mu$. By definition, for any $z \in \mathcal{Z}$, we have

$$\begin{aligned} r(z) &= \frac{1}{2}\|z - \mathrm{mid}(l, u, q(z))\|^2 \\ &\leq \frac{n}{2}\|z - \mathrm{mid}(l, u, q(z))\|_\infty^2 \\ &\leq \frac{n}{2}\max\left(\|z - l\|_\infty^2, \|z - u\|_\infty^2\right) \\ &\leq \frac{n}{2}R^2. \end{aligned} \tag{2.3.4}$$

Moreover, from (i) in Lemma 2.1, we know

$$\begin{aligned}
|\tilde{r}(z,\mu) - r(z)| &= \frac{1}{2}\left|\|\tilde{F}(z,\mu)\|^2 - \|F(z)\|^2\right| \\
&\leq \frac{1}{2}\sum_{i=1}^{n}\left|\tilde{F}(z,\mu)_i + F(z)_i\right| \cdot \left|\tilde{F}(z,\mu)_i - F(z)_i\right| \\
&\leq \frac{1}{2}\sum_{i=1}^{n}\|\tilde{F}(z,\mu) - F(z)\|_\infty \cdot \left|\tilde{F}(z,\mu)_i + F(z)_i\right| \\
&\leq \frac{1}{2}\sum_{i=1}^{n}\kappa\mu\left|\tilde{F}(z,\mu)_i + F(z)_i\right| \\
&= \frac{1}{2}\sum_{i=1}^{n}\kappa\mu\left|\tilde{F}(z,\mu)_i - F(z)_i + 2F(z)_i\right| \\
&\leq \frac{1}{2}\kappa\mu\|\tilde{F}(z,\mu) - F(z)\|_1 + \kappa\mu\|F(z)\|_1 \\
&\leq \frac{n}{2}\kappa\mu\|\tilde{F}(z,\mu) - F(z)\|_\infty + \sqrt{n}\kappa\mu\|F(z)\| \\
&\leq \frac{n}{2}(\kappa\mu)^2 + \sqrt{2n}\kappa\mu\sqrt{r(z)}.
\end{aligned} \tag{2.3.5}$$

This implies that

$$\begin{aligned}
\tilde{r}(z,\mu) &\leq r(z) + \sqrt{2n}\kappa\mu\sqrt{r(z)} + \frac{n}{2}(\kappa\mu)^2 \\
&\leq \frac{n}{2}R^2 + n\kappa\mu R + \frac{n}{2}(\kappa\mu)^2 \\
&= \frac{n}{2}(R + \kappa\mu)^2 \\
&= \frac{n}{2}(R + \frac{\sqrt{n}}{8}\mu)^2
\end{aligned} \tag{2.3.6}$$

for any $z \in \mathcal{Z}$, which means $\mathcal{Z} \subseteq \mathcal{S}_\mu$. Next, we show that $\mathcal{S}_\mu \subseteq \mathcal{S}$. By Lemma

2.1 (i), we know

$$
\begin{aligned}
|\tilde{r}(z,\mu) - r(z)| &\leq \frac{1}{2}\sum_{i=1}^{n} \kappa\mu \left|\tilde{F}(z,\mu)_i + F(z)_i\right| \\
&= \frac{1}{2}\sum_{i=1}^{n} \kappa\mu \left|F(z)_i - \tilde{F}(z,\mu)_i + 2\tilde{F}(z,\mu)_i\right| \\
&\leq \frac{1}{2}\kappa\mu\|\tilde{F}(z,\mu) - F(z)\|_1 + \kappa\mu\|\tilde{F}(z,\mu)\|_1 \\
&\leq \frac{n}{2}\kappa\mu\|\tilde{F}(z,\mu) - F(z)\|_\infty + \sqrt{n}\kappa\mu\|\tilde{F}(z,\mu)\| \\
&\leq \frac{n}{2}(\kappa\mu)^2 + \sqrt{2n}\kappa\mu\sqrt{\tilde{r}(z,\mu)}.
\end{aligned}
\tag{2.3.7}
$$

This implies for any $z \in \mathcal{S}_\mu$,

$$
\begin{aligned}
r(z) &\leq \tilde{r}(z,\mu) + \sqrt{2n}\kappa\mu\sqrt{\tilde{r}(z,\mu)} + \frac{n}{2}(\kappa\mu)^2 \\
&\leq \frac{n}{2}(R + \kappa\mu)^2 + n\kappa\mu(R + \kappa\mu) + \frac{n}{2}(\kappa\mu)^2 \\
&= \frac{n}{2}(R + 2\kappa\mu)^2.
\end{aligned}
\tag{2.3.8}
$$

Plugging the equality that $\kappa = \frac{\sqrt{n}}{8}$ and the inequality

$$0 < \mu \leq \min_i (u_i - l_i) \leq \max_i (u_i - l_i) = R$$

into (2.3.8), then we get the relationship that $\mathcal{S}_\mu \subseteq \mathcal{S}$ immediately. Finally, we show $\mathcal{S} \subseteq \bar{\mathcal{S}}$ by proving its contrapositive. For all $z \notin \bar{\mathcal{S}}$, we have $z \notin \mathcal{S}$ due to

$$
\begin{aligned}
r(z) &= \|z - \mathrm{mid}(l, u, q(z))\|^2 \\
&\geq \left(\mathrm{d}(z, \mathcal{Z})\right)^2 \\
&> \frac{n}{2}(1 + \frac{\sqrt{n}}{4})^2 R^2.
\end{aligned}
\tag{2.3.9}
$$

Then we complete the proof. □

The relationship in Lemma 2.2 brings some useful pieces of information:

(1) For an arbitrary initial point $z_0 \in \mathcal{Z}$, we know $z_0 \in \mathcal{S}$.

(2) Suppose $\{z_k\}_{k=1}^{\infty}$, $\{\mu_k\}_{k=1}^{\infty}$ are the sequences generated by Algorithm 1 with given $z_0 \in \mathcal{Z}$ and $\mu_0 \in (0, \hat{\mu})$, we know $z_k \in \mathcal{S}$, $\forall k \in \mathbb{N}$.

(3) $\mathcal{S}$ is bounded and $\|z - z'\| \leq \sqrt{\frac{n}{2}}(1 + \frac{\sqrt{n}}{4})R$, $\forall z, z' \in \mathcal{S}$.

Denote $\mathcal{S}(R_0) := \{z \in \mathbb{R}^n | \ \mathrm{d}(z, \mathcal{S}) \leq R_0\}$ for some $R_0 > 0$ and obviously $\mathcal{S}(R_0)$ is compact. Then we know $\nabla_z \tilde{r}(\cdot, \mu)$ is Lipschitz continuous on $\mathcal{S}(R_0)$.

Lemma 2.3. *Let $q(z) = z - H(z)$ and $\tilde{F}(\cdot, \mu)$ be the smoothing approximation of $F(\cdot)$ in (2.1.8). There exists a constant $\hat{M} > 0$, such that*

$$\|q(z) - q(z')\| \leq \hat{M}\|z - z'\|, \tag{2.3.10}$$

$$\|\nabla q(z) - \nabla q(z')\| \leq \hat{M}\|z - z'\| \tag{2.3.11}$$

for all $z, z' \in \mathcal{S}(R_0)$. With the same constant $\hat{M}$, we have

$$\|\nabla q(z)\| \leq \hat{M}, \tag{2.3.12}$$

$$\|\tilde{F}(z, \mu)\| \leq \hat{M}, \tag{2.3.13}$$

$$\|\nabla_z \tilde{F}(z, \mu)\| \leq \hat{M} \tag{2.3.14}$$

for all $z \in \mathcal{S}(R_0)$, and all $\mu \in (0, \hat{\mu})$. Moreover, $\nabla_z \tilde{r}(\cdot, \mu)$ is Lipschitz continuous in $\mathcal{S}(R_0)$, that is

$$\|\nabla_z \tilde{r}(z, \mu) - \nabla_z \tilde{r}(z', \mu)\| \leq \frac{C}{\mu}\|z - z'\|, \qquad \forall z, z' \in \mathcal{S}(R_0),$$

where $C = \sqrt{n}\hat{M}^3 + (\sqrt{n} + 1)\hat{M}^2$.

Proof. First of all, because of the continuity of ∇q, and the boundedness of $\mathcal{S}(R_0)$, we know there exists $M_1 > 0$, such that

$$\|\nabla q(z)\| \leq M_1 \tag{2.3.15}$$

for all $z \in \mathcal{S}(R_0)$. (2.3.15) also implies (2.3.10). Moreover, by the local Lipschitz continuity of ∇H, we know ∇q is locally Lipschitz continuous. Then we know there exists $M_2 > 0$ such that

$$\|\nabla q(z) - \nabla q(z')\| \leq M_2 \|z - z'\| \tag{2.3.16}$$

for all $z, z' \in \mathcal{S}(R_0)$. For any $z \in \mathcal{S}(R_0)$, we know

$$\begin{aligned} \|\tilde{F}(z,\mu)\| &\leq \|F(z)\| + \|F(z) - \tilde{F}(z,\mu)\| \\ &\leq \|F(z)\| + \frac{\sqrt{n}}{8}\hat{\mu}. \end{aligned} \tag{2.3.17}$$

Since F is continuous and $\mathcal{S}(R_0)$ is compact, we know $\|F(z)\|$ is bounded by M_3 for some $M_3 > 0$ in $\mathcal{S}(R_0)$. Thus, we know

$$\|\tilde{F}(z,\mu)\| \leq M_3 + \frac{\sqrt{n}}{8}\hat{\mu}. \tag{2.3.18}$$

For $\|\nabla \tilde{F}(z,\mu)\|$, we know

$$\begin{aligned} \|\nabla \tilde{F}(z,\mu)\| &\leq \|\nabla \tilde{F}(z,\mu)\|_F \\ &\leq \sum_{i=1}^{n} \|\nabla \tilde{F}(z,\mu)_i\| \\ &\leq n\|e_i - \nabla_z h(z,\mu)_i\| \\ &\leq n\|e_i\| + n\|\nabla q(z)_i\| \\ &\leq n(1 + \|\nabla q(z)\|) \\ &\leq n(1 + M_1). \end{aligned} \tag{2.3.19}$$

The third to last inequality holds because of the fact that $\|h(z,\mu)_i\| \leq \|\nabla q(z)_i\|$ in (2.1.5). Let $\hat{M} := \max\left(n(1+M_1), M_2, M_3 + \frac{\sqrt{n}}{8}\hat{\mu}\right)$, we proved (2.3.10), (2.3.11), (2.3.12), (2.3.13), (2.3.14) already. Noting that (2.3.14) also implies

$$\|\tilde{F}(z,\mu) - \tilde{F}(z',\mu)\| \leq \hat{M}\|z - z'\|, \tag{2.3.20}$$

and

$$\begin{aligned}
\|\nabla\tilde{F}(z,\mu) - \nabla\tilde{F}(z',\mu)\|^2 \leq & \|\nabla\tilde{F}(z,\mu) - \nabla\tilde{F}(z',\mu)\|_F^2 \\
= & \sum_{i=1}^{n} \|\nabla_z h(z,\mu)_i - \nabla_z h(z',\mu)_i\|^2 \\
\leq & \sum_{i=1}^{n} \left(\frac{\|\nabla q(z)_i\|}{\mu}\|q(z)_i - q(z')_i\| + \|\nabla q(z)_i - \nabla q(z')_i\|\right)^2 \\
\leq & \sum_{i=1}^{n} \frac{\|\nabla q(z)_i\|^2}{\mu^2}\|q(z)_i - q(z')_i\|^2 + \|\nabla q(z)_i - \nabla q(z')_i\|^2 \\
& + 2\frac{\|\nabla q(z)_i\|}{\mu}\|q(z)_i - q(z')_i\|\|\nabla q(z)_i - \nabla q(z')_i\| \\
\leq & \frac{n\hat{M}^4}{\mu^2}\|z - z'\|^2 + n\hat{M}^2\|z - z'\|^2 + 2n\frac{\hat{M}^3}{\mu}\|z - z'\|^2 \\
= & n\hat{M}^2(\frac{\hat{M}}{\mu} + 1)^2\|z - z'\|^2,
\end{aligned} \tag{2.3.21}$$

where the second last inequality holds because of Proposition 2.1 (iii). Then we have

$$\|\nabla\tilde{F}(z,\mu) - \nabla\tilde{F}(z',\mu)\| \leq \sqrt{n}\hat{M}(\frac{\hat{M}}{\mu} + 1)\|z - z'\|. \tag{2.3.22}$$

Now, we can prove the Lipschitz continuity of $\nabla\tilde{r}(\cdot,\mu)$.

$$\begin{aligned}
\|\nabla\tilde{r}(z,\mu)-\nabla\tilde{r}(z',\mu)\| &= \|\nabla\tilde{F}(z,\mu)^\top\tilde{F}(z,\mu)-\nabla\tilde{F}(z',\mu)^\top\tilde{F}(z',\mu)\| \\
&\le \|\nabla\tilde{F}(z,\mu)-\nabla\tilde{F}(z',\mu)\|\|\tilde{F}(z,\mu)\|+\|\nabla\tilde{F}(z',\mu)\|\|\tilde{F}(z,\mu)-\tilde{F}(z',\mu)\| \\
&\le \hat{M}\|\nabla\tilde{F}(z,\mu)-\nabla\tilde{F}(z',\mu)\|+\hat{M}\|\tilde{F}(z,\mu)-\tilde{F}(z',\mu)\| \\
&\le \left(\sqrt{n}\frac{\hat{M}^3}{\mu}+(\sqrt{n}+1)\hat{M}^2\right)\|z-z'\|.
\end{aligned} \tag{2.3.23}$$

When $\mu<1$, we immediately have

$$\|\nabla_z\tilde{r}(z,\mu)-\nabla_z\tilde{r}(z',\mu)\| \le \frac{C}{\mu}\|z-z'\|$$

with $C := \sqrt{n}\hat{M}^3+(\sqrt{n}+1)\hat{M}^2$. □

Lemma 2.4. *Let $\{z_k\}_{k=1}^\infty$ be the sequence generate by Algorithm 1 and $\{H_k\}_{k=1}^\infty$ be the sequence generated by the formulas* (2.2.3)*,* (2.2.4) *and* (2.2.5)*, then we know there exists a constant $M_1>0$, and $\|H_k\|\le M_1, \forall k\in\mathbb{N}$.*

Proof. First of all, by the conclusion in Lemma 2.3, we know $\{z_k\}_{k=1}^\infty\in\mathcal{S}$ is bounded. According to (2.3.19), we know there exists a constant $\bar{M}>0$, such that $\|J_k^\top J_k\|\le\bar{M}$. Let

$$\mathcal{K}_1 := \{k\in\mathbb{N}|\ \|B_{k+1}\|\le\gamma \text{ and } (v_k^\top s_k)/(s_k^\top s_k)\ge\bar{\epsilon}\},\quad \text{and}\quad \mathcal{K}_2 := \mathbb{N}\setminus\mathcal{K}_1.$$

Obviously, by the update rules in (2.2.4) and (2.2.5), we know $\|A_k\|\le\gamma$ for $k\in\mathcal{K}_1$. For $k\in\mathcal{K}_2$, we know

$$\|A_k\| = \|\tilde{F}(z_k,\mu_k)\| \le \|F(z_k)\|+\frac{\sqrt{n}}{8}R \le n(1+\frac{\sqrt{n}}{4})R+\frac{\sqrt{n}}{8}R. \tag{2.3.24}$$

The first inequality holds because of the fact that $\mu\le\max_{1\le i\le n}(u_i-l_i)=R$ and Lemma 2.1 (i). Let $M_1 := \bar{M}+\max\left(\left(n(1+\frac{\sqrt{n}}{4})+\frac{\sqrt{n}}{8}\right)R,\gamma\right)$, we obtained the lemma. □

2.3.1 Asymptotic convergence of the QNSTR to exact solution of VIs

We first consider the convergence of the QNSTR with the assumption that $\|\nabla\tilde{r}(z_{k+1},\mu_k)\| > \tau\mu_k$ for all $k \in \mathbb{N}$. In this case, $\mu_0 = \mu_1 = \cdots = \mu$ reduce to a single fixed constant and $\tilde{F}(z,\mu)$, $\tilde{r}(z,\mu)$ reduce to a function only correlated with z. For simplicity, we denote $\tilde{F}(z,\mu)$, $\tilde{r}(z,\mu)$ as $\tilde{F}(z)$, $\tilde{r}(z)$ respectively. Let $M := \max\left(\frac{C}{\mu}, M_1\right)$, we now consider the following one-dimensional problem:

$$\min_{\tau} m_k(\tau\alpha_k^s) \qquad \text{s.t.} \quad \|\tau V_k\alpha_k^s\| \le \Delta_k, \quad \tau > 0, \tag{2.3.25}$$

where α_k^s is an optimal solution of

$$\min_{\alpha} \ c_k^\top\alpha \qquad \text{s.t. } \|V_k\alpha\| \le \Delta_k. \tag{2.3.26}$$

Let τ_k denote an arbitrary optimal solution of problem (2.3.25). Then $\alpha_k^C := \tau_k\alpha_k^s$ is a feasible solution of problem (2.2.7).

In what follows, we give the closed form of α_k^C step by step. For this purpose, we consider the Karush-Kuhn-Tucker (KKT) condition of problem (2.3.26) as follows:

$$\lambda G_k\alpha + c_k = 0, \qquad 0 \le \lambda \perp \Delta_k^2 - \alpha^\top G_k\alpha \ge 0,$$

where λ is a multiplier. Since $g_k \neq 0$ and V_k is of full column rank, we have $c_k \neq 0$ and G_k is invertible. Thus, we know $\lambda > 0$, and the above KKT system gives

$$\alpha = -\frac{1}{\lambda}G_k^{-1}c_k, \qquad \Delta_k^2 = \alpha^\top G_k\alpha.$$

Then we obtain $\frac{1}{\lambda} = \sqrt{\frac{\Delta_k^2}{(G_k^{-1}c_k)^\top G_k(G_k^{-1}c_k)}}$, and the solution of (2.3.26) can be written

as

$$\begin{aligned}\alpha_k^s &= -\frac{\Delta_k}{\sqrt{(G_k^{-1}c_k)^\top G_k (G_k^{-1}c_k)}} G_k^{-1}c_k \\ &= -\frac{\Delta_k}{\sqrt{c_k^\top G_k^{-1}c_k}} G_k^{-1}c_k \\ &= -\frac{\Delta_k}{\sqrt{g_k^\top V_k (V_k^\top V_k)^{-1} V_k^\top g_k}} G_k^{-1}c_k \\ &= -\frac{\Delta_k}{\|g_k\|} G_k^{-1}c_k.\end{aligned}$$

The above last equality holds because $V_k(V_k^\top V_k)^{-1}V_k^\top d = d$ if d is the column vector of V_k. Hence, the objective function of (2.3.25) has the following form

$$\begin{aligned}m_k(\tau\alpha_k^s) &= \tilde{r}(z_k) + \tau c_k^\top \alpha_k^s + \frac{\tau^2}{2}(\alpha_k^s)^\top Q_k \alpha_k^s \\ &= \tilde{r}(z_k) - \tau\frac{\Delta_k}{\|g_k\|} c_k^\top G_k^{-1}c_k + \frac{\tau^2}{2}\left(\frac{\Delta_k}{\|g_k\|}\right)^2 (G_k^{-1}c_k)^\top Q_k G_k^{-1}c_k \\ &= \tilde{r}(z_k) - \Delta_k\|g_k\|\tau + \frac{\tau^2}{2}\left(\frac{\Delta_k}{\|g_k\|}\right)^2 g_k^\top H_k g_k\end{aligned}$$

and the constraint of (2.3.25) satisfies

$$\|\tau\alpha_k^s\|_{G_k} := \sqrt{\tau^2(\alpha_k^s)^\top G_k \alpha_k^s} = \tau\frac{\Delta_k}{\|g_k\|}\sqrt{c_k^\top G_k^{-1} G_k G_k^{-1} c_k} = \tau\Delta_k \le \Delta_k,$$

which is equivalent to $0 < \tau \le 1$.

Therefore, problem (2.3.25) can be equivalently rewritten as

$$\min_\tau \quad -\Delta_k\|g_k\|\tau + \tfrac{\tau^2}{2}\left(\tfrac{\Delta_k}{\|g_k\|}\right)^2 g_k^\top H_k g_k, \quad \text{s.t.} \quad 0 < \tau \le 1. \tag{2.3.27}$$

Since H_k is positive definite (see (2.2.4), (2.2.5)), problem (2.3.27) has the unique solution and the closed form of its solution can be written as

$$\tau_k = \min\left(\|g_k\|^3/(\Delta_k g_k^\top H_k g_k), 1\right).$$

Finally, we obtain

$$\alpha_k^C = -\min\left(\|g_k\|^3/(\Delta_k g_k^\top H_k g_k), 1\right)\frac{\Delta_k}{\|g_k\|}G_k^{-1}c_k. \tag{2.3.28}$$

Lemma 2.5. *Let α_k be the unique optimal solution of subproblem (2.2.7) at the k-th step. Then*

$$m_k(0) - m_k(\alpha_k) \geq \frac{1}{2}\|g_k\|\min\left(\Delta_k, \frac{\|g_k\|}{\|H_k\|}\right).$$

Proof. Since α_k^C is a feasible solution of problem (2.2.7), we have

$$m_k(0) - m_k(\alpha_k) \geq m_k(0) - m_k(\alpha_k^C).$$

In what follows, we verify

$$m_k(0) - m_k(\alpha_k^C) \geq \frac{1}{2}\|g_k\|\min\left(\Delta_k, \frac{\|g_k\|}{\|H_k\|}\right).$$

If $\|g_k\|^3/(\Delta_k g_k^\top H_k g_k) < 1$, substituting α_k^C (see (2.3.28)) into (2.2.7), we have

$$\begin{aligned} m_k(0) - m_k(\alpha_k^C) &= -c_k^\top \alpha_k^C - \frac{1}{2}(\alpha_k^C)^\top Q_k \alpha_k^C \\ &= \frac{\|g_k\|^2}{g_k^\top H_k g_k}c_k^\top G_k^{-1}c_k - \frac{1}{2}\frac{\|g_k\|^4}{(g_k^\top H_k g_k)^2}c_k^\top G_k^{-1}Q_k G_k^{-1}c_k \\ &= \frac{\|g_k\|^4}{g_k^\top H_k g_k} - \frac{1}{2}\frac{\|g_k\|^4}{(g_k^\top H_k g_k)^2}g_k^\top H_k g_k \\ &= \frac{1}{2}\frac{\|g_k\|^4}{g_k^\top H_k g_k} \geq \frac{1}{2}\frac{\|g_k\|^4}{\|H_k\|\|g_k\|^2} = \frac{1}{2}\frac{\|g_k\|^2}{\|H_k\|}. \end{aligned} \tag{2.3.29}$$

If $\|g_k\|^3/(\Delta_k g_k^\top H_k g_k) \geq 1$ (i.e., $g_k^\top H_k g_k \leq \frac{\|g_k\|^3}{\Delta_k}$), we have

$$\begin{aligned} m_k(0) - m_k(\alpha_k^C) &= -c_k^\top \alpha_k^C - \frac{1}{2}(\alpha_k^C)^\top Q_k \alpha_k^C \\ &= \frac{\Delta_k}{\|g_k\|} c_k^\top G_k^{-1} c_k - \frac{1}{2}\frac{\Delta_k^2}{\|g_k\|^2} c_k^\top G_k^{-1} Q_k G_k^{-1} c_k \\ &= \Delta_k \|g_k\| - \frac{1}{2}\frac{\Delta_k^2}{\|g_k\|^2} g_k^\top H_k g_k \\ &\geq \Delta_k \|g_k\| - \frac{1}{2}\frac{\Delta_k^2}{\|g_k\|^2}\frac{\|g_k\|^3}{\Delta_k} \\ &= \frac{1}{2}\|g_k\|\Delta_k. \end{aligned} \tag{2.3.30}$$

Combining (2.3.29) and (2.3.30), we complete the proof. □

In the remainder of Section 2.3.1, we only consider the case that $g_k \neq 0$ because the conclusion holds obviously if $g_k = 0$ for some $k > 0$.

Lemma 2.6. *Let $\{z_k\}_{k=0}^\infty$ be a sequence generated by Algorithm 1. Then for any index k, there exists a $\bar{k} > k$ such that $\|g_{\bar{k}}\| < \|g_k\|/2$.*

Proof. We give the proof by contradiction. Suppose that there exists a $\hat{k}$ with $\|g_{\hat{k}}\| = 2\epsilon$ for some $\epsilon > 0$, and $\|g_k\| \geq \epsilon$, $\forall k \geq \hat{k}$. Then we know from Lemma 2.5 that

$$m_k(0) - m_k(\alpha_k) \geq \frac{1}{2}\|g_k\| \min\left(\Delta_k, \frac{\|g_k\|}{\|H_k\|}\right) \geq \frac{1}{2}\epsilon \min\left(\Delta_k, \frac{\epsilon}{M}\right). \tag{2.3.31}$$

According to the definition of ρ_k in (2.2.12), we have

$$\begin{aligned} |\rho_k - 1| &= \left|\frac{\tilde{r}(z_k) - \tilde{r}(z_k + V_k\alpha_k) - (m_k(0) - m_k(\alpha_k))}{m_k(0) - m_k(\alpha_k)}\right| \\ &= \left|\frac{m_k(\alpha_k) - \tilde{r}(z_k + V_k\alpha_k)}{m_k(0) - m_k(\alpha_k)}\right|. \end{aligned} \tag{2.3.32}$$

By Taylor expansion, we have

$$\tilde{r}(z_k + V_k\alpha_k) = \tilde{r}(z_k) + g_k^\top V_k \alpha_k + \int_0^1 (\nabla \tilde{r}(z_k + tV_k\alpha_k) - \nabla \tilde{r}(z_k))^\top V_k \alpha_k \mathrm{d}t.$$

Then

$$|m_k(\alpha_k) - \tilde{r}(z_k + V_k\alpha_k)| = \left|\frac{1}{2}\alpha_k^\top Q_k\alpha_k - \int_0^1 (\nabla\tilde{r}(z_k + tV_k\alpha_k) - \nabla\tilde{r}(z_k))^\top V_k\alpha_k \mathrm{d}t\right|$$
$$\leq (M/2)\|V_k\alpha_k\|^2 + M\|V_k\alpha_k\|^2 \leq 3\Delta_k^2 M/2, \tag{2.3.33}$$

where the first inequality follows from $Q_k = V_k^\top H_k V_k$ and the mean-value theorem, and the second inequality follows from $\|V_k\alpha_k\| \leq \Delta_k$ due to the constraint in problem (2.2.7).

Then, by (2.3.31), (2.3.32) and (2.3.33), we get

$$|\rho_k - 1| \leq \frac{3\Delta_k^2 M/2}{(\epsilon/2)\min(\Delta_k, \epsilon/M)}.$$

Denote

$$\tilde{\Delta} := \min\left(\frac{(1-\zeta_1)\epsilon}{3M}, R_0\right).$$

For any $\Delta_k \leq \tilde{\Delta}$, we have

$$|\rho_k - 1| \leq \frac{3\Delta_k^2 M/2}{(\epsilon/2)\min(\Delta_k, \epsilon/M)} = \frac{3M\Delta_k^2}{\epsilon\Delta_k} = \frac{3M\Delta_k}{\epsilon} \leq \frac{3M\tilde{\Delta}}{\epsilon} \leq 1 - \zeta_1,$$

which implies $\rho_k \geq \zeta_1$, where the first equality follows from the fact that

$$\Delta_k \leq \tilde{\Delta} = \min\left(\frac{(1-\zeta_1)\epsilon}{3M}, R_0\right) \leq \frac{\epsilon}{3M} < \frac{\epsilon}{M}.$$

The above observation together with update rules in (2.2.13) indicates that $\Delta_{k+1} \geq \Delta_k$ when $\Delta_k \leq \tilde{\Delta}$ (and thus $\rho_k \geq \zeta_1$). In other words, if $\Delta_k > \tilde{\Delta}$, $\rho_k < \zeta_1$ holds. In this case,

$$\Delta_{k+1} = \beta_1\Delta_k > \beta_1\tilde{\Delta}$$

when $\Delta_k > \tilde{\Delta}$. To summarize the two cases, we then have

$$\Delta_k \geq \min(\Delta_{k-1}, \beta_1\tilde{\Delta}) \geq \cdots \geq \min(\Delta_{\hat{k}}, \beta_1\tilde{\Delta}), \quad \forall k \geq \hat{k}. \tag{2.3.34}$$

Then we prove the sequence $\{\Delta_k\}_{k\geq\hat{k}}^{\infty}$ is bounded from below. Denote a subsequence of $\{\hat{k}, \hat{k}+1, \cdots\}$ as

$$\mathcal{K} := \{k \in \mathbb{N}_+ |\ \rho_k \geq \zeta_1, k \geq \hat{k}\}. \tag{2.3.35}$$

Then one of the following two cases holds.

Case 1: $\mathcal{K}$ is a finite set. Then, there exists a $K > \hat{k}$, such that for all $k > K$, $\Delta_{k+1} = \beta_1 \Delta_k$. It is easy to obtain $\Delta_k \to 0$ as $k \to \infty$, since $\beta_1 < 1$. However, this is contradicted by the fact that Δ_k is bounded from below (see (2.3.34)).

Case 2: $\mathcal{K}$ is a infinite set. In this case, we have from the definition of ρ_k (see (2.2.12)) and $\rho_k \geq \zeta_1$ that

$$\tilde{r}(z_k) - \tilde{r}(z_{k+1}) \geq \zeta_1(m_k(0) - m_k(\alpha_k)) \geq \zeta_1 \frac{1}{2}\epsilon \min(\Delta_k, \epsilon/M) > 0,$$

where the second inequality follows from Lemma 2.5 and $\|g_k\| \geq \epsilon$ for all $k \in \mathcal{K}$.

Therefore, $\{\tilde{r}(z_k)\}_{k\in\mathcal{K}}$ is strictly decreasing. Since $\{\tilde{r}(z_k)\}_{k\in\mathcal{K}}$ is bounded from below (note that $\tilde{r}(z) \geq 0$ for any z), we know that the sequence $\{\tilde{r}(z_k)\}_{k\in\mathcal{K}}$ is convergent and $\tilde{r}(z_k) - \tilde{r}(z_{k+1}) \downarrow 0$ as $k \overset{\mathcal{K}}{\to} \infty$. Thus, $\Delta_k \to 0$ as $k \overset{\mathcal{K}}{\to} \infty$, which is also contradicted by (2.3.34). □

Now we are ready to give the proof of Theorem 2.1.

Theorem 2.1. *Let $\{z_k\}_{k=0}^{\infty}$ be an infinite sequence generated by Algorithm 1. We have* $\lim_{k\to\infty} \|g_k\| = 0$.

Proof. Let

$$\epsilon := \frac{1}{2}\|g_k\| \quad \text{and} \quad \underline{R} := \min\left(\frac{\epsilon}{M}, R_0\right). \tag{2.3.36}$$

Note that $\mathcal{B}(z_k, \underline{R}) = \{z : \|z - z_k\| \leq \underline{R}\} \subseteq \mathcal{S}(R_0)$, and thus $\nabla\tilde{r}(\cdot)$ is Lipschitz continuous on $\mathcal{B}(z_k, \underline{R})$ with Lipschitz modulus M. Thus, for $\forall z \in \mathcal{B}(z_k, \underline{R})$, we have

$$\|\nabla\tilde{r}(z) - \nabla\tilde{r}(z_k)\| \leq M\|z - z_k\| \leq M\underline{R} = M\min\left(\frac{\epsilon}{M}, R_0\right) \leq \epsilon.$$

For $\forall z \in \mathcal{B}(z_k, \underline{R})$, we have by the triangle inequality that

$$\|\nabla\tilde{r}(z)\| \geq \|g_k\| - \|\nabla\tilde{r}(z) - \nabla\tilde{r}(z_k)\| = 2\epsilon - \|\nabla\tilde{r}(z) - \nabla\tilde{r}(z_k)\| \geq 2\epsilon - \epsilon = \epsilon.$$

According to Lemma 2.6, we know that there exists an index $l \geq k$ satisfying $\|g_{l+1}\| < \epsilon$. Furthermore, we assume that z_{l+1} is the first point that iterates out of the ball $\mathcal{B}(z_k, \underline{R})$ after z_k as well as satisfying $\|g_{l+1}\| < \epsilon$. Consequently, $\|g_i\| \geq \epsilon$ for $i = k, k+1, \cdots, l$. Then we have

$$\begin{aligned}\tilde{r}(z_k) - \tilde{r}(z_{l+1}) = \sum_{i=k}^{l} \tilde{r}(z_i) - \tilde{r}(z_{i+1}) = \sum_{\substack{i=k,\\ z_i \neq z_{i+1}}}^{l} \rho_i(m_i(0) - m_i(\alpha_i)) \\ \geq \sum_{\substack{i=k,\\ z_i \neq z_{i+1}}}^{l} \eta(m_i(0) - m_i(\alpha_i)) \geq \frac{\eta}{2}\epsilon \sum_{\substack{i=k,\\ z_i \neq z_{i+1}}}^{l} \min\left(\Delta_i, \frac{\epsilon}{M}\right),\end{aligned} \tag{2.3.37}$$

where the second equality follows from (2.2.12), the first inequality follows from $\rho_i \geq \eta$ when $z_i \neq z_{i+1}$. Since $\|g_k\| = 2\epsilon$ and $\|g_{l+1}\| < \epsilon$, we have $z_{l+1} \neq z_k$, which implies that $\{k, \cdots, l\} \cap \{j : z_j \neq z_{j+1}\} \neq \emptyset$.

If $\Delta_i \leq \epsilon/M$ for all $i \in \{k, \cdots, l\} \cap \{j : z_j \neq z_{j+1}\}$, we continue (2.3.37) as follows:

$$\begin{aligned}\tilde{r}(z_k) - \tilde{r}(z_{l+1}) &\geq \frac{\eta}{2}\epsilon \sum_{\substack{i=k,\\ z_i \neq z_{i+1}}}^{l} \Delta_i \geq \frac{\eta}{2}\epsilon \sum_{i=k}^{l} \|z_{i+1} - z_i\| \\ &\geq \frac{\eta}{2}\epsilon \|z_k - z_{l+1}\| \geq \frac{\eta}{2}\epsilon\underline{R} = \frac{\eta}{2}\epsilon \min\left(\frac{\epsilon}{M}, R_0\right),\end{aligned}$$

where the second inequality follows from $\|z_{i+1} - z_i\| \leq \Delta_i$, the third inequality follows from the triangle inequality, the last inequality follows from the fact that z_{l+1} is the first point that iterates out of the ball $\mathcal{B}(z_k, \underline{R})$ after z_k.

If $\Delta_i > \epsilon/M$ for some $i \in \{k, \cdots, l\} \cap \{j : z_j \neq z_{j+1}\}$, we continue (2.3.37) as follows:

$$\tilde{r}(z_k) - \tilde{r}(z_{l+1}) \geq \frac{\eta}{2}\epsilon \sum_{\substack{i=k,\\ z_i \neq z_{i+1}}}^{l} \frac{\epsilon}{M} \geq \frac{\eta}{2}\epsilon\frac{\epsilon}{M},$$

where the last inequality follows from $\{k, \cdots, l\} \cap \{j : z_j \neq z_{j+1}\} \neq \emptyset$. To summarize, we obtain

$$\tilde{r}(z_k) - \tilde{r}(z_{l+1}) \geq \frac{\eta}{2}\epsilon \min\left(\frac{\epsilon}{M}, R_0\right). \tag{2.3.38}$$

Since the sequence $\{\tilde{r}(z_i)\}_{i=0}^{\infty}$ is a decreasing and bounded sequence from below, there exists $r^* \geq 0$ such that $\lim_{i\to\infty} \tilde{r}(z_i) = r^*$. Hence

$$\tilde{r}(z_k) - r^* \geq \tilde{r}(z_k) - \tilde{r}(z_{l+1}) \geq \frac{\eta}{2}\epsilon \min\left(\frac{\epsilon}{M}, R_0\right) = \frac{\eta}{4}\|g_k\| \min\left(\frac{\|g_k\|}{2M}, R_0\right),$$

where the second inequality follows from (2.3.38), the last equality follows from (2.3.36).

Due to the arbitrariness of k, by letting $k \to \infty$, we know

$$\frac{\eta}{4}\|g_k\| \min\left(\frac{\|g_k\|}{2M}, R_0\right) \to 0,$$

which implies $\lim_{k\to\infty} \|g_k\| = 0$. □

Next, we give the convergence of the QNSTR in Algorithm 1. Since $\underline{\Delta} > 0$ and $0 < \nu < 1$, $\{\mu_k\}_{k=0}^{\infty}$ will be a nonincreasing sequence of value and we have the following global convergence theorem for Algorithm 1.

Lemma 2.7. *Let $\{z_k\}_{k=0}^{\infty}$ and $\{\mu_k\}_{k=0}^{\infty}$ be the sequences generated by Algorithm 1. Define the index set*

$$\mathcal{K}_\mu := \{k|\ \|\nabla_z \tilde{r}(z_k, \mu_k)\| \leq \tau\mu_k\}. \tag{2.3.39}$$

Then $\mathcal{K}_\mu$ defined in (2.3.39) *is an infinite set.*

Proof. Suppose $\mathcal{K}_\mu$ is finite. Then based on the definition of $\mathcal{K}_\mu$ and the design of Algorithm 1, there exists a nonnegative integer $\hat{K}$, such that for all nonnegative integers j, the smoothing parameter reduce to a constant, $\mu_{\hat{K}+j} = \mu_{\hat{K}}$ and

$$\|\nabla_z \tilde{r}(z_{\hat{K}+j}, \mu_{\hat{K}+j})\| > \tau\mu_{\hat{K}}. \tag{2.3.40}$$

Noting that $\tilde{r}(z_{\hat{K}+j}, \mu_{\hat{K}}) \leq \frac{n}{2}(1+\frac{\sqrt{n}}{8})^2 R^2$ for all $j > 0$. According to Theorem 2.1, we know

$$\lim_{j\to\infty} \nabla_z \tilde{r}(z_{\hat{K}+j}, \mu_{\hat{K}+j}) = 0.$$

Since $\mu_{\hat{K}} > 0$, there exists a positive integer $\bar{J}$, such that

$$\|\nabla_z \tilde{r}(z_{\hat{K}+j}, \mu_{\hat{K}})\| < \tau\mu_{\hat{K}}, \qquad \forall j \geq \bar{J}. \tag{2.3.41}$$

This contradicts to the conclusion in (2.3.40), which is derived under the assumption that $\mathcal{K}_\mu$ is finite. Thus, $\mathcal{K}_\mu$ is an infinite set. □

Theorem 2.2. *Let $\{z_k\}_{k=0}^{\infty}$ and $\{\mu_k\}_{k=0}^{\infty}$ be the sequences generated by Algorithm 1. Then*

$$\liminf_{k\to\infty} \|\nabla \tilde{r}(z_k, \mu_k)\| = 0. \tag{2.3.42}$$

Proof. Let $\mathcal{K}_\mu = \{k_j |\ j = 1, 2, \cdots\}$ with $k_1 < k_2 < k_3 < \cdots$. From $0 < \nu < 1$, we know that

$$0 < \mu_{k_j} < \nu\mu_{k_j-1} = \nu\mu_{k_j-2} = \cdots = \nu\mu_{k_{j-1}+1} < \nu^2\mu_{k_{j-1}} = \nu^2\mu_{k_{j-1}-1} = \cdots \nu^{j-1}\mu_{k_1}.$$

Since $\mathcal{K}_\mu$ is an infinite set, we get $\lim_{j\to\infty} \mu_{k_j} = 0$. This implies that $\lim_{k\to\infty} \mu_k = 0$ because our algorithm generates a monotonically decreasing sequence $\{\mu_k\}_{k=1}^{\infty}$. Therefore, the following inequality

$$\|\nabla_z \tilde{r}(z_{k_j}, \mu_{k_j})\| \leq \tau\mu_{k_j} \leq \tau\nu^{j-1}\mu_{k_0} \tag{2.3.43}$$

gives that $\lim_{j\to\infty} \|\nabla_z \tilde{r}(z_{k_j}, \mu_{k_j})\| = 0$. Consequently, $\liminf_{k\to\infty} \|\nabla_z \tilde{r}(z_k, \mu_k)\| = 0$. □

In the above analysis, we have shown that $\liminf_{k\to\infty} \nabla \tilde{r}(z_k, \mu_k) = 0$. Then we can conclude that there exists an accumulation point of $\{z_k\}_{k=0}^{\infty}$ generated by the QNSTR (in Algorithm 1) is a Clarke stationary point of (1.2.7). Moreover,

if all elements in the Clarke generalized Jacobian of F at the accumulation points are nonsingular, the accumulation point is a solution of VI($\mathcal{Z}, H$). To ensure that any accumulation point of $\{z_k\}_{k=0}^{\infty}$ generated by the QNSTR algorithm is a Clarke stationary point of (1.2.7), we need functions $F(\cdot)_i$, $i = 1, \cdots, n$ to be regular and their smoothing functions $\tilde{F}_i$ to satisfy the gradient consistency. However, "mid" function does not satisfy the Clarke regularity. Therefore, we introduce the QNSTR with fixed smoothing parameter μ to a inexact solution of (1.1.1). In the next section, we will study the complexity of the QNSTR with fixed smoothing parameter to receive an inexact solution of (1.1.1) with tolerance error ϵ.

2.3.2 Complexity analysis of the QNSTR to an inexact solution of VIs

In this subsection, we analyze the complexity of the QNSTR with fixed smoothing parameter μ converging to an inexact solution of (1.1.1) with ϵ tolerance. Since μ is a fixed scalar, we inherit the simplified notations of the discussion for fixed smoothing parameter in subsection 2.3.1. We use $\tilde{r}(z)$, $\tilde{F}(z)$ and $\nabla\tilde{r}(z)$ to represent $\tilde{r}(z, \mu)$, $\tilde{F}(z, \mu)$ and $\nabla\tilde{r}(z, \mu)$ respectively.

We first give the definition of ϵ-solution of (1.1.1).

Definition 2.4 (ϵ-solution of (1.1.1))**.** *For given $\epsilon > 0$, a point z is called an ϵ-solution of* (1.1.1)*, if $\|F(z)\| \leq \epsilon$.*

Next, we propose the inexact-QNSTR algorithm with fixed smoothing parameter μ to converge to an ϵ-solution of (1.1.1) and study its complexity.

Algorithm 2 The inexact-QNSTR Algorithm

Input: $\bar{\Delta} > 0$, $\Delta_0 \in (0, \bar{\Delta})$, $0 < \beta_1 < 1 < \beta_2$, $0 \leq \eta < \zeta_1 < \zeta_2 \leq 1$, tolerance parameter $\delta > 0$, $\epsilon > 0$, smoothing parameter μ in suitable range, $z_0 \in \mathcal{Z}$.

1: **while** $k < K_{\max}$ **do**
2: If $\|F(z_k)\| \leq \epsilon$ or $\|g_k\| \leq \delta$, terminate.
3: Otherwise, go to Step 1.
4: **Step 1:** Solve (2.2.7) for α_k.
5: Compute the reduction ratio at iterate k:

$$\rho_k = \frac{\tilde{r}(z_k) - \tilde{r}(z_k + V_k\alpha_k)}{m_k(0) - m_k(\alpha_k)}. \tag{2.3.44}$$

6: **Step 2:** Update Δ_{k+1} as

$$\Delta_{k+1} = \begin{cases} \beta_1\Delta_k, & \text{if } \rho_k < \zeta_1, \\ \min\{\beta_2\Delta_k, \bar{\Delta}\}, & \text{if } \rho_k > \zeta_2 \text{ and } \|V_k\alpha_k\| = \Delta_k, \\ \Delta_k, & \text{otherwise.} \end{cases} \tag{2.3.45}$$

7: **Step 3:** Update

$$\bar{z}_k = \begin{cases} z_k + V_k\alpha_k, & \text{if } \rho_k > \eta, \\ z_k, & \text{otherwise.} \end{cases} \tag{2.3.46}$$

8: **Step 4:** Update $z_{k+1} = \bar{z}_k$ and let $k = k + 1$.
9: **end while**

The complexity analysis of the inexact-QNSTR algorithm is based on the following assumption.

Assumption 2.1. *Let $\{z_k\}_{k=0}^{\infty}$ be the sequence generated by Algorithm 2 and $\mathcal{Z}_*$ be the set of all accumulation points of $\{z_k\}_{k=0}^{\infty}$. We assume that, for all $z^* \in \mathcal{Z}_*$, we have $\nabla\tilde{F}(z^*)$ is nonsingular.*

Let $z^* \in \mathcal{Z}_*$, based on the above assumption and the continuity of $\nabla\tilde{F}$, we know for all z in a neighbourhood of z^*, $\nabla\tilde{F}(z)$ is nonsingular and $\|\nabla\tilde{F}(z)^{-1}\| \leq C_1$ for some $C_1 > 0$. Thus

$$\|\tilde{F}(z)\| = \|\nabla(\tilde{F}(z)^\top)^{-1}\nabla\tilde{r}(z)\| \leq \|\nabla\tilde{F}(z)^{-1}\|\|\nabla\tilde{r}(z)\| \leq C_1\|\nabla\tilde{r}(z)\|. \tag{2.3.47}$$

Moreover, according to (i) in Lemma 2.1, we know

$$\|F(z)\| - \|\tilde{F}(z)\| \leq \|F(z) - \tilde{F}(z)\| \leq \frac{\sqrt{n}}{8}\mu. \tag{2.3.48}$$

Combining (2.3.47) and (2.3.48), we have

$$\|F(z)\| \leq C_1\delta + \frac{\sqrt{n}}{8}\mu.$$

If Algorithm 2 is terminated by $\|F(z_k)\| \leq \epsilon$, then z_k is an ϵ-solution of (1.1.1). If Algorithm 2 is terminated by $\|g_k\| \leq \delta$, z_k can be guaranteed as an ϵ-solution of (1.1.1) by properly selected parameters δ and μ such that $C_1\delta + \frac{\sqrt{n}}{8}\mu \leq \epsilon$. In particular, if $\mu\sqrt{n} \ll \epsilon$, finding an ϵ-solution of (1.1.1) is essentially equivalent to finding a point that satisfies $\|g_k\| \leq \delta$ with $\delta = \epsilon/C_1$. This implies if we set μ in a suitable range, we can approximately regard that finding an ϵ-solution of (1.1.1) and finding a point that satisfies $\|g_k\| \leq \epsilon$ have the same complexity. Next, we will analyze the complexity of Algorithm 2 for (1.1.1) by examining the number of iterations k required for Algorithm 2 to produce a point z_k that satisfies $\|g_k\| \leq \epsilon$.

Lemma 2.8. *Let $\{z_k\}_{k=0}^{\infty}$ be the sequence generated by Algorithm 2 with fixed smoothing parameter $\mu \in (0, \min_{1\leq i\leq n}(1, u_i - l_i))$ and given tolarence parameter $\epsilon > 0$. Then there exists a constant $\bar{C} \geq 1$ such that $\|\nabla\tilde{r}(z) - \nabla\tilde{r}(z')\| \leq \frac{\bar{C}}{\mu}\|z - z'\|$ for any $z, z' \in \mathcal{S}(R_0)$ and $\|H_k\| \leq \bar{C}$ for $k \in \mathbb{N}$. Moreover, the searching radius Δ_{k+1} satisfies the following inequality*

$$\frac{\beta_1(1-\zeta_1)\mu\|g_k\|}{3\bar{C}} \leq \Delta_{k+1}, \qquad \forall k \in \mathbb{N}. \tag{2.3.49}$$

Proof. First of all, by the conclusions in Lemma 2.4 and (2.3.23), there exists $\bar{C} = \max\{1, C, M\}$.

Next, we will prove that $\Delta_{k+1} \geq \frac{\beta_1(1-\zeta_1)\mu\|g_k\|}{3\bar{C}}$ for all $k \in \mathbb{N}$. Substituting the Lipschitz constant $\frac{\bar{C}}{\mu}$ of $\nabla\tilde{r}(\cdot)$ and the upper bound $\bar{C}$ of $\|H_k\|$ into (2.3.33), we immediately have

$$|\rho_k - 1| \leq \frac{(\frac{\bar{C}}{\mu} + \frac{\bar{C}}{2})\Delta_k^2}{\frac{1}{2}\|g_k\|\min(\Delta_k, \|g_k\|/\bar{C})} \leq \frac{\frac{3\bar{C}}{\mu}\Delta_k^2}{\|g_k\|\min(\Delta_k, \|g_k\|/\bar{C})}.$$

The second inequaliy holds because of the setting that $\mu \leq 1$.

Denote

$$\tilde{\Delta}_k := \frac{(1-\zeta_1)\mu\|g_k\|}{3\bar{C}}.$$

For any $\Delta_k \leq \tilde{\Delta}_k$, we have

$$\Delta_k \leq \tilde{\Delta}_k = \frac{(1-\zeta_1)\mu\|g_k\|}{3\bar{C}} \leq \frac{\|g_k\|}{3\bar{C}} < \frac{\|g_k\|}{\bar{C}}.$$

Then

$$|\rho_k - 1| \leq \frac{\frac{3\bar{C}}{\mu}\Delta_k^2}{\|g_k\|\min(\Delta_k, \|g_k\|/\bar{C})} = \frac{3\bar{C}\Delta_k^2}{\mu\|g_k\|\Delta_k} = \frac{3\bar{C}\Delta_k}{\mu\|g_k\|} \leq \frac{3\bar{C}\tilde{\Delta}_k}{\mu\|g_k\|} \leq 1 - \zeta_1,$$

which implies $\rho_k \geq \zeta_1$. The above observation together with update rules in Algorithm 2 indicates that

$$\Delta_{k+1} \geq \beta_1\tilde{\Delta}_k = \frac{\beta_1(1-\zeta_1)\mu\|g_k\|}{3\bar{C}}. \tag{2.3.50}$$

Because if $\Delta_{k+1} < \beta_1\tilde{\Delta}_k$ implies that $\Delta_k < \tilde{\Delta}_k$ and $\Delta_{k+1} \geq \tilde{\Delta}_k$, which causes a contradiction. Thus, we proved (2.3.49). □

Lemma 2.9. *At iteration $k \geq 1$, let α_k be the solution of (2.2.7). if $\rho_k > \eta$, we have the following amount of decrease on $m_k(\alpha_k)$:*

$$\tilde{r}(z_{k+1}) - \tilde{r}(z_k) \leq -\frac{\eta\beta_1(1-\zeta_1)\mu\|g_k\|\min(\|g_k\|, \|g_{k-1}\|)}{3\bar{C}}. \tag{2.3.51}$$

Proof. According to Lemma 2.5 and the update rule of z_k, we have:

$$\tilde{r}(z_{k+1}) - \tilde{r}(z_k) \leq \eta(m_k(\alpha_k) - m_k(0)) \leq -\frac{\eta}{2}\|g_k\| \min\left(\Delta_k, \frac{\|g_k\|}{\|H_k\|}\right). \tag{2.3.52}$$

If $\Delta_k \leq \frac{\|g_k\|}{\|H_k\|}$, by the boundedness $\Delta_k \geq \beta_1(1-\zeta_1)\mu\frac{\|g_{k-1}\|}{3\bar{C}}$ given in Lemma 2.8, we have that

$$\begin{aligned} m_k(\alpha_k) - m_k(0) &\leq -\frac{1}{2}\|g_k\|\Delta_k \\ &\leq -\frac{1}{2}\|g_k\|\beta_1(1-\zeta_1)\mu\frac{\|g_{k-1}\|}{3\bar{C}} \\ &\leq -\frac{1}{2}\beta_1\frac{(1-\zeta_1)\mu\|g_k\|\|g_{k-1}\|}{3\bar{C}}. \end{aligned} \tag{2.3.53}$$

If $\frac{\|g_k\|}{\|H_k\|} < \Delta_k$, we have

$$\begin{aligned} m_k(\alpha_k) - m_k(0) &\leq -\frac{1}{2}\|g_k\|\frac{\|g_k\|}{\bar{C}} \\ &< -\frac{1}{2}\beta_1\frac{(1-\zeta_1)\mu\|g_k\|^2}{3\bar{C}}. \end{aligned} \tag{2.3.54}$$

The second inequality holds because of the fact that $\beta_1, \zeta_1, \mu < 1$. Combining (2.3.52), (2.3.53) and (2.3.54), then we complete the proof. □

Lemma 2.10. *Let $\{(\Delta_k, \rho_k)\}_{k=1}^{\infty}$ be the sequence of radius generated by Algorithm 2. Denote*

$$\mathcal{K} := \{k \in \mathbb{N} |\ \rho_k \geq \eta\}$$

and let k_i be the i-th element in $\mathcal{K}$ in increasing order. Let $k_0 = 0$ and divide the sequence into groups $\{(\Delta_k, \rho_k)\}_{k=k_0}^{k_1}$, $\{(\Delta_k, \rho_k)\}_{k=k_1}^{k_2}$, $\cdots$. Then we know $k_{i+1} - k_i \leq \lceil \log_{\beta_1}\left(\frac{(1-\zeta_1)\mu\|g_{k_i}\|}{3\bar{C}\Delta}\right)\rceil + 2$ for any $i \in \mathbb{N}$.

Proof. We give the proof by contradiction. Suppose that there exist an index i such that $\{(\Delta_k, \rho_k)\}_{k=k_i}^{k_{i+1}-1}$, and $k_{i+1} - k_i > \lceil \log_{\beta_1}(\frac{(1-\zeta_1)\mu\|g_{k_i}\|}{3\bar{C}\Delta})\rceil + 2$. This implies

$\rho_k < \eta < \zeta_1$ for all $k \in [k_i, k_{i+1} - 1]$. According to the update rules of Δ_k and z_k, we know $g_k = g_{k_i}$ for all $k \in [k_i, k_{i+1} - 1]$ and $\Delta_{k_{i+1}-1} = \beta_1^{k_{i+1}-k_i-1}\Delta_{k_i}$. Moreover, since $\Delta_{k_i} \leq \bar{\Delta}$, we have

$$\Delta_{k_{i+1}-1} \leq \frac{\beta_1(1-\zeta_1)\mu\|g_{k_i}\|}{3\bar{C}} = \frac{\beta_1(1-\zeta_1)\mu\|g_{k_{i+1}-2}\|}{3\bar{C}}.$$

However, this is contradicted with the conclusion in Lemma 2.8, then we completed the proof. □

Theorem 2.3. *Under the parameters setting in Algorithm 2, the QNSTR runs at most* $\mathcal{O}(\log_{\beta_1}(\frac{(1-\zeta_1)\epsilon^2}{3\bar{C}\bar{\Delta}})\frac{\frac{3}{2}n\bar{C}(R+\frac{\sqrt{n}}{8})^2}{\eta\beta_1(1-\zeta_1)}\epsilon^{-3})$ *iterations to reach an iterate* z_{k+1} *that satisfies an* ϵ*-first order stationary point.*

Proof. Let K be the iteration for sequence $\{z_k\}_{k=0}^{\infty}$ that first time reaches an ϵ-first order stationary point. This implies that $\|g_k\| > \epsilon$ for all $k \leq K$. Lemma 2.9 indicates that the residual function $\tilde{r}$ has an amount of decrease once the QNSTR (in Algorithm 2) succeed in updating, that is

$$\tilde{r}(z_{k+1}) - \tilde{r}(z_k) \leq -\frac{\eta\beta_1(1-\zeta_1)\mu\|g_k\|\min(\|g_k\|, \|g_{k-1}\|)}{3\bar{C}} \leq -\frac{\eta\beta_1(1-\zeta_1)\mu\epsilon^2}{3\bar{C}}. \tag{2.3.55}$$

And Lemma 2.10 indicates that in each $\lceil \log_{\beta_1}\left(\frac{(1-\zeta_1)\mu\|g_{k_i}\|}{3\bar{C}\bar{\Delta}}\right)\rceil + 2$ iterations, one of the iterations for successful updating must exist. Moreover, the total amount of decrease cannot exceed $\frac{n}{2}(R + \frac{\sqrt{n}}{8})^2$ because

$$\tilde{r}(z_0, \mu) - \tilde{r}_{\inf} \leq \tilde{r}(z_0, \mu) \leq \frac{n}{2}(R + \frac{\sqrt{n}}{8})^2 \tag{2.3.56}$$

for any $z_0 \in \mathcal{Z}$. Combining the results of Lemma 2.9, Lemma 2.10 and (2.3.56), we immediately know

$$K \leq \log_{\beta_1}(\frac{(1-\zeta_1)\mu\epsilon}{3\bar{C}\bar{\Delta}})\frac{\frac{3}{2}n\bar{C}(R+\frac{\sqrt{n}}{8})^2}{\eta\beta_1(1-\zeta_1)\mu}\epsilon^{-2}.$$

Moreover, by the dicussion in last section, $\sqrt{n}\mu \ll \epsilon$. Let $\mu = \frac{\epsilon}{C_2\sqrt{n}}$ with C_2 being a large given constant. Since $\beta_1 < 1$, then we can conclude the complexity of the QNSTR in Algorithm 2 is $O(\ln(\epsilon^{-2})\epsilon^{-3})$. $\square$

Chapter 3

Application of the QNSTR on nonconvex-nonconcave minimax problems

In this chapter, QNSTR is applied to solve the nonconvex-nonconcave minimax problems.

3.1 Nonconvex-nonconcave minimax problems

Minimax optimization problems have a wide range of applications in game theory [42], distributional robustness optimization [43], robust machine learning [44], Generative Adversarial Networks (GANs) [45], reinforcement learning [46], distributed optimization [47], among others.

Mathematically, a convexly constrained minimax optimization problem can be formulated as follows:

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}}\ f(x,y) := \mathbb{E}_P\left[\ell(x,y,\xi)\right], \tag{3.1.1}$$

where $\mathcal{X} \subseteq \mathbb{R}^{m_1}$ and $\mathcal{Y} \subseteq \mathbb{R}^{m_2}$ are nonempty, closed and convex sets, ξ is an s-dimensional random vector obeying the probability distribution P with support set Ξ, $\ell : \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \times \mathbb{R}^s \to \mathbb{R}$ is nonconvex-nonconcave for fixed ξ, i.e., $\ell(x, y, \xi)$ is

neither convex with respect to x nor concave with respect to y. Hence the objective function $f(x, y)$ is also nonconvex-nonconcave in general.

Due to the nonconvexity-nonconcavity of the objective function f, problem (3.1.1) may not have a saddle point. Consequently, the concept of global or local saddle points are untimely to characterize the optimality of problem (3.1.1). Recently, motivated by practical applications, the so-called global and local minimax points are proposed to describe the global and local optima of nonconvex-nonconcave minimax optimization problems in [48] from the viewpoint of sequential games. Moreover, the optimality necessary condition for a local minimax point is studied in [48] for unconstrained minimax optimization problems. In [49, 50], the optimality condition for a local minimax point is studied for constrained minimax optimization problems.

Numerical methods for minimax optimization problems have been extensively studied. These algorithms can be broadly categorized into four classes based on the convexity or concavity of problems: (i) the convex-concave case (see, e.g., [14, 15, 51, 52]), (ii) the nonconvex-concave case (see, e.g., [53, 54, 55]), (iii) the convex-nonconcave case (see, e.g., [53, 54, 55]) and (iv) the nonconvex-nonconcave case (see, e.g., [56, 57]).

To solve (3.1.1) numerically, we first apply the SAA approach to obtain a discrete form. We collect N independent identically distributed (i.i.d.) samples of ξ (e.g. generated by the Monte Carlo method), denoted by $\xi^1, \cdots, \xi^N$, The resulting of a discrete counterpart of (3.1.1) is then given by:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \ \hat{f}_N(x, y) := \frac{1}{N}\sum_{i=1}^{N} \ell(x, y, \xi^i). \tag{3.1.2}$$

3.2 Sample Average Approximation (SAA) and asymptotic convergence

In this section, we focus on the asymptotic convergence between problems (3.1.2) and (3.1.1) regarding to the global minimax point and the first-order stationary point. To this end, we first give some preliminaries on how to describe the optima of a minimax optimization problem.

Definition 3.1 (global and local minimax points, [48, Definitions 9 & 14])**.**

(i) $(\hat{x}, \hat{y}) \in \mathcal{X} \times \mathcal{Y}$ *is called a* global minimax point *of problem* (3.1.1)*, if*

$$f(\hat{x}, y) \leq f(\hat{x}, \hat{y}) \leq \max_{y' \in \mathcal{Y}} f(x, y'),\ \forall (x, y) \in \mathcal{X} \times \mathcal{Y}.$$

(ii) $(\hat{x}, \hat{y}) \in \mathcal{X} \times \mathcal{Y}$ *is called a* local minimax point *of problem* (3.1.1)*, if there exist a* $\delta_0 > 0$ *and a function* $\tau : \mathbb{R}_+ \to \mathbb{R}_+$ *satisfying* $\tau(\delta) \to 0$ *as* $\delta \to 0$*, such that for any* $\delta \in (0, \delta_0]$ *and any* $(x, y) \in \mathcal{X} \times \mathcal{Y}$ *satisfying* $\|x - \hat{x}\| \leq \delta$ *and* $\|y - \hat{y}\| \leq \delta$*, we have*

$$f(\hat{x}, y) \leq f(\hat{x}, \hat{y}) \leq \max_{y' \in \{y \in \mathcal{Y} : \|y - \hat{y}\| \leq \tau(\delta)\}} f(x, y').$$

Remark 3.1. *The concept of saddle points has been commonly used to characterize the optima of minimax problems. A point* $(\hat{x}, \hat{y}) \in \mathcal{X} \times \mathcal{Y}$ *is called a* saddle point *of problem* (3.1.1)*, if*

$$f(\hat{x}, y) \leq f(\hat{x}, \hat{y}) \leq f(x, \hat{y}),\ \forall (x, y) \in \mathcal{X} \times \mathcal{Y}, \tag{3.2.1}$$

and $(\hat{x}, \hat{y}) \in \mathcal{X} \times \mathcal{Y}$ *is called a* local saddle point *of problem* (3.1.1) *if there exists* $\delta > 0$ *such that*

$$f(\hat{x}, y) \leq f(\hat{x}, \hat{y}) \leq f(x, \hat{y}),\ \forall (x, y) \in \mathcal{X} \cap \mathcal{B}(\hat{x}, \delta) \times \mathcal{Y} \cap \mathcal{B}(\hat{y}, \delta). \tag{3.2.2}$$

However, as pointed out in [48], saddle points and local saddle points may not exist in many applications of machine learning, especially in the nonconvex-nonconcave case. Also, (local) saddle points are solutions from the viewpoint of simultaneous game, where the minimization operator and the maximization operator act simultaneously. However, many applications, such as GANs and adversarial training, seek for solutions in the sense of sequential game, where the minimization operator acts first and the maximization operator acts latter. The global and local minimax points exist under some mild conditions (see [48, Proposition 11 and Lemma 16]) and also describe the optima in the sense of sequential game.

The following lemma gives the first-order and second-order necessary optimality conditions of local minimax points for problem (3.1.1) when $\mathcal{X}$ and $\mathcal{Y}$ are boxes, which is a special form of [50, Theorem 3.2 & Corollary 3.1].

Lemma 3.1. *Let $\mathcal{X} = [a, b]$ and $\mathcal{Y} = [c, d]$, where $a < b$ with $a, b \in \mathbb{R}^{m_1}$ and $c < d$ with $c, d \in \mathbb{R}^{m_2}$. Assume that f is continuously differentiable and the tuple $(\hat{x}, \hat{y}) \in \mathcal{X} \times \mathcal{Y}$ is a local minimax point of problem* (3.1.1).

(i) Then

$$\begin{cases} 0 \in \nabla_x f(\hat{x}, \hat{y}) + \mathcal{N}_{\mathcal{X}}(\hat{x}), \\ 0 \in -\nabla_y f(\hat{x}, \hat{y}) + \mathcal{N}_{\mathcal{Y}}(\hat{y}), \end{cases} \tag{3.2.3}$$

where $\mathcal{N}_{\mathcal{X}}(\hat{x}) = \mathcal{N}_{[a_1,b_1]}(\hat{x}_1) \times \cdots \times \mathcal{N}_{[a_{m_1},b_{m_1}]}(\hat{x}_{m_1})$ and $\mathcal{N}_{\mathcal{Y}}(\hat{y}) = \mathcal{N}_{[c_1,d_1]}(\hat{y}_1) \times \cdots \times \mathcal{N}_{[c_{m_2},d_{m_2}]}(\hat{y}_{m_2})$ and for $i = 1, \cdots, m_1$ and $j = 1, \cdots, m_2$,

$$\mathcal{N}_{[a_i,b_i]}(\hat{x}_i) = \begin{cases} [0, +\infty), & \hat{x}_i = b_i, \\ \{0\}, & a_i < \hat{x}_i < b_i, \\ (-\infty, 0], & \hat{x}_i = a_i \end{cases} \text{and} \mathcal{N}_{[c_j,d_j]}(\hat{y}_j) = \begin{cases} [0, +\infty), & \hat{y}_j = d_j, \\ \{0\}, & c_j < \hat{y}_j < d_j, \\ (-\infty, 0], & \hat{y}_j = c_j. \end{cases}$$

(ii) If f is twice continuously differentiable, then

$$\begin{cases} \langle w_1, \nabla^2_{xx} f(\hat{x},\hat{y}) w_1\rangle \geq 0, \ \forall w_1 \in \bigcup_{\delta>0} \left(\bigcap_{y' \in \mathbb{B}(\hat{y},\delta)} \Gamma_1(\hat{x}, y') \right), \\ \left\langle w_2, \nabla^2_{yy} f(\hat{x},\hat{y}) w_2 \right\rangle \leq 0, \ \forall w_2 \in \Gamma_2(\hat{x},\hat{y}), \end{cases} \tag{3.2.4}$$

where

$$\Gamma_1(\hat{x}, y') := \{w_1 \in \mathcal{T}_{\mathcal{X}}(\hat{x}) : w_1 \perp \nabla_x f(\hat{x}, y')\},$$

$$\Gamma_2(\hat{x}, \hat{y}) := \{w_2 \in \mathcal{T}_{\mathcal{Y}}(\hat{y}) : w_2 \perp \nabla_y f(\hat{x}, \hat{y})\},$$

$\mathcal{T}_{\mathcal{X}}(\hat{x}) = \mathcal{T}_{[a_1,b_1]}(\hat{x}_1) \times \cdots \times \mathcal{T}_{[a_{m_1},b_{m_1}]}(\hat{x}_{m_1})$, $\mathcal{T}_{\mathcal{Y}}(\hat{y}) = \mathcal{T}_{[c_1,d_1]}(\hat{y}_1) \times \cdots \times \mathcal{T}_{[c_{m_2},d_{m_2}]}(\hat{y}_{m_2})$,

$$\mathcal{T}_{[a_i,b_i]}(\hat{x}_i) = \begin{cases} (-\infty, 0], & \hat{x}_i = b_i, \\ (-\infty, +\infty), & a_i < \hat{x}_i < b_i, \\ [0, +\infty), & \hat{x}_i = a_i \end{cases} \text{ and } \mathcal{T}_{[c_j,d_j]}(\hat{y}_j) = \begin{cases} (-\infty, 0], & \hat{y}_j = d_j, \\ (-\infty, +\infty), & c_j < \hat{y}_j < d_j, \\ [0, +\infty), & \hat{y}_j = c_j. \end{cases}$$

Assume, further, that $\tau(\delta) = \mathcal{O}(\delta)$ where $\tau(\cdot)$ is defined in Definition 3.1. Then the first assertion in (3.2.4) *can be replaced by*

$$\left\langle w_1, \nabla^2_{xx} f(\hat{x},\hat{y}) w_1 \right\rangle \geq 0, \ \forall w_1 \in \Gamma_1(\hat{x},\hat{y}). \tag{3.2.5}$$

In what follows, we tacitly assume that f is continuously differentiable, $\mathcal{X} = [a, b]$ and $\mathcal{Y} = [c, d]$ are defined in Lemma 3.1.

Definition 3.2. *$(\hat{x},\hat{y}) \in \mathcal{X} \times \mathcal{Y}$ is called a* first-order stationary point *of problem* (3.1.1) *if* (3.2.3) *holds. If, in addition,* (3.2.4) *holds, $(\hat{x},\hat{y})$ is called a* second-order stationary point *of problem* (3.1.1). *Specially, if $(\hat{x},\hat{y})$ satisfies both* (3.2.4) *and* (3.2.5)*, we call such point a* strong second-order stationary point. *$(\hat{x},\hat{y}) \in \mathcal{X} \times \mathcal{Y}$ is called a* first-order stationary point*,* second-order stationary point*,* strong second-order stationary point *of problem* (3.1.2) *if* (3.2.3) *holds,* (3.2.3)*,*(3.2.4) *holds,* (3.2.3)*,* (3.2.4)*,* (3.2.5) *hold with replacing f by $\hat{f}_N$, respectively.*

Hereafter, we will focus on finding a first-order stationary point of (3.1.2).

As for the exponential rate of convergence of the first-order and second-order stationary points of a specific GAN, one can refer to [50, Proposition 4.3]. In what follows, we mainly focus on the almost surely convergence analysis between problems (3.1.1) and (3.1.2). If the problem is well-behaved and the global minimax points are achievable, we will consider the convergence of global minimax points between problems (3.1.1) and (3.1.2). Otherwise, at best, the first-order and second-order stationary points (Definition 3.2) can be calculated. Thus, we will also consider the convergence of first-order and second-order stationary points between problems (3.1.1) and (3.1.2) as N tends to infinity.

Denote the optimal value, the set of global minimax points, the set of first-order stationary points and the set of second-order stationary points of problem (3.1.1) by ϑ_g, $\mathcal{S}_g$ and $\mathcal{S}_{1\mathrm{st}}$, $\mathcal{S}_{2\mathrm{ed}}$, respectively. Let $\widehat{\vartheta}_g^N$, $\widehat{\mathcal{S}}_g^N$, $\widehat{\mathcal{S}}_{1\mathrm{st}}^N$ and $\widehat{\mathcal{S}}_{2\mathrm{ed}}^N$ denote the optimal value, the set of global minimax points, the set of first-order stationary points and the set of second-order stationary points of problem (3.1.2), respectively.

Lemma 3.2. *Suppose that: (a) $\mathcal{X}$ and $\mathcal{Y}$ are compact sets; (b) $\ell(x, y, \xi)$ is dominated by an integrable function for every $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Then*

$$\sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \left|\hat{f}_N(x, y) - f(x, y)\right| \to 0$$

w.p.1 as $N \to \infty$.

If, further, (c) $\|\nabla_x \ell(x, y, \xi)\|$ and $\|\nabla_y \ell(x, y, \xi)\|$ are dominated by an integrable function for every $(x, y) \in \mathcal{X} \times \mathcal{Y}$, then

$$\sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \left\|\nabla \hat{f}_N(x, y) - \nabla f(x, y)\right\| \to 0$$

w.p.1 as $N \to \infty$.

If (d) $\|\nabla^2_{xx} \ell(x, y, \xi)\|$ and $\left\|\nabla^2_{yy} \ell(x, y, \xi)\right\|$ are dominated by an integrable function

for every $(x, y) \in \mathcal{X} \times \mathcal{Y}$*, then*

$$\sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \left\| \nabla^2_{xx} \hat{f}_N(x, y) - \nabla^2_{xx} f(x, y) \right\| \to 0,$$

$$\sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \left\| \nabla^2_{yy} \hat{f}_N(x, y) - \nabla^2_{yy} f(x, y) \right\| \to 0$$

w.p.1 as $N \to \infty$*.*

Proof. Since the samples are i.i.d. and $\mathcal{X}$ and $\mathcal{Y}$ are compact, it is known from [58, Theorem 7.53] that the above uniform convergence results hold. □

The following proposition gives the nonemptiness of $\widehat{\mathcal{S}}^N_g$, $\mathcal{S}_g$, $\widehat{\mathcal{S}}^N_{\rm 1st}$ and $\mathcal{S}_{\rm 1st}$.

Proposition 3.1. *If conditions (a)-(c) in Lemma 3.2 hold, then* $\mathcal{S}_g$ *and* $\mathcal{S}_{\rm 1st}$ *are nonempty and* $\widehat{\mathcal{S}}^N_g$ *and* $\widehat{\mathcal{S}}^N_{\rm 1st}$ *are nonempty for any* $N \in \mathbb{N}_+$*.*

Proof. Since the continuity of $f(x, y)$ and $\hat{f}_N(x, y)$ w.r.t. (x, y) and the boundedness of $\mathcal{X}$ and $\mathcal{Y}$, we know from [48, Proposition 11] the nonemptiness of $\mathcal{S}_g$ and $\widehat{\mathcal{S}}^N_g$. Note that both $\mathcal{S}_{\rm 1st}$ and $\widehat{\mathcal{S}}^N_{\rm 1st}$ are solution sets of VI$(\mathcal{Z}, H)$. Then we have from [7, Corollary 2.2.5] that $\mathcal{S}_{\rm 1st}$ and $\widehat{\mathcal{S}}^N_{\rm 1st}$ are nonempty. □

Based on the uniform laws of large numbers in Lemma 3.2, we have the following convergence results.

Theorem 3.1. *Let conditions (a)-(c) in Lemma 3.2 hold. Then*

$$\mathrm{d}\left(\widehat{\mathcal{S}}^N_g, \mathcal{S}_g\right) \to 0, \tag{3.2.6}$$

$$\mathrm{d}\left(\widehat{\mathcal{S}}^N_{\rm 1st}, \mathcal{S}_{\rm 1st}\right) \to 0 \tag{3.2.7}$$

w.p.1 as $N \to \infty$*. If, further, (d) in Lemma 3.2 is satisfied; (ii)* $\mathcal{S}_{\rm 2ed}$ *and* $\widehat{\mathcal{S}}^N_{\rm 2ed}$ *(for sufficiently large* N*) are nonempty, then*

$$\mathrm{d}\left(\widehat{\mathcal{S}}^N_{\rm 2ed}, \mathcal{S}_{\rm 2ed}\right) \to 0 \tag{3.2.8}$$

w.p.1 as $N \to \infty$.

Proof. (3.2.7) follows from [59, Proposition 19] directly. Thus, in what follows, we only consider (3.2.6) and (3.2.8). We first vertify (3.2.6). From Proposition 3.1, we know that $\widehat{\mathcal{S}}_g^N$ and $\mathcal{S}_g$ are nonempty for any $N \in \mathbb{N}_+$. Let $z^N = (x^N, y^N) \in \widehat{\mathcal{S}}_g^N$ and $z^N \to \bar{z} = (\bar{x}, \bar{y})$ w.p.1 as $N \to \infty$. Then we just verify that $\bar{z} \in \mathcal{S}_g$ w.p.1. If $\{z^N\}$ is not a convergent sequence, due to the boundedness of $\mathcal{X}$ and $\mathcal{Y}$, we can choose a convergent subsequence. Denote $\varphi(x) := \max_{y\in\mathcal{Y}} f(x,y)$ and $\hat{\varphi}_N(x) := \max_{y\in\mathcal{Y}} \hat{f}_N(x,y)$. Note that

$$\begin{aligned} \max_{x\in\mathcal{X}} |\hat{\varphi}_N(x) - \varphi(x)| &= \max_{x\in\mathcal{X}} \left| \max_{y\in\mathcal{Y}} \hat{f}_N(x,y) - \max_{y\in\mathcal{Y}} f(x,y) \right| \\ &\le \max_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \left| \hat{f}_N(x,y) - f(x,y) \right| \\ &\to 0 \end{aligned} \tag{3.2.9}$$

w.p.1 as $N \to \infty$, where the last convergence assertion follows from Lemma 3.2.

Next, we show

$$\mathrm{Proj}_x \widehat{\mathcal{S}}_g^N = \arg\min_{x\in\mathcal{X}} \hat{\varphi}_N(x) \quad \text{and} \quad \mathrm{Proj}_x \mathcal{S}_g = \arg\min_{x\in\mathcal{X}} \varphi(x), \tag{3.2.10}$$

where Proj_x denotes the projection onto the x's space. Based on the definition of global minimax points, we have, for any $(\hat{x}, \hat{y}) \in \mathcal{S}_g$, that

$$f(\hat{x}, y) \le f(\hat{x}, \hat{y}) \le \max_{y'\in\mathcal{Y}} f(x, y'), \ \forall (x,y) \in \mathcal{X} \times \mathcal{Y},$$

which implies

$$\varphi(\hat{x}) = \max_{y\in\mathcal{Y}} f(\hat{x}, y) \le \max_{y'\in\mathcal{Y}} f(x, y') = \varphi(x), \ \forall x \in \mathcal{X}.$$

This means $\mathrm{Proj}_x \mathcal{S}_g \subseteq \arg\min_{x\in\mathcal{X}} \varphi(x)$. On the other hand, for any $\hat{x} \in \arg\min_{x\in\mathcal{X}} \varphi(x)$, we let $\hat{y} \in \arg\max_{y\in\mathcal{Y}} f(\hat{x}, y)$. Then it is not difficult to examine that $(\hat{x}, \hat{y})$ is a global

minimax point, i.e., $\arg\min_{x\in\mathcal{X}}\varphi(x)\subseteq \mathrm{Proj}_x\mathcal{S}_g$. The $\mathrm{Proj}_x\widehat{\mathcal{S}}_g^N=\arg\min_{x\in\mathcal{X}}\hat{\varphi}_N(x)$ can be similarly verified. Hence (3.2.10) holds.

Then (3.2.9) and (3.2.10) indicate, according to [60, Lemma 4.1], that

$$\mathrm{d}\left(\mathrm{proj}_x\widehat{\mathcal{S}}_g^N,\mathrm{proj}_x\mathcal{S}_g\right)\to 0 \tag{3.2.11}$$

w.p.1 as $N\to\infty$. We know from (3.2.11) that $\bar{x}\in\mathrm{Proj}_x\mathcal{S}_g$.

Moreover, we know that

$$\begin{aligned}\left|\widehat{\vartheta}_g^N-\vartheta_g\right|&=\left|\min_{x\in\mathcal{X}}\hat{\varphi}_N(x)-\min_{x\in\mathcal{X}}\varphi(x)\right|\\&\le\max_{x\in\mathcal{X}}|\hat{\varphi}_N(x)-\varphi(x)|\\&\to 0\end{aligned}$$

w.p.1 as $N\to\infty$, where ϑ_g and $\widehat{\vartheta}_g^N$ are optimal values of problems (3.1.1) and (3.1.2), respectively. Due to Lemma 3.2 and the continuity of f, we know that

$$\begin{aligned}\left|\hat{f}_N(x^N,y^N)-f(\bar{x},\bar{y})\right|&\le\left|\hat{f}_N(x^N,y^N)-f(x^N,y^N)\right|+\left|f(x^N,y^N)-f(\bar{x},\bar{y})\right|\\&\to 0.\end{aligned}$$

Since $\widehat{\vartheta}_g^N=\hat{f}_N(x^N,y^N)$, we know that $\vartheta_g=f(\bar{x},\bar{y})$, which, together with $\bar{x}\in\mathrm{Proj}_x\mathcal{S}_g$, implies that $(\bar{x},\bar{y})\in\mathcal{S}_g$.

Next, we focus on verifying (3.2.8). Analogously, let $z^N=(x^N,y^N)\in\widehat{\mathcal{S}}_{2\mathrm{ed}}^N$ and $z^N\to\bar{z}=(\bar{x},\bar{y})$ w.p.1 as $N\to\infty$, and we verify that $\bar{z}\in\mathcal{S}_{2\mathrm{ed}}$ w.p.1. Note that $z^N\in\widehat{\mathcal{S}}_{2\mathrm{ed}}^N$ is equivalent to

$$\begin{cases}\left\langle w_1,\nabla_{xx}^2\hat{f}_N(x^N,y^N)w_1\right\rangle\ge 0,\ \forall w_1\in\Gamma_{1,N}(x^N,y^N),\\ \left\langle w_2,\nabla_{yy}^2\hat{f}_N(x^N,y^N)w_2\right\rangle\le 0,\ \forall w_2\in\Gamma_{2,N}(x^N,y^N),\end{cases} \tag{3.2.12}$$

where

$$\begin{aligned}\Gamma_{1,N}(x^N,y^N):=&\{w_1\in\mathcal{T}_{\mathcal{X}}(x^N):w_1\perp\nabla_x\hat{f}_N(x^N,y^N)\},\\ \Gamma_{2,N}(x^N,y^N):=&\{w_2\in\mathcal{T}_{\mathcal{Y}}(y^N):w_2\perp\nabla_y\hat{f}_N(x^N,y^N)\}.\end{aligned}$$

Since both $\Gamma_{1,N}(x^N, y^N)$ and $\Gamma_{2,N}(x^N, y^N)$ are cones, (3.2.12) can be equivalently rewritten as

$$\begin{cases} \left\langle w_1, \nabla^2_{xx} \hat{f}_N(x^N, y^N) w_1 \right\rangle \geq 0, \ \forall w_1 \in \widetilde{\Gamma}_{1,N}(x^N, y^N), \\ \left\langle w_2, \nabla^2_{yy} \hat{f}_N(x^N, y^N) w_2 \right\rangle \leq 0, \ \forall w_2 \in \widetilde{\Gamma}_{2,N}(x^N, y^N), \end{cases}$$

where $\widetilde{\Gamma}_{1,N}(x^N, y^N) := \Gamma_{1,N}(x^N, y^N) \cap \mathbb{B}$, $\widetilde{\Gamma}_{2,N}(x^N, y^N) := \Gamma_{2,N}(x^N, y^N) \cap \mathbb{B}$. Assume, by contradiction, that there exist $\bar{w}_1 \in \widetilde{\Gamma}_1(\bar{x}, \bar{y}) := \Gamma_1(\bar{x}, \bar{y}) \cap \mathbb{B}$ and $\bar{w}_2 \in \widetilde{\Gamma}_2(\bar{x}, \bar{y}) := \Gamma_2(\bar{x}, \bar{y}) \cap \mathbb{B}$ where $\Gamma_1(\bar{x}, \bar{y})$ and $\Gamma_2(\bar{x}, \bar{y})$ are defined in Lemma 3.1, such that

$$\left\langle \bar{w}_1, \nabla^2_{xx} f(\bar{x}, \bar{y}) \bar{w}_1 \right\rangle < 0 \text{ or } \left\langle \bar{w}_2, \nabla^2_{yy} f(\bar{x}, \bar{y}) \bar{w}_2 \right\rangle > 0. \tag{3.2.13}$$

Note in Lemma 3.1 that $\mathcal{T}_{\mathcal{X}}(\bar{x}) \subseteq \mathcal{T}_{\mathcal{X}}(x^N)$ and $\mathcal{T}_{\mathcal{Y}}(\bar{y}) \subseteq \mathcal{T}_{\mathcal{Y}}(y^N)$ for sufficiently large N. Moreover, due to the convergence results in Lemma 3.2, $\{w_1 \in \mathbb{R}^{m_1} : w_1 \perp \nabla_x \hat{f}_N(x^N, y^N)\} \cap \mathbb{B}$ converges to $\{w_1 \in \mathbb{R}^{m_1} : w_1 \perp \nabla_x f(\bar{x}, \bar{y})\} \cap \mathbb{B}$ in the sense of Hausdorff distance, and $\{w_2 \in \mathbb{R}^{m_2} : w_2 \perp \nabla_y \hat{f}_N(x^N, y^N)\} \cap \mathbb{B}$ converges to $\{w_2 \in \mathbb{R}^{m_2} : w_2 \perp \nabla_y f(\bar{x}, \bar{y})\} \cap \mathbb{B}$ in the sense of Hausdorff distance. Therefore, for any $\epsilon > 0$, there exists N_0 (depending only on ϵ) such that

$$\widetilde{\Gamma}_1(\bar{x}, \bar{y}) \subseteq \widetilde{\Gamma}_{1,N}(x^N, y^N) + \epsilon \mathbb{B} \text{ and } \widetilde{\Gamma}_2(\bar{x}, \bar{y}) \subseteq \widetilde{\Gamma}_{2,N}(x^N, y^N) + \epsilon \mathbb{B}$$

for any $N \geq N_0$. Thus, we have

$$\bar{w}_1 \in \widetilde{\Gamma}_1(\bar{x}, \bar{y}) \subseteq \widetilde{\Gamma}_{1,N}(x^N, y^N) + \epsilon \mathbb{B}$$

and

$$\bar{w}_2 \in \widetilde{\Gamma}_2(\bar{x}, \bar{y}) \subseteq \widetilde{\Gamma}_{2,N}(x^N, y^N) + \epsilon \mathbb{B},$$

which implies that

$$\bar{w}_1 = w_1^N + \epsilon u_1^N \text{ and } \bar{w}_2 = w_2^N + \epsilon u_2^N,$$

where $w_1^N \in \widetilde{\Gamma}_1(\bar{x}, \bar{y})$, $w_2^N \in \widetilde{\Gamma}_2(\bar{x}, \bar{y})$ and $u_1^N, u_2^N \in \mathbb{B}$. Notice that

$$\begin{cases} \left\langle w_1^N, \nabla_{xx}^2 \hat{f}_N(x^N, y^N) w_1^N \right\rangle \geq 0, \\ \left\langle w_2^N, \nabla_{yy}^2 \hat{f}_N(x^N, y^N) w_2^N \right\rangle \leq 0, \end{cases}$$

which implies that

$$\begin{cases} \left\langle \bar{w}_1 - \epsilon u_1^N, \nabla_{xx}^2 \hat{f}_N(x^N, y^N)(\bar{w}_1 - \epsilon u_1^N) \right\rangle \geq 0, \\ \left\langle \bar{w}_2 - \epsilon u_2^N, \nabla_{yy}^2 \hat{f}_N(x^N, y^N)(\bar{w}_2 - \epsilon u_2^N) \right\rangle \leq 0. \end{cases}$$

By letting $N \to \infty$, due to the arbitrariness of ϵ, we obtain

$$\begin{cases} \langle \bar{w}_1, \nabla_{xx}^2 f(\bar{x}, \bar{y}) \bar{w}_1 \rangle \geq 0, \\ \left\langle \bar{w}_2, \nabla_{yy}^2 f(\bar{x}, \bar{y}) \bar{w}_2 \right\rangle \leq 0, \end{cases}$$

which contradicts with (3.2.13). Thus, (3.2.8) holds. □

Based on Theorem 3.1, it is well-founded for us to employ problem (3.1.2) to approximately solve problem (3.1.1). In the sequel, we will focus on how to compute an ϵ-first-order stationary point of problem (3.1.2).

Let $z := (x^\top, y^\top)^\top \in \mathbb{R}^n$, $\mathcal{Z} := \mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^n$ and $n = m_1 + m_2$. Suppose $\hat{f}_N$ is continuously differentiable. Let

$$H_N(z) := \begin{pmatrix} \nabla_x \hat{f}_N(x, y) \\ -\nabla_y \hat{f}_N(x, y) \end{pmatrix}.$$

Then the first-order optimality condition for a local minimax point of problem (3.1.2) can be presented as the following variational inequality

$$0 \in H_N(z) + \mathcal{N}_{[l,u]}(z), \tag{3.2.14}$$

where $l, u \in \mathbb{R}^n$ with $l = (a^\top, c^\top)^\top$ and $u = (b^\top, d^\top)^\top$. We call z^* a first-order stationary point of problem (3.1.2) if it satisfies (3.2.14).

The VI (3.2.14) can be equivalently reformulated as

$$F_N(z) := z - \text{Proj}_{[l,u]}(z - H_N(z)) = 0, \tag{3.2.15}$$

where the closed form of $\text{Proj}_{[l,u]}(\cdot)$ is given in (1.2.6).

Obviously, z^* is a first-order stationary point of (3.1.2) if it is an optimal solution of the following least-squares problem:

$$\min_{z\in\mathbb{R}^n} \; r_N(z) := \frac{1}{2}\|F_N(z)\|^2 \tag{3.2.16}$$

and $r_N(z^*) = 0$. We can apply (2.1.8) to approximate (3.2.15). Then, we can find a first-order stationary point of (3.1.2) via solving the smoothing minimization problem in (2.1.10).

3.3 Numerical experiments

In this section, we report some numerical results by using the QNSTR algorithm for finding an ϵ-first-order stationary point of problem (3.1.2). Also, we compare the QNSTR algorithm with several state-of-the-art algorithms for minimax problems. All of the numerical experiments in this thesis are implemented using TensorFlow 1.13.1, Python 3.6.9 and Cuda 10.0 on a server with 1 Tesla P100-PCIE GPU (16 GB memory, 1.3285 GHz) and an operating system of 64 bits in the University Research Facility in Big Data Analytics (UBDA) of the Hong Kong Polytechnic University. (UBDA website: https://www.polyu.edu.hk/ubda/.)

We evaluate the performance of our algorithm with two problems. A logistic regression minimax problem with synthetic data and a GAN-based image generation problem using MNIST hand-writing data. In these experiments, we use notation QNSTR(L) to denote that in the QNSTR algorithm, the dimension of the subspace spanned by the columns of V_k is L, and we test the efficiency of the QNSTR(L) under

different choices of L and directions $\{d_k^i\}_{i=1}^{L-1}$. Next, we investigate different choices of the subspace spanned by the columns of V_k. Specifically, denote

$$V_k^z = [-g_k, (z_k - z_{k-1}), \cdots, (z_{k-L+2} - z_{k-L+1})],$$
$$V_k^F = [-g_k, F(z_k), \cdots, F(z_{k-L+2})],$$
$$V_k^g = -[g_k, g_{k-1}, \cdots, g_{k-L+1}],$$
$$V_k^{z,H} = [-g_k, (z_k - z_{k-1}), H(z_k), \cdots, (z_{k-L'+2} - z_{k-L'+1}), H(z_{k-L'+2})],$$

for $L \geq 2$ and $L' \geq 2$. Note that the choice of L' is related to the subspace dimension L by the equation $L = 2L' - 1$.

3.3.1 Logistic regression min-max problems

We consider the following nonconvex-nonconcave minimax problem

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f_\mu^R(x,y) := c(x,y) + \lambda_1 \sum_{i=1}^{m_1} s(x_i,\mu) - \lambda_2 \sum_{j=1}^{m_2} s(y_j,\mu) \tag{3.3.1}$$

with

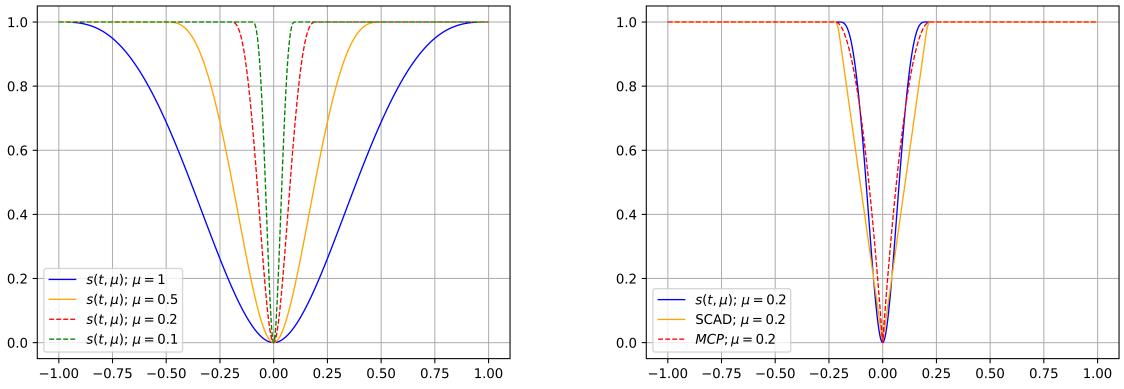
$$c(x,y) := \sum_{k=1}^{N} \log(1 + e^{-\alpha_k a_k^\top x}) + x^\top A y - \sum_{k=1}^{N} \log(1 + e^{-\beta_k b_k^\top y}),$$

where $\mathcal{X} := [-1,1]^{m_1}$, $\mathcal{Y} := [-1,1]^{m_2}$, $a_k \in \{0,1\}^{m_1}$, $b_k \in \{0,1\}^{m_2}$, $A \in \{0,1\}^{m_1\times m_2}$ and $\alpha_k, \beta_k \in \{-1,1\}$ for all $k \in [N]$.

$$s(t,\mu) := \begin{cases} 1, & \text{if } |t| > \mu, \\ 3(\frac{t}{\mu})^4 - 8\text{sign}(t)(\frac{t}{\mu})^3 + 6(\frac{t}{\mu})^2, & \text{otherwise}, \end{cases} \tag{3.3.2}$$

is a smoothing approximation form of ℓ_0 norm, and we report the approximation results in Figure 3.1. Thus, it can be regarded as a smoothing relaxation model of the following sparse min-max model [24]

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y) := c(x,y) + \lambda_1\|x\|_0 - \lambda_2\|y\|_0. \tag{3.3.3}$$



(a) (b)

Figure 3.1: Results for the smoothing function $s(\cdot, \mu)$: (a) $s(\cdot, \mu)$ in (3.3.2) with different values of μ; (b) comparison of $s(\cdot, \mu)$ in (3.3.2) and SCAD [1], MCP [2] with $\mu = 0.2$.

Although (3.3.1) is neither convex nor concave, the following proposition establishes the existence of the saddle point for (3.3.1).

Proposition 3.2. *There exists a $\bar{\mu} > 0$ such that* (3.3.1) *has a local saddle point for any $\mu \in (0, \bar{\mu})$.*

Proof. Obviously, we know that the saddle point of $\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} c(x, y)$ exists since $c(\cdot, \cdot)$ is convex for x with respect to fixed y and concave for y with respect to fixed x. Let (x^*, y^*) be a saddle point of $\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} c(x, y)$, i.e. (3.2.2) holds. Denote

$$\mathcal{A}(x^*) := \{i|\ |x_i^*| \neq 0\} \quad \text{and} \quad \mathcal{B}(y^*) := \{j|\ |y_j^*| \neq 0\}.$$

Let $\vartheta = \min\{1, |x_i^*|, |y_j^*| : i \in \mathcal{A}(x^*), j \in \mathcal{B}(y^*)\}$ and set $\bar{\mu} = \vartheta$, then $|x_i^*| \geq \bar{\mu}$, $|y_j^*| \geq \bar{\mu}$, $\forall i \in \mathcal{A}(x^*), j \in \mathcal{B}(y^*)$. Choose $\mu \in (0, \bar{\mu})$. Obviously, there exists a $\delta_1 > 0$, such that for any $i \in \mathcal{A}(x^*)$, $j \in \mathcal{B}(y^*)$, $x \in \mathcal{B}(x^*, \delta_1)$ and $y \in \mathcal{B}(y^*, \delta_1)$, it holds $|x_i| > \mu$ and $|y_j| > \mu$, since $s(t, \mu) = \|t\|_0$, $\forall t \in (-\infty, -\mu] \cup [\mu, +\infty)$, we know that

$$s(x_i, \mu) = 1 \qquad \text{and} \qquad s(y_j, \mu) = 1.$$

This means that, for any $x \in \mathcal{B}(x^*, \delta_1)$ and $y \in \mathcal{B}(y^*, \delta_1)$,

$$\sum_{i=1}^{m_1} s(x_i^*, \mu) \leq \sum_{i=1}^{m_1} s(x_i, \mu) \qquad \text{and} \qquad \sum_{j=1}^{m_2} s(y_j^*, \mu) \leq \sum_{j=1}^{m_2} s(y_j, \mu). \tag{3.3.4}$$

Together (3.2.2) with (3.3.4), we obtain

$$f_\mu^R(x^*, y) \leq f_\mu^R(x^*, y^*) \leq f_\mu^R(x, y^*), \qquad x \in \mathcal{B}(x^*, \delta) \cap \mathcal{X}, y \in \mathcal{B}(y^*, \delta) \cap \mathcal{Y},$$

which means that (x^*, y^*) is a local saddle point of (3.3.1). □

Let $z = (x^\top, y^\top)^\top$, $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, $H(z) = \begin{pmatrix} \nabla_x f_\mu^R(x, y) \\ -\nabla_y f_\mu^R(x, y) \end{pmatrix}$, we say $z^* = ((x^*)^\top, (y^*)^\top)^\top$ a first-order stationary point of (3.3.1) if

$$\langle H(z^*), z - z^* \rangle \geq 0, \quad \forall z \in \mathcal{Z}. \tag{3.3.5}$$

The synthetic data sets, chosen independently for each problem setting, indexed by m_1, m_2, and N, where each entry is sampled (with respect to the binary choice listed) with probability $p = 0.5$. We compare the performance of our QNSTR algorithm, with $L = 5$ for V_k^z, V_k^F, V_k^g and $V_k^{z,H}$, against the alternating Adam method. The parameters in Algorithm 2 are set as follows: $\bar{\Delta} = 100$, $\Delta_0 = 1$, $\beta_1 = 0.5$, $\beta_2 = 5$, $\zeta_1 = 0.2$, $\zeta_2 = 0.5$ and $\eta = 0.1$. The parameter $\bar{\epsilon}$ in (2.2.4) and (2.2.5) is chosen as $\bar{\epsilon} = 10^{-4}$. We apply limit memory strategy with $L_1 = 20$ in these experiments. To determine the optimal step-size for Adam, we perform a grid search over the estimates $\{0.005, 0.001, 0.0005\}$ and select the optimal one among all parameter choices. We present our results in Table 3.1 and Table 3.2 with different initialization of x and y. For each problem setting among combinations of $(m_1, m_2, N) \in \{500, 1000, 2000\} \times \{500, 1000, 2000\} \times \{2000, 5000, 10000\}$, where $m_1 \geq m_2$. Each experiment with individual setting was repeated 5 times, and the average results were taken. In Table 3.1, we report the comparison of the performance of the QNSTR with $V_k^{z,H}$ (the most effective choice of V_k) and Adam with

Time to reach $\|F(x)\| \leq 10^{-10}$						
	N=2000		N=5000		N=10000	
m_1/m_2	Adam	QNSTR	Adam	QNSTR	Adam	QNSTR
500/500	22.6s	**19.7s**	**59.2s**	83.6s	224.1s	**192.6s**
1000/500	37.9s	**22.1s**	339.1s	**180.3s**	340.4s	**222.4s**
1000/1000	51.0s	**29.8s**	231.2s	**128.5s**	450.8s	**336.0s**
2000/500	144.9s	**81.2s**	338.1s	**229.3s**	683.2s	**340.0s**
2000/1000	104.6s	**86.1s**	398.7s	**268.5s**	702.7s	**368.1s**
2000/2000	148.2s	**141.4s**	475.2s	**276.2s**	1305.7s	**392.5s**

Table 3.1: Performance of the QNSTR algorithm and alternating Adam with $x_0 = 0.2e$, $y_0 = 0.2e$.

Time to reach $\|F(x)\| \leq 10^{-10}$						
	N=2000		N=5000		N=10000	
m_1/m_2	Adam	QNSTR	Adam	QNSTR	Adam	QNSTR
500/500	218.9s	**43.0s**	277.6s	**163.1s**	590.2s	**201.8s**
1000/500	624.0s	**50.5s**	408.5s	**174.4s**	802.6s	**427.9s**
1000/1000	722.4s	**89.2s**	682.1s	**159.2s**	1155.9s	**727.7s**
2000/500	**576.9s**	844.6s	608.8s	**477.2s**	734.2s	**517.9s**
2000/1000	1142.6s	**991.0s**	1098.7s	**468.5s**	1868.2s	**982.4s**
2000/2000	1676.1s	**1284.2s**	2817.1s	**584.2s**	4177.8s	**1210.5s**

Table 3.2: Performance of the QNSTR algorithm and alternating Adam with $x_0 = 0.8e$, $y_0 = 0.8e$.

initial points $x_0 = 0.2e \in \mathcal{X}$, $y_0 = 0.2e \in \mathcal{Y}$. In Table 3.2, we report the comparison of the performance of the QNSTR with $V_k^{z,H}$ (the most effective choice of V_k) and Adam with initial points $x_0 = 0.8e \in \mathcal{X}$, $y_0 = 0.8e \in \mathcal{Y}$. In each table in 3.1 and 3.2, we report the time (in seconds) before achieving a residual value $\|F(z)\|$ less than 10^{-10}. Furthermore, Tables 3.3 and 3.4 provide a comparison of the sparsity of the solutions obtained by the QNSTR and Adam under the two kinds of initial point setting. Our results collectively indicate that the QNSTR outperforms Adam in solving the logistic regression minimax problem, demonstrating its superior efficiency.

Sparsity of the solution						
	N=2000		N=5000		N=10000	
m_1/m_2	Adam	QNSTR	Adam	QNSTR	Adam	QNSTR
500/500	28.2/42.2	28.2/42.2	26.0/22.8	27.6/22.4	22.4/16.2	22.4/16.2
1000/500	80.8/37.6	80.8/37.6	39.0/18.4	39.0/18.4	39.6/14.2	39.6/14.2
1000/1000	65.4/72.6	65.4/72.6	51.8/51.6	51.8/51.6	33.6/36.2	33.6/36.2
2000/500	119.8/27.4	123.8/27.4	84.6/19.0	89.8/19.2	73.4/17.2	73.4/17.2
2000/1000	163.0/76.4	171.4/58.2	161.2/81.2	161.2/81.2	168.6/74.6	168.6/74.6
2000/2000	167.4/152.2	167.4/152.2	104.8/107.6	104.4/112.4	109.8/103.0	109.8/103.0

Table 3.3: Number of elements whose absolute value is less or equal to 10^{-5} in stationary point found by the QNSTR algorithm and alternating Adam for (3.3.1) with $x_0 = 0.2e$, $y_0 = 0.2e$.

Sparsity of the solution						
	N=2000		N=5000		N=10000	
m_1/m_2	Adam	QNSTR	Adam	QNSTR	Adam	QNSTR
500/500	32.8/35.2	19.6/23.2	31.6/26.6	27.0/27.4	21.4/19.2	21.4/19.2
1000/500	77.0/37.4	82.4/37.4	47.0/25.0	47.0/25.2	30.0/21.2	30.2/24.6
1000/1000	73.2/75.4	79.4/73.6	48.2/57.2	48.2/57.2	34.6/36.0	34.6/36.0
2000/500	123.0/32.2	121.4/28.2	86.4/22.2	79.2/18.4	84.6/19.0	84.6/19.0
2000/1000	153.8/72.0	141.4/78.2	112.8/82.0	111.2/82.2	78./40.8	78./40.8
2000/2000	147.4/145.8	160.8/147.6	104.4/108.8	105.4/110.2	74.8/81.2	74.8/77.6

Table 3.4: Number of elements whose absolute value is less or equal to 10^{-5} in stationary point found by the QNSTR algorithm and alternating Adam for (3.3.1) with $x_0 = 0.8e$, $y_0 = 0.8e$.

3.3.2 Two-layer GAN on MNIST data

In this subsection, we are mainly interested in problem (3.1.1) arising from GANs [45], which is formulated as follows:

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} f(x,y) := \mathbb{E}_{P_1}\big[\log(D(y,\xi_2))\big] + \mathbb{E}_{P_2}\big[\log(1-D(y,G(x,\xi_1)))\big], \tag{3.3.6}$$

where $\mathcal{X}\subseteq\mathbb{R}^{m_1}$ and $\mathcal{Y}\subseteq\mathbb{R}^{m_2}$ are nonempty, closed and convex sets, ξ_i is $\mathbb{R}^{s_i}$-valued random vector with probability distribution P_i for $i=1,2$, $G:\mathbb{R}^{m_1}\times\mathbb{R}^{s_1}\to\mathbb{R}^{s_2}$ is a generator, $D:\mathbb{R}^{m_2}\times\mathbb{R}^{s_2}\to(0,1)$ is a discriminator. Generally, the generator G and discriminator D are two feedforward neural networks. For example, G can be a

p-layer neural network and D can be a q-layer neural network:

$$G(x,\xi_2) = \sigma_G^p(W_G^p\sigma_G^{p-1}(\cdots\sigma_G^1(W_G^1\xi_2 + b_G^1) + \cdots) + b_G^p),$$
$$D(y,\xi_1) = \sigma_D^q(W_D^q\sigma_D^{q-1}(\cdots\sigma_D^1(W_D^1\xi_1 + b_D^1) + \cdots) + b_D^q),$$

where $W_G^1, \cdots, W_G^p$, $b_G^1, \cdots, b_G^p$ and $W_D^1, \cdots, W_D^q$, $b_D^1, \cdots, b_D^q$ are the weight matrices, biases vectors of G and D with suitable dimensions, $\sigma_G^1, \cdots, \sigma_G^p$ and $\sigma_D^1, \cdots, \sigma_D^q$ are proper activation functions, such as Rectified Linear Unit (ReLU), Gaussian Error Linear Units (GELU), Sigmoid, etc. Denote

$$x := (\mathrm{vec}(W_G^1)^\top, \cdots, \mathrm{vec}(W_G^p)^\top, (b_G^1)^\top, \cdots, (b_G^p)^\top)^\top,$$
$$y := (\mathrm{vec}(W_D^1)^\top, \cdots, \mathrm{vec}(W_D^q)^\top, (b_D^1)^\top, \cdots, (b_D^q)^\top)^\top,$$

where $\mathrm{vec}(\cdot)$ denotes the vectorization operator. Then problem (3.3.6) can be formulated in (3.1.1) if let $\xi := (\xi_1, \xi_2) \in \Xi$ and

$$\ell(x, y, \xi) := \log(D(y, \xi_2)) + \log(1 - D(y, G(x, \xi_1))).$$

Therefore, (3.3.6) is a special case of problem (3.1.1).

To ensure that H_N is continuously differentiable, we employ GELU [61] as the activation function in each hidden layer of both the discriminator D and the generator G. The GELU activation function is defined as:

$$\sigma(x) = x\int_{-\infty}^{x} \frac{e^{\frac{-t^2}{2\Sigma^2}}}{\sqrt{2\pi}\Sigma}\mathrm{d}t$$

with a standard deviation of $\Sigma = 10^{-4}$. In contrast, we use the Sigmoid activation function:

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

in the output layers for both D and G.

In this subsection, we report some preliminary numerical results of the QNSTR algorithm for solving the following SAA counterpart

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}}\frac{1}{N}\sum_{i=1}^{N}\Big(\log(D(y,\xi_2^i))+\log(1-D(y,G(x,\xi_1^i)))\Big) \tag{3.3.7}$$

of problem (3.3.6). We consider a two-layer GAN on MNIST handwritting data, where

$$\xi_1^i\in\mathbb{R}^{100},\ \xi_2^i\in\mathbb{R}^{784},$$

$$W_G^1\in\mathbb{R}^{N_G^1\times 100},\ W_G^2\in\mathbb{R}^{784\times N_G^1},\ W_D^1\in\mathbb{R}^{N_D^1\times 784},\ W_D^2\in\mathbb{R}^{1\times N_D^1}$$

with different choices of dimensions N_G^1 and N_D^1 for hidden outputs. Here $\{\xi_2^i\}$ are generated by an uniform distribution $\mathcal{U}(-1,1)^{100}$. All initial weight matrices W_G^i and W_D^i for $i=1,2$ are randomly generated by using the Gaussian distribution with mean 0 and standard deviation 0.1, and all initial bias vectors b_G^i and b_D^i for $i=1,2$ are set to be zero. $\mathcal{X}$ and $\mathcal{Y}$ are set as $[-1,1]^{m_1}$ and $[-1,1]^{m_2}$, respectively. Among all the experiments in the sequel, the initial point z_0 is the result for running 10000 steps of the alternating Adam with step size 0.0005. The parameters in Algorithm 2 are set as $\bar{\Delta}=100$, $\Delta_0=1$, $\beta_1=0.5$, $\beta_2=2$, $\eta=0.01$, $\zeta_1=0.02$, $\zeta_2=0.05$. The parameter $\bar{\epsilon}$ in (2.2.4) is chosen as $\bar{\epsilon}=10^{-4}$. In practical, an important problem is how to choose a suitable sample size N and smoothing parameter μ. We will first study how to choose N and μ before the generation experiments of MNIST hand-writing data.

We first study the performance of SAA problems under different sample sizes on a GAN model with a two-layer generator with $N_G^1=64$ and a two-layer discriminator with $N_D^1=64$, respectively. We set $\hat{N}=10000$ as the benchmark to approximate the original problem (3.3.6) and let $N=100,500,1000,2000,5000$. For each N, we solve $F_N(z)=0$, 50 times with different samples by using the QNSTR algorithm.

We stop the iteration either

$$\|F_N(z_k)\| \leq 10^{-5} \tag{3.3.8}$$

or the number of iterations exceeds 5000. In these experiments, we set $\mu = 10^{-8}$. We use z_N^* to denote the first point that satisfies (3.3.8) in the iteration for $N = 100, 500, 1000, 2000, 5000$, and we measure its optimality by the residual

$$\text{res}_N := \|z_N^* - \text{mid}\,(l, u, z_N^* - H_{\hat{N}}(z_N^*)\,)\|.$$

Figure 3.2 shows the convergence of res_N to zero as N grows. Table 3.5 presents the average of mean, standard deviation (std) and the width of 95% CI of res_N. It shows that all values decrease as the sample size N increases. Both Figure 3.2 and Table 3.5 validate the convergence results in Section 3.2.

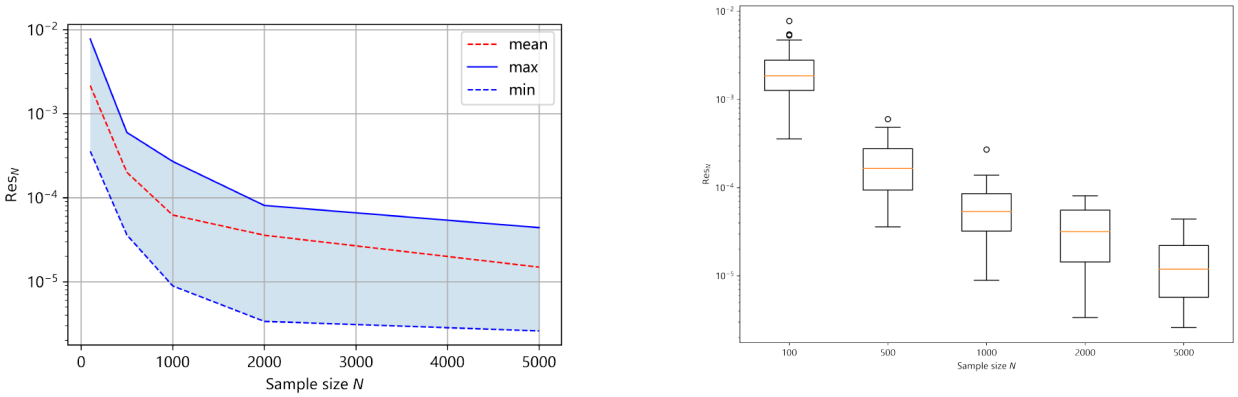


Figure 3.2: The convergence of res_N as N grows (Left: the range of res_N with different N; Right: the boxplot of res_N with different N)

N	100	500	1000	2000	5000
Mean	2.16×10^{-3}	2.01×10^{-4}	6.22×10^{-5}	3.58×10^{-5}	1.49×10^{-5}
std	1.76×10^{-3}	1.65×10^{-4}	4.95×10^{-5}	2.93×10^{-5}	1.18×10^{-5}
95% CI	$[1.76, 2.56]_{\times 10^{-3}}$	$[1.65, 2.36]_{\times 10^{-4}}$	$[4.95, 7.49]_{\times 10^{-5}}$	$[2.93, 4.23]_{\times 10^{-5}}$	$[1.18, 1.79]_{\times 10^{-5}}$

Table 3.5: Means, variances and 95% CIs of res_N with different N

Next, we study how the smoothing parameter μ affects the residual $\|\tilde{F}(z, \mu)\|$. Specifically, we fixed the sample size as $N = 2000$, and we generate 50 test problems

for $\mu = 10^{-t}, t = 1, 2, 4, 5, 6, 8$, respectively. For each μ, we solve problem $\tilde{F}(z, \mu) = 0$ by the QNSTR algorithm. We stop the iteration either condition $\|\tilde{F}(z_k, \mu)\| \leq 10^{-5}$ holds or the number of iterations exceeds 5000. We use z_μ^* to denote the first point that satisfies (3.3.8) in the iteration, and we measure the residual of z_μ^* by

$$\text{res}_\mu := \|z_\mu^* - \text{mid}(l, u, z_\mu^* - H_N(z_\mu^*))\|.$$

The numerical results are presented in Figure 3.3, which shows that res_μ decreases as smoothing parameter μ decreases. In fact, the residual res_μ becomes stable when $\mu \leq 10^{-5}$. We consider problem (3.3.7) with a two-layer discriminator and

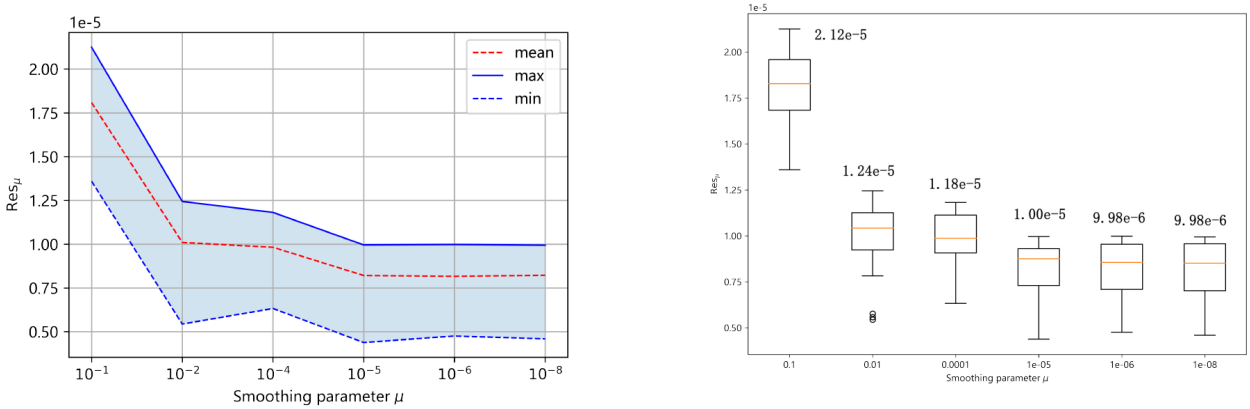


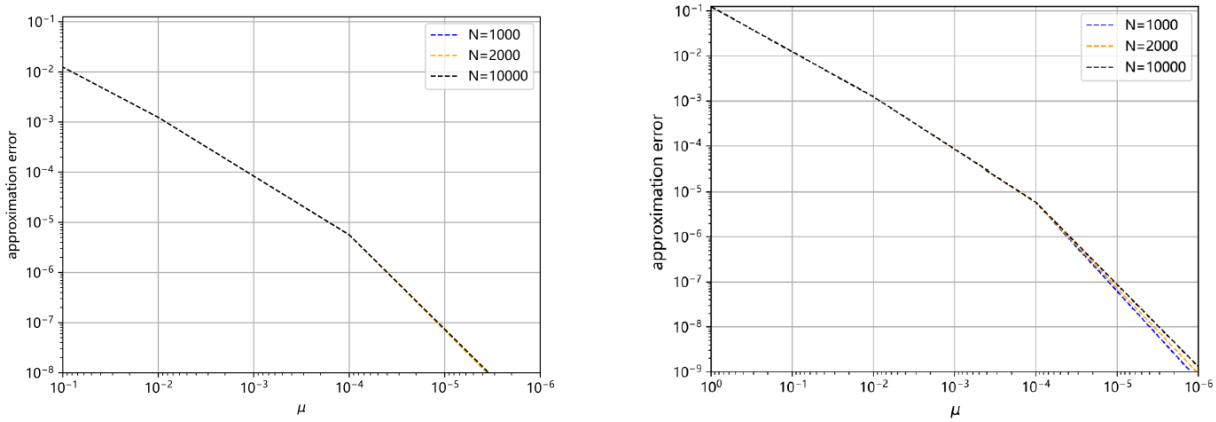
Figure 3.3: The convergence of res_μ as μ decreases (Left: the range of res_μ with different μ; Right: the boxplot of res_μ with different μ).

a two-layer generator using MNIST handwritting data. We set $\mathcal{X} = [-5, 5]^{m_1}$, $\mathcal{Y} = [-5, 5]^{m_2}$, $N_G^1 = N_D^1 = 64$, $N_G^1 = N_D^1 = 128$, $\mu = 10^{-t}, t = 0, 1, \ldots, 6$ and $N = 1000, 2000, 10000$, respectively. Based on the uniform distribution over $[-10, 10]^{m_1+m_2}$, we generate 1000 points $z^i \in [-10, 10]^{m_1+m_2}$, $i = 1, \cdots, 1000$. Denote the "approximation error" by

$$\text{approximation error} := \frac{1}{1000} \sum_{i=1}^{1000} \|F_{N,\mu}(z^i) - F_N(z^i)\|_\infty.$$

In Figure 3.4, we plot the average of "approximation error" under different choices of μ and N with 20 sets of 1000 points in $[-10, 10]^{m_1+m_2}$. From the figure, we can

observe that for each $N = 1000, 2000, 10000$, the "approximation error" converges to zero as μ tends to zero.



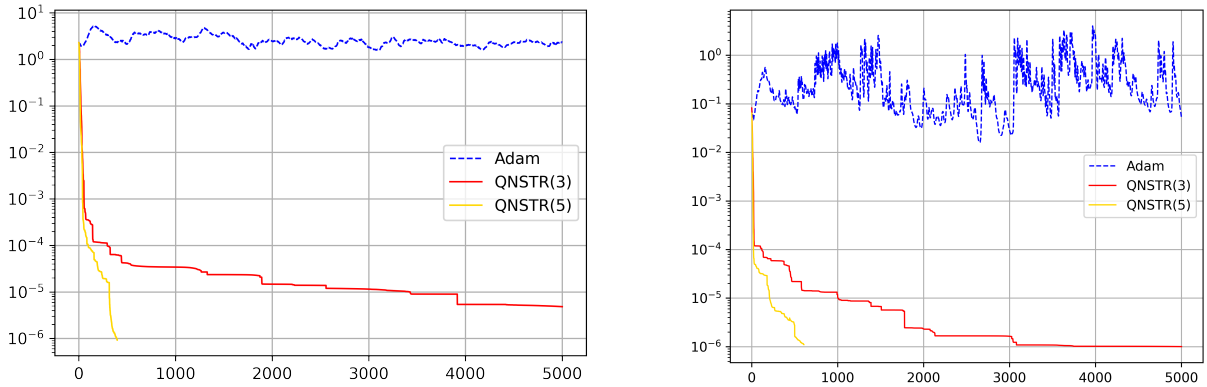
(a) $N_G^1 = N_D^1 = 64$ (b) $N_G^1 = N_D^1 = 128$

Figure 3.4: Smoothing approximation error of $F_{N,\mu}(z)$ to $F_N(z)$ for $N = 1000, 2000, 10000$.

Finally, we report some numerical results to compare the QNSTR algorithm with some commonly-used methods. To this end, we set $N = 2000$ and $\mu = 10^{-8}$. We use $\|F_{N,\mu}(z_k)\|$ to measure performance of these algorithms and apply Frechet Inception Distance (FID) score to measure the quality of image generated by the generator G trained by different algorithms. We terminate all these algorithms when one of the three cases holds: $\|F_{N,\mu}(z_k)\| \leq 10^{-6}$, $\|\nabla F_{N,\mu}(z_k)^\top F_{N,\mu}(z_k)\| \leq 10^{-10}$, the number of iterations exceeds 5000.

We present the numerical results in Figure 3.5 and Figure 3.6. Specially, for highlighting the outperforms of $V_k^{z,H}$, we put itself in Figure 3.5 and put the results of others in Figure 3.6. It is evident from these figures that the QNSTR algorithm performs exceptionally well, and $V_k^{z,H}$ is the optimal choice for the search subspace.

We also compare the QNSTR algorithm with some commonly-used algorithms in training GANs including simultaneous and alternate Gradient Descent-Ascent (GDA) [62, 63, 25], simultaneous and alternate Optimistic Gradient Descent-Ascent

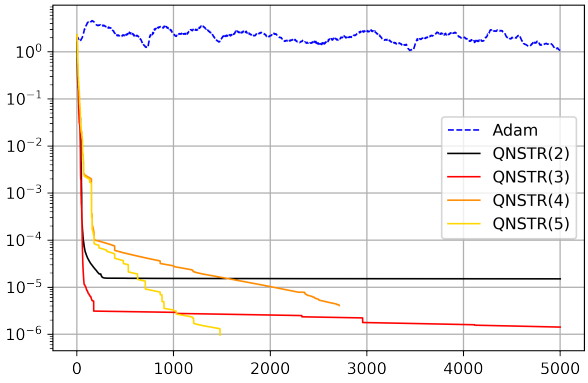


(a) $V_k^{z,H}$ (b) $V_k^{z,H}$

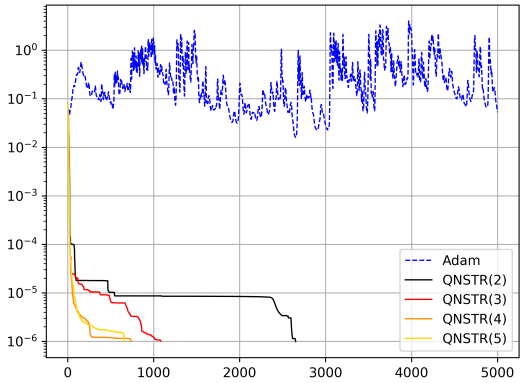
Figure 3.5: The residual $\|F_{N,\mu}(z_k)\|$ with $V_k^{z,H}$ (left: $N_G^1 = N_D^1 = 64$; right: $N_G^1 = N_D^1 = 128$)

(OGDA) [62, 63], γ-alternate Adam, Projected Point Algorithm (PPA) [64, 56]. The initial point z_0 of all these methods are given by alternating Adam with step size 0.0005 and 10000 iterations. We use the grid search for the selection of hyper-parameters in these methods. For simultaneous and alternate GDA and simultaneous and alternate OGDA, we choose the step size $\alpha \in \{0.5, 0.05, 0.005, 0.001, 0.0005\}$. For γ-alternate Adam, we set the step size as 0.0005 and the ratio is choosen in $\gamma \in \{1, 2, 3, 5, 10\}$. For PPA, we use the extragradient method with step size as $\frac{0.1}{2L}$ to solve the subproblem of the PPA at the k-th step, the stopping criteria of the k-th subproblem is set as the residual is less than $\frac{0.01}{k^2}$ or the number of iterations exceeds 100. The comparison results between the QNSTR algorithm and the γ-alternate Adam, the QNSTR algorithm and PPA are given in Figures 3.7 and 3.8, respectively. According to the results, we can see the γ-alternate Adam also shows a outstanding convergence tendency under a suitable hyper-parameter and the QNSTR algorithm can even better than the γ-alternate Adam if a suitable searching space V_k is selected. The PPA performs poorly in these comparisons.

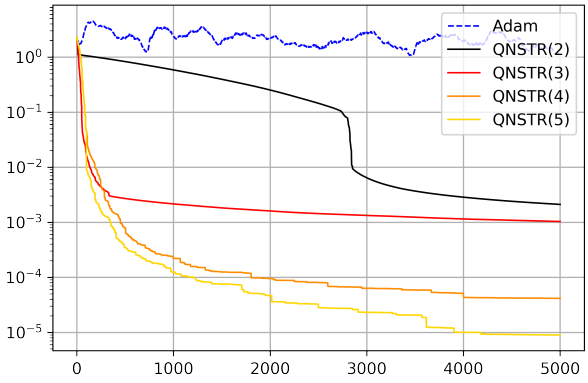
The comparison results between the QNSTR algorithm and simultaneous and



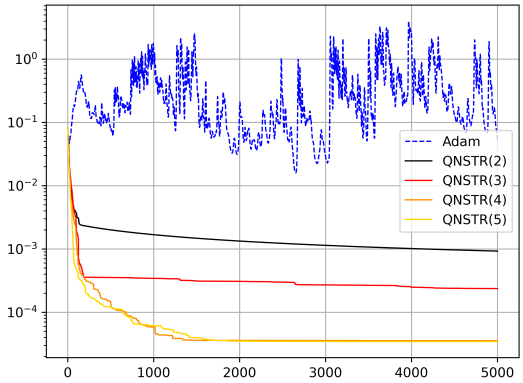
(a) V_k^z



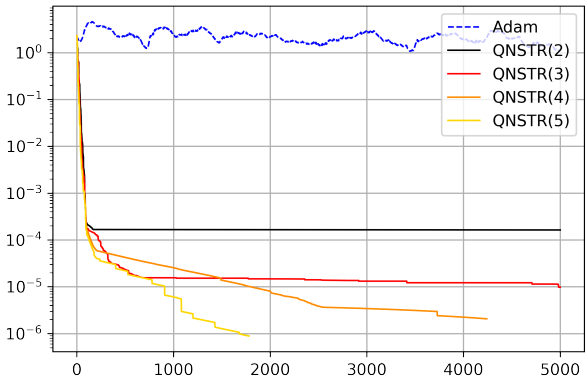
(b) V_k^z



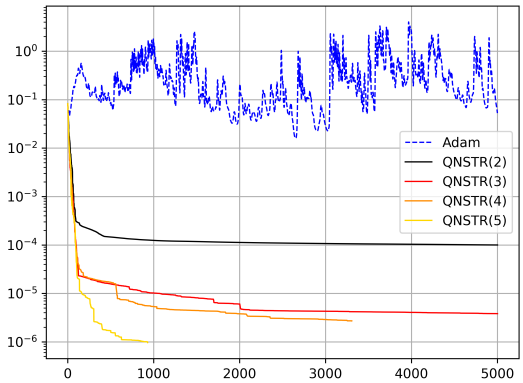
(c) V_k^F



(d) V_k^F

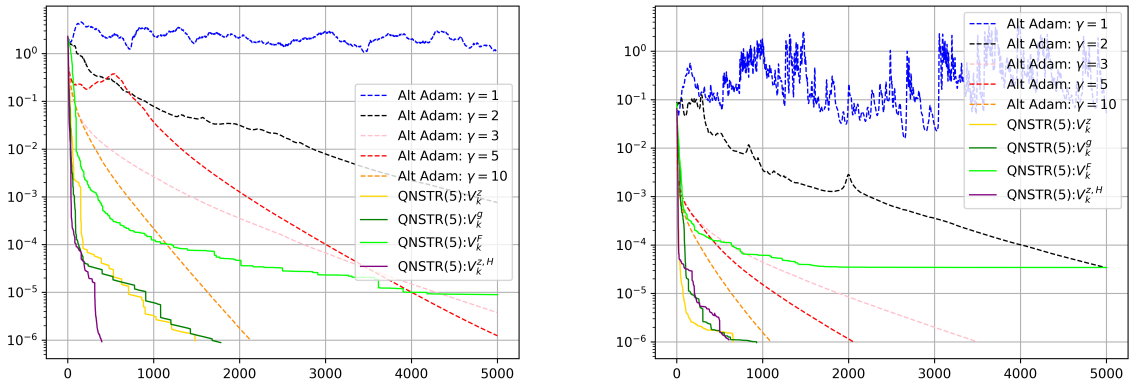


(e) V_k^g



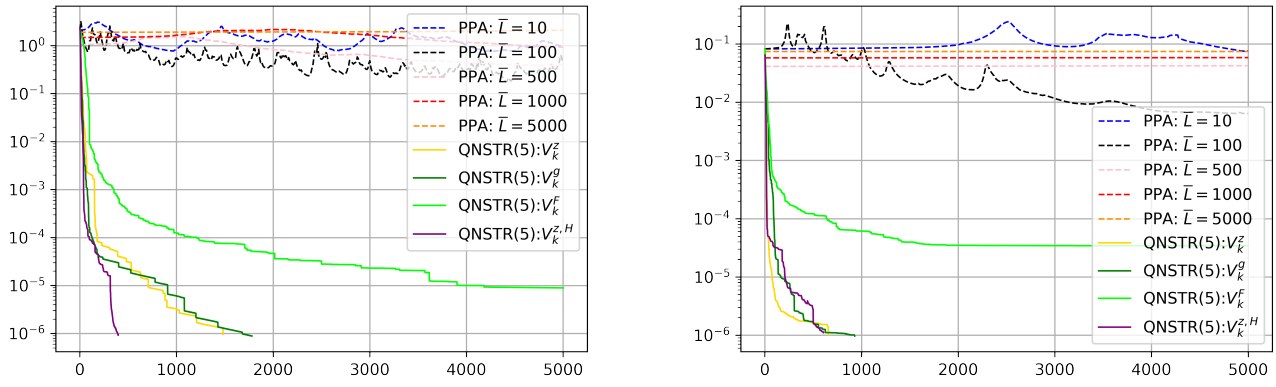
(f) V_k^g

Figure 3.6: The residual $\|F_{N,\mu}(z_k)\|$ with different choice of V_k (left: $N_G^1 = N_D^1 = 64$; right: $N_G^1 = N_D^1 = 128$)



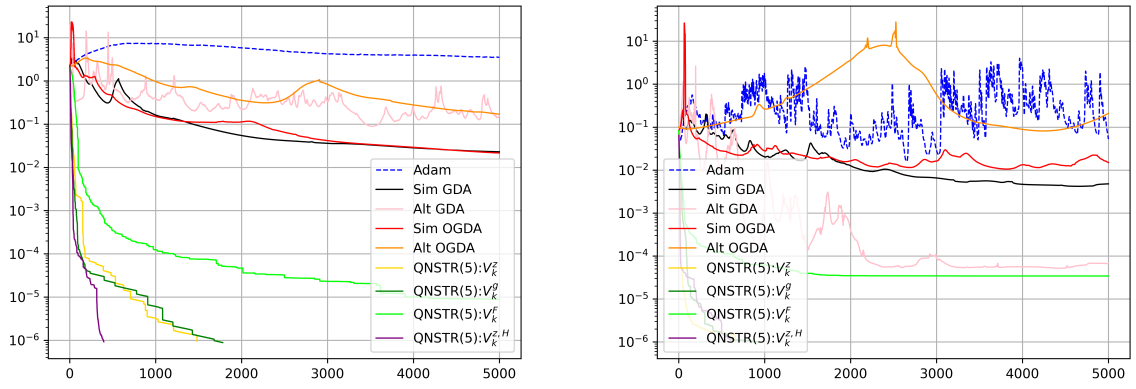
(a) $N_G^1 = N_D^1 = 64$ (b) $N_G^1 = N_D^1 = 128$

Figure 3.7: Comparison results between the QNSTR algorithm for V_k^z, V_k^F, V_k^g, $V_k^{z,H}$ and γ-alternate Adam (left: $N_G^1 = N_D^1 = 64$; right $N_G^1 = N_D^1 = 128$)



(a) $N_G^1 = N_D^1 = 64$ (b) $N_G^1 = N_D^1 = 128$

Figure 3.8: Comparison results between the QNSTR algorithm for V_k^z, V_k^F, V_k^g, $V_k^{z,H}$ and PPA (left: $N_G^1 = N_D^1 = 64$; right $N_G^1 = N_D^1 = 128$)



(a) $N_G^1 = N_D^1 = 64$ (b) $N_G^1 = N_D^1 = 128$

Figure 3.9: Comparison results between the QNSTR algorithm for V_k^z, V_k^F, V_k^g, $V_k^{z,H}$ and γ-alternate Adam (left: $N_G^1 = N_D^1 = 64$; right $N_G^1 = N_D^1 = 128$)

alternate GDA, simultaneous and alternate OGDA are given in Figure 3.9. To show these comparison results more clearly, we only report the results of each method with the optimal stepsize α in the search range.

We can observe from these figures that the QNSTR algorithm outperforms, which validates that the QNSTR algorithm is more efficient in finding an ϵ-first-order stationary point of problem (3.3.7).

We also record the final Frechet Inception Distance (FID) score of each algorithm's output. All results are given in Tables 3.6 and 3.7. They show the generator of GANs trained by the QNSTR algorithm can generate high quality images. In next chapter, we apply the QNSTR on several applications for image segmentation problems to show its feasibility on real problems. Because of its outstanding performance for the experiments in this section, we will apply $V_k^{z,H}$ on these real problems.

QNSTR				
L	2	3	4	5
V_k^z	31.27	32.34	34.48	36.74
V_k^F	31.76	31.01	30.16	33.80
V_k^g	33.97	32.62	34.07	33.13
$V_k^{z,H}$	-	31.76	-	30.01

PPA					
$\overline{L}$	10	100	500	1000	5000
	30.44	31.43	34.88	33.14	34.42

γ-alt Adam					
γ	1	2	3	5	10
	30.13	31.58	32.69	34.55	36.60

sGDA, aGDA, sOGDA, aOGDA					
α	0.5	0.05	0.01	0.005	0.0005
sGDA	33.09	31.89	31.64	31.32	32.52
aGDA	32.21	31.72	31.46	30.64	30.33
sOGDA	30.98	30.93	31.26	32.51	34.06
aOGDA	32.97	32.53	32.53	30.13	33.52

Table 3.6: FID scores of different algorithms with $N_D^1 = N_G^1 = 64$

QNSTR				
L	2	3	4	5
V_k^z	32.33	32.95	31.86	32.01
V_k^F	31.81	30.98	31.02	32.91
V_k^g	32.09	33.17	36.21	31.46
$V_k^{z,H}$	-	28.97	-	30.46

PPA					
$\overline{L}$	10	100	500	1000	5000
	30.18	29.96	32.12	32.13	32.42

γ-alt Adam					
γ	1	2	3	5	10
	27.12	26.98	31.69	33.55	34.60

sGDA, aGDA, sOGDA, aOGDA					
α	0.5	0.05	0.01	0.005	0.0005
sGDA	32.86	31.35	30.98	33.04	33.87
aGDA	32.01	30.31	29.97	31.05	30.17
sOGDA	33.21	31.32	30.98	29.04	29.06
aOGDA	32.97	31.05	29.84	30.42	32.82

Table 3.7: FID scores of different algorithms with $N_D^1 = N_G^1 = 128$

Chapter 4

Image segmentation by mix generative adversarial networks using the QNSTR

In this chapter, we focus on some practical image segmentation problems that are implemented using GANs. Specifically, we apply the QNSTR method to solve these problems.

4.1 Image segmentation problem

Image segmentation is an important component in many visual understanding systems, which is the process of partitioning a digital image into multiple image segments [65]. Image segmentation plays a central role in a broad range of applications [66], including computer aided diagnosis, autonomous vehicles (e.g., navigable surface and pedestrian detection), video surveillance and augmented reality. In recent years, machine learning-based methods have achieved excellent performance in computer vision. Supervised learning based Convolutional Neural Network (CNN) is one of the extensive application approaches for image segmentation. These problems can

be formulated as the following minimization:

$$\min_{x\in\mathcal{X}} \ \frac{1}{N}\sum_{i=1}^{N}\psi\big(\xi_1^i, G(x,\xi_2^i)\big). \tag{4.1.1}$$

In (4.1.1), $\{\xi_1,\xi_2\}_{i=1}^N$ are the N pairs images data with annotations, where $\xi_1^i \in \mathbb{R}^{s_1}$ is the original image and $\xi_2^i \in \mathbb{R}^{s_2}$ is the corresponding manual segmentation result. $G(x,\cdot)$ is the CNN model with parameters x. $\psi(\cdot,\cdot)$ is some given loss function to judge the error between the output of give ξ_2 on $G(x,\cdot)$ and its corresponding manual annotation ξ_1. Although CNN models in supervisied learning have been achieved outstanding performance in the domain of image segmentation, the over-requirement of high quality data has become the bottleneck of the development of CNN models. This shortage is prominent in the medical domain because the medical images data always expresses as characters like small size of the samples, difficulty of manual annotation and strict requirement of the segmentation accuracy.

One of the ideas to improve the performance of CNN on medical image segmentation domain is to apply a mix model with CNN and a GAN. The mix model has the following form

$$\begin{aligned}\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} \ \hat{f}_N(x,y) :=& \frac{1}{N}\sum_{i=1}^{N}\lambda\cdot\psi\big(\xi_1^i, G(x,\xi_2^i)\big)+ \\ & \Big(\frac{1}{N}\sum_{i=1}^{N}\Big(\log(D(y,\xi_1^i)) + \log(1-D(y,G(x,\xi_2^i)))\Big),\end{aligned} \tag{4.1.2}$$

where $\mathcal{X}$, $\mathcal{Y}$ are two bounded boxes, $\{(\xi_1^i,\xi_2^i)\}_{i=1}^N$ is the finite collected data, ξ_2^i is the original data while the ξ_1^i is the corresponding label. The model can be regarded as a combination of a classical supervised learning problem and a generative adversarial problem with a trade-off parameter $\lambda \in [0,\infty)$. When $\lambda = 0$, problem (4.1.2) reduces to a classical supervised learning problem (4.1.1). When $\lambda \to \infty$, problem

(4.1.2) tends to become a vanilla GAN. The mixed model has been widely used in the medical domain, and most studies have shown that it can improve segmentation results compared to traditional supervision models[67, 68].

In the remainder of this chapter, QNSTR is applied to find a solution of a nonconvex-nonconcave minimax problem in which to use the mix model in (4.1.2) to implement medical eye images segmentation on three different data sets with two types of data. To achieve better segmentation results, a series of common data preprocessing methods are applied in all the experiments. The details of the process are as follows: (1) We converted raw RGB images into single channel images because single channel images show the better vessel background contrast than RGB images [69]; (2) We applied normalization and Contrast Limited Adaptive Histogram Equalization (CLAHE) [70] to the entire dataset to enhance the foreground-background contrast. (3) We introduced gamma correction to further improve the image quality.

All the experiments in this section apply a Unet structure [71] of segmentation model $G(x, \cdot)$ which includes 18 layers with $m_1 = 121435$ parameters, and a deep Convolutional Neural Network (CNN) structure of discrimination model $D(y, \cdot)$ which contains 5 convolutional layers and 1 fully connected layer with $m_2 = 142625$ parameters. The feasible sets $\mathcal{X}$ and $\mathcal{Y}$ are set as $[-5, 5]^{m_1}$ and $[-5, 5]^{m_2}$, respectively. We employ the GELU activation function for each hidden layer in both D and G, and the Sigmoid activation function for the output layer of both D and G. We compare our results based on problem (4.1.2) with some existing models. In our experiment, we use $\lambda = 100$ and $V_k^{z,H}$ with $L = 5$. Next, we introduce 2 partical problems in medical domains and use the mix GAN model in (4.1.2) for solving these problems.

4.2 Blood vessel segmentation of Digital Retinal Images (DRI)

The fundoscopic exam is a crucial procedure to provide information to diagnose different retinal degenerative diseases such as Diabetic Retinopathy, Macular Edema and Cytomegalovirus Retinitis. A highly accurate system to sketch out the blood vessel and find abnormalities on fundoscopic images is necessary. Although the supervisied learning models with framework of network such as Unet are able to segment macrovessels accurately, they failed for segmenting microvessels with high certainty. We apply two different open source Digital Retinal Images (DRI) data, DRIVE (https://drive.grand-challenge.org/) and CHASE_BD1 (https://researchdata.kingston.ac.uk/96/).

We compare the segmentation results by the traditional metrics such as F1-score, Sensitivity, Specificity and Accuracy. The form of these metrics are given as follows:

$$\text{Sensivity} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\text{GT}_i \cap \text{SR}_i|}{|\text{GT}_i \cap \text{SR}_i| + |\text{GT}_i \cap \text{SR}_i^c|}, \tag{4.2.1}$$

$$\text{Specificity} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\text{GT}_i^c \cap \text{SR}_i^c|}{|\text{GT}_i^c \cap \text{SR}_i^c| + |\text{GT}_i^c \cap \text{SR}_i|}, \tag{4.2.2}$$

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\text{GT}_i \cap \text{SR}_i| + |\text{GT}_i^c \cap \text{SR}_i^c|}{\Omega}, \tag{4.2.3}$$

$$\text{Precision} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\text{GT}_i \cap \text{SR}_i|}{|\text{GT}_i \cap \text{SR}_i| + |\text{GT}_i^c \cap \text{SR}_i|}, \tag{4.2.4}$$

$$\text{F1} = \frac{2\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}}, \tag{4.2.5}$$

where Ω is the universal set of all pixel indices in an image. GT_i represents the ground truth vessel indices for the i-th image, and SR_i represents the indices of

pixels labeled as vessels in the segmentation result of the i-th image. Furthermore, we apply the Area Under Curve-Receiver Operating Characteristic (AUC-ROC) [72] and Structural Similarity Index Measure (SSIM) [73] to judge the segmentation quality from mixed model (4.1.2) trained by QNSTR.

4.2.1 Segmentation results of DRIVE data

We utilized the DRIVE dataset, which comprises 20 retinal images with manually annotated segmentation labels. We allocated 16 images for training and reserved 4 images for testing purpose.

The segmentation results of the test data obtained using the QNSTR algorithm for model (4.1.2) are presented in Figure 4.1. The top two rows of subfigures in Figure 4.1 display the original fundus images and their corresponding manual annotations, respectively. The third row of subfigures show the segmentation results produced by model (4.1.2) using the QNSTR algorithm. The fourth and fifth rows of subfigures illustrate the error maps, highlighting the discrepancies between the segmentation results of model (4.1.2) and the manual annotations. The results presented in Figure 4.1 provide an intuitive visualization of the segmentation performance achieved by the QNSTR algorithm. To further demonstrate its effectiveness, we will present a comparative analysis with other methods, highlighting the superior performance of the QNSTR.

We report the comparison results of alternating Adam and QNSTR for model (4.1.2) in Figure 4.2. The first column of subfigures display the manual segmentation results, serving as the ground truth. The second and fourth columns of subfigures show the segmentation results obtained by model (4.1.2) using alternating Adam and QNSTR, respectively. The third and fifth columns of subfigures illustrate the error maps, highlighting the discrepancies between the segmentation results produced by Adam, QNSTR, and the ground truth in the first column, respectively. Through the

Molds	F1 score	Sensitivity	Specificity	Accuracy	AUC-ROC	SSIM
Residual Unet[3]	0.8149	0.7726	0.9820	0.9553	0.9779	-
RecurrentUnet[3]	0.8155	0.7751	0.9816	0.9556	0.9782	-
R2Unet[3]	0.8171	0.7792	0.9813	0.9556	0.9784	-
DFUNet[4]	0.8190	0.7863	0.9805	0.9558	0.9779	0.8789
IterNet[5]	**0.8205**	0.7735	**0.9838**	0.9573	**0.9816**	0.9008
Alt Adam	0.7956	0.7830	0.9807	0.9591	0.9751	0.8912
QNSTR	0.8028	**0.8363**	0.9804	**0.9654**	0.9791	**0.9015**

Table 4.1: Performance of the QNSTR algorithm and alternating Adam for model (4.1.2), and other methods in [3, 4, 5] for model (4.1.1) on DRIVE

intuitive visualization provided by Figure 4.2, it is evident that QNSTR outperforms Adam. To further substantiate this observation, we present a comprehensive comparison of QNSTR and Adam, as well as other state-of-the-art methods [3, 4, 5], in terms of sensitivity (4.2.1), specificity (4.2.2), accuracy (4.2.3), precision (4.2.4), and F1-score (4.2.5) in Table 4.1. Table 4.1 shows that the QNSTR algorithm outperforms Adam across all evaluation metrics. Compared to other methods, it achieves superior performance in Specificity, Accuracy, and SSIM, indicating that it is comparable to some state-of-the-art methods. Overall, we can conclude that QNSTR is a promising approach for solving problem (4.1.2) in the context of blood vessel segmentation in the DRIVE dataset.

4.2.2 Segmentation results of CHASE_DB1 data

We download the CHASE_BD1 includes 28 retinal images with manual segmentation label. We applied 20 of the 28 images as training data. Similarly, we report in Figure 4.3, Figure 4.4 to visually display the segmentation results of the QNSTR on model (4.1.2) and its comparison with Adam. We also report the comparison of segmentation results by the QNSTR, alternating Adam and the state-of-art results in [3, 5]. Table 4.2 shows that the QNSTR algorithm is a promising approach for solving problem (4.1.2) in the context of blood vessel segmentation in the CHASE dataset. Next, we report the segmentation results of the mix GAN model in (4.1.2)

Molds	F1 score	Sensitivity	Specificity	Accuracy	AUC-ROC	SSIM
DenseBlock-Unet[3]	0.8006	**0.8178**	0.9775	0.9631	0.9826	0.8867
DFUNet[3]	0.8001	0.7859	0.9822	0.9644	0.9834	0.9175
IterNet[5]	**0.8073**	0.7970	0.9823	0.9655	**0.9851**	0.9123
Alt Adam	0.7762	0.7957	0.9788	0.9652	0.9715	0.9096
QNSTR	0.7831	0.7937	**0.9879**	**0.9695**	0.9838	**0.9182**

Table 4.2: Performance of the QNSTR algorithm and alternating Adam for model (4.1.2), and other methods in [3, 4, 5] for model (4.1.1) on CHASE_DB1.

for another different partical problem.

4.3 Eye image segmentation for optic disc

The optic nerve disc, commonly referred to as the optic disc, is typically the brightest area in a normal color fundus camera image. As a vital tissue of the eye, the optic disc plays a crucial role in diagnosing various eye diseases. For instance, in the case of glaucoma, doctors assess the cup-disc diameter ratio (CDR) to determine the likelihood of the disease. A normal CDR is relatively small, whereas a ratio exceeding 0.7 often indicates a high risk of glaucoma. Furthermore, since Age-related Macular Degeneration (AMD) typically occurs near the macula, many macular localization and segmentation algorithms rely on identifying the optic disc's position first. By leveraging prior knowledge that the distance between the optic disc and macula is approximately twice the width of the optic disc, these algorithms can effectively localize the macula region. Additionally, in coronary retinal image analysis, optic disc localization is a precursor to measuring retinal vessel caliber. Therefore, accurate localization and segmentation of the optic disc are essential components of intelligent diagnostic algorithms for fundus diseases.

RIM-ONE(https://www.kaggle.com/datasets/lucascunhadecarvalho/rimoner2), which comprises 455 retinal images with manually annotated segmentation labels, is implemented on optic segmentation problem. We allocated 440 images for training

and reserved 15 images for testing purposes.

The segmentation results of the test data obtained using the QNSTR algorithm for model (4.1.2) are presented in Figure 4.5. The top of two rows of subfigures in Figure 4.5 display the original fundus images and their corresponding manual annotations, respectively. The third row of subfigures show the segmentation results produced by model (4.1.2) using the QNSTR algorithm. The fourth and fifth rows of subfigures illustrate the error maps, highlighting the discrepancies between the segmentation results of model (4.1.2) and the manual annotations.

We report the comparison results of alternating Adam and QNSTR for model (4.1.2) in Figures 4.6 and 4.7. The first column of subfigures display the manual segmentation results, serving as the ground truth. The second and fourth columns of subfigures show the segmentation results obtained by model (4.1.2) using alternating Adam and QNSTR, respectively. The third and fifth columns of subfigures illustrate the error maps, highlighting the discrepancies between the segmentation results produced by Adam and QNSTR, respectively, and the ground truth in the first column. Figure 4.7 visually displays the results of manual annotations, segmentation results of model trained by Adam and segmentation results of model trained by QNSTR. Thoughout the comparison, we can easily conclude that QNSTR is outperformed compared with Adam.

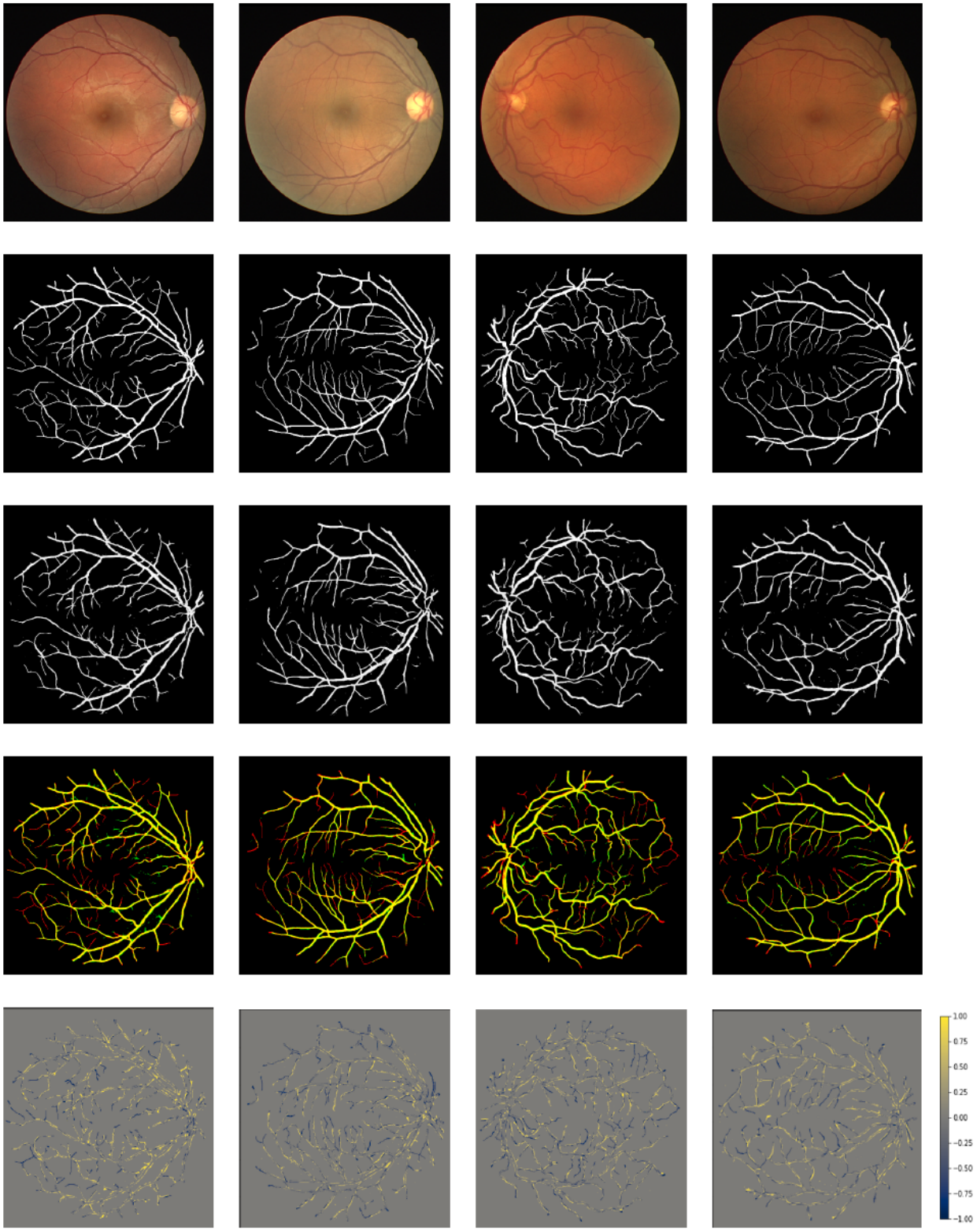


Figure 4.1: Results on DRIVE: Row 1. fundus image, Row 2. manual segmentation, Row 3. vessel map generated by GANs with the QNSTR algorithm, Row 4. yellow (correct); red (wrong); green (missing), Row 5. error.

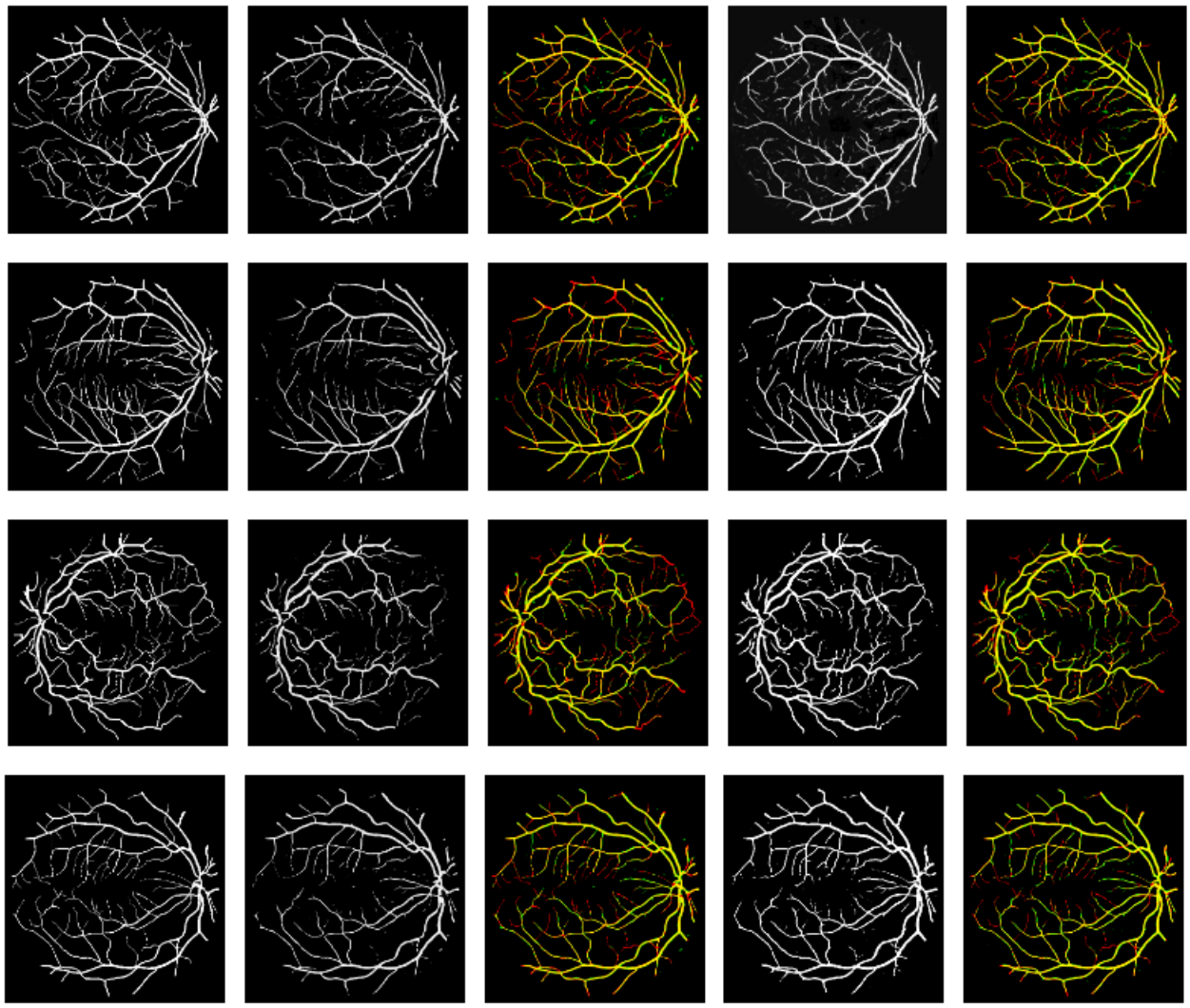


Figure 4.2: Comparison of alternating Adam and the QNSTR algorithm on DRIVE. Columns from left to right are: 1. manual segmentation, 2. vessel map generated by GANs with alternating Adam, 3. yellow (correct); red (wrong); green (missing) of alternating Adam, 4. vessel map generated by GANs with QNSTR algorithm, 5. yellow (correct); red (wrong); green (missing) of the QNSTR algorithm.

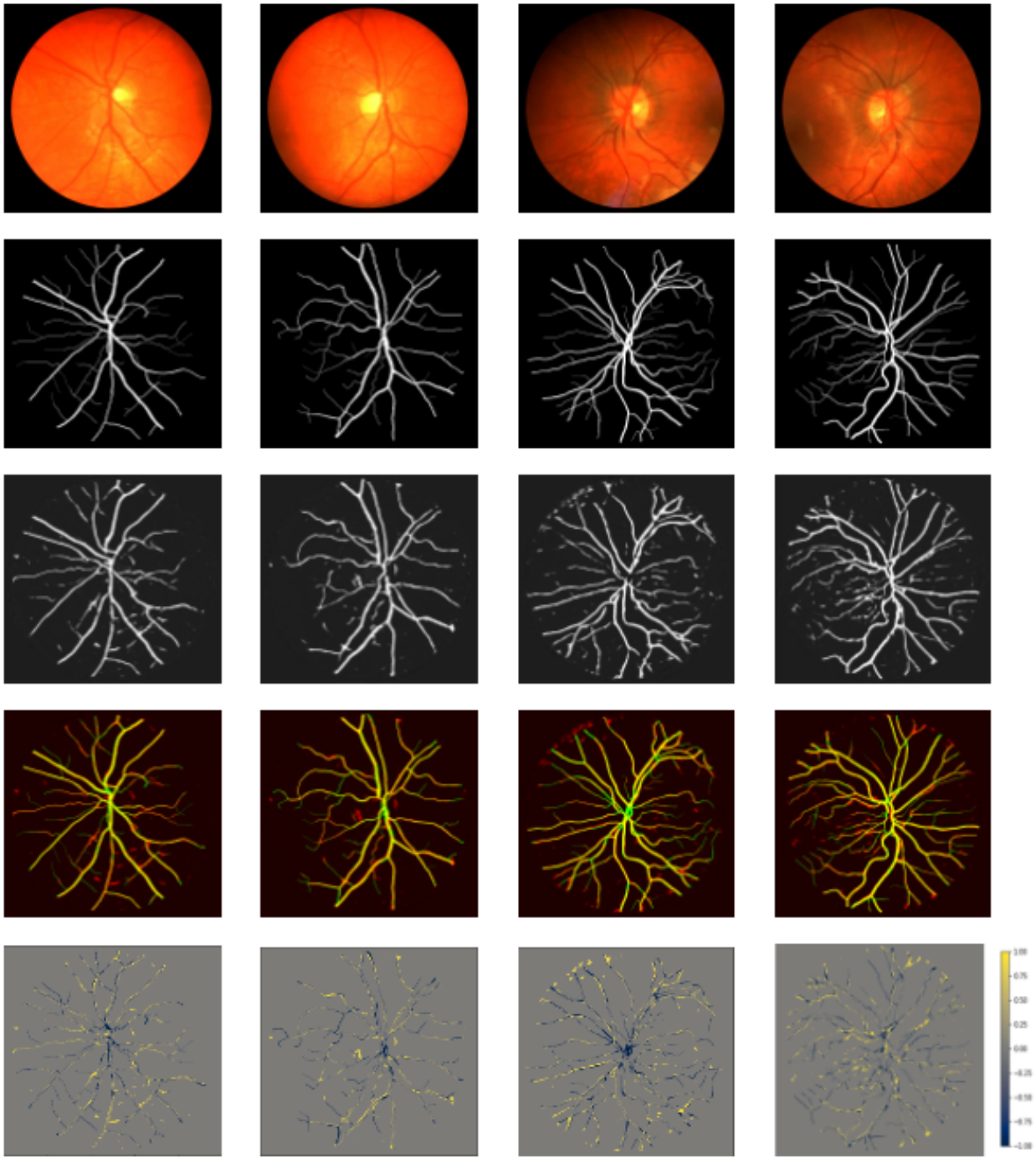


Figure 4.3: Results on CHASE_DB1: Row 1. fundus image, Row 2. manual segmentation, Row 3. vessel map generated by GANs with QNSTR algorithm, Row 4. yellow (correct); red (wrong); green (missing), Row 5. error.

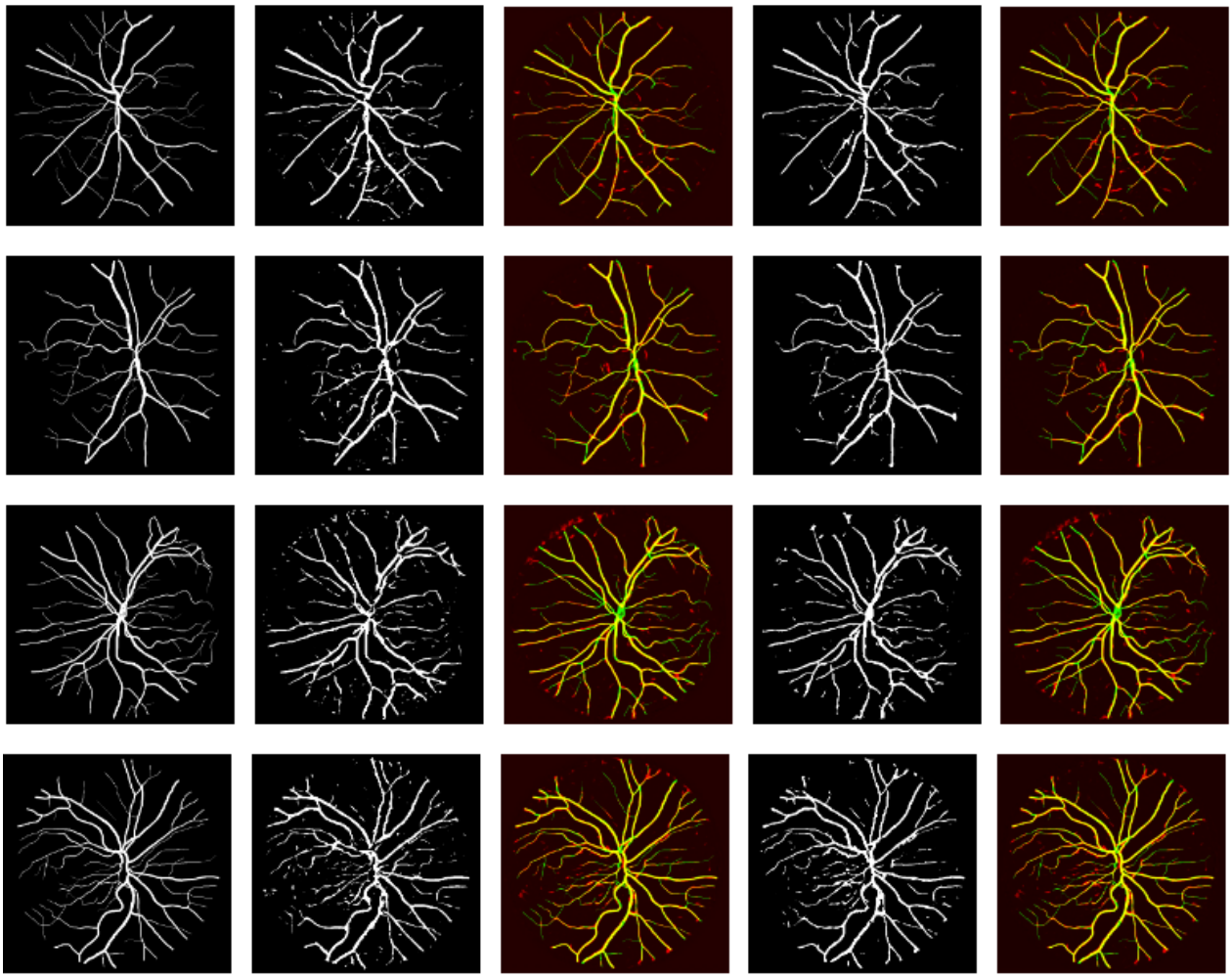


Figure 4.4: Comparison of alternating Adam and the QNSTR algorithm on CHASE_DB1. Columns from left to right are: 1. manual segmentation, 2. vessel map generated by GANs with alternating Adam, 3. yellow (correct); red (wrong); green (missing) of alternating Adam, 4. vessel map generated by GANs with QNSTR algorithm, 5. yellow (correct); red (wrong); green (missing) of the QNSTR algorithm.

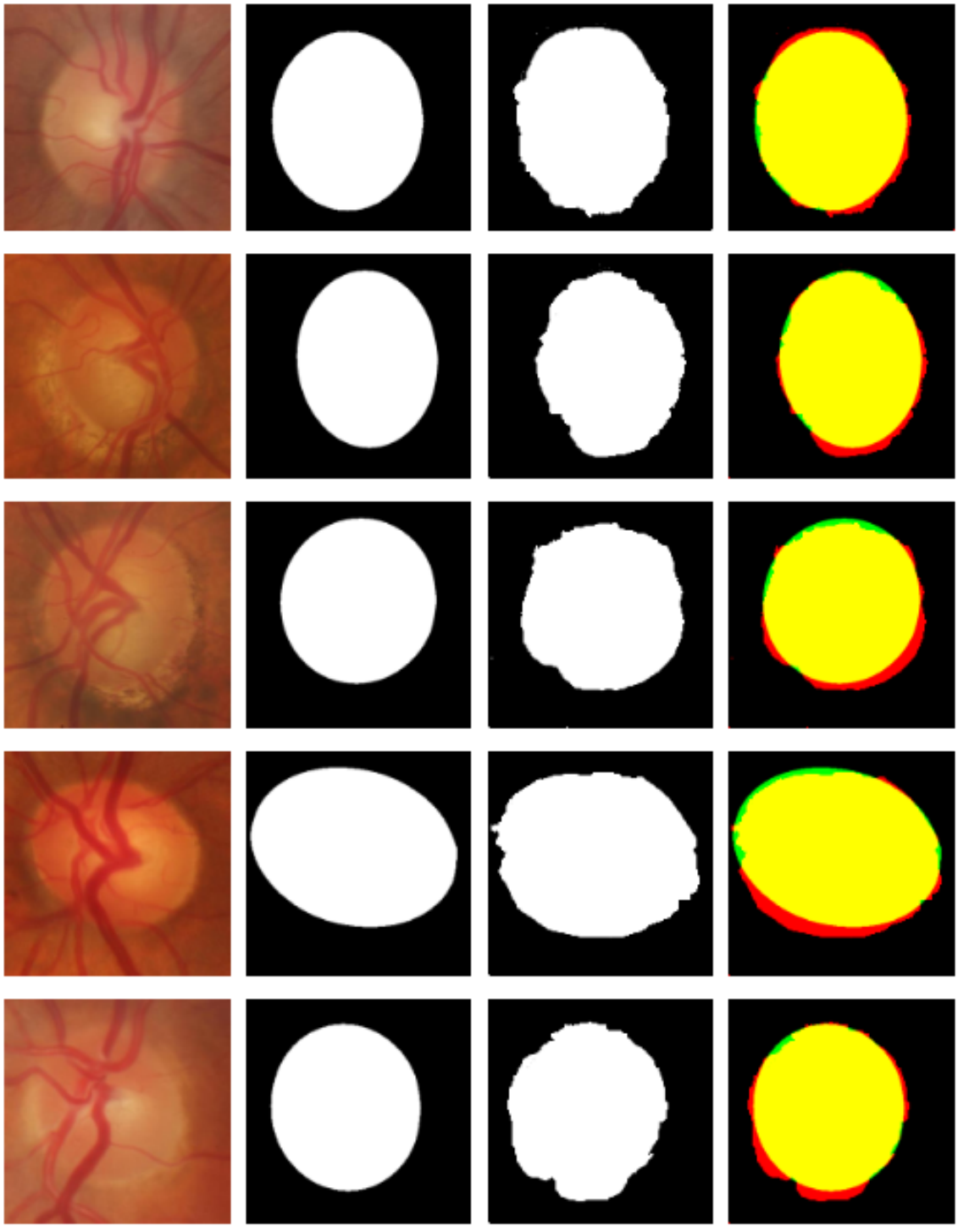


Figure 4.5: Results on RIM-ONE: Column 1. fundus image, Column 2. manual segmentation, Column 3. optic dist segmented by GANs with QNSTR algorithm, Column 4. yellow (correct); red (wrong); green (missing), Column 5. Error.

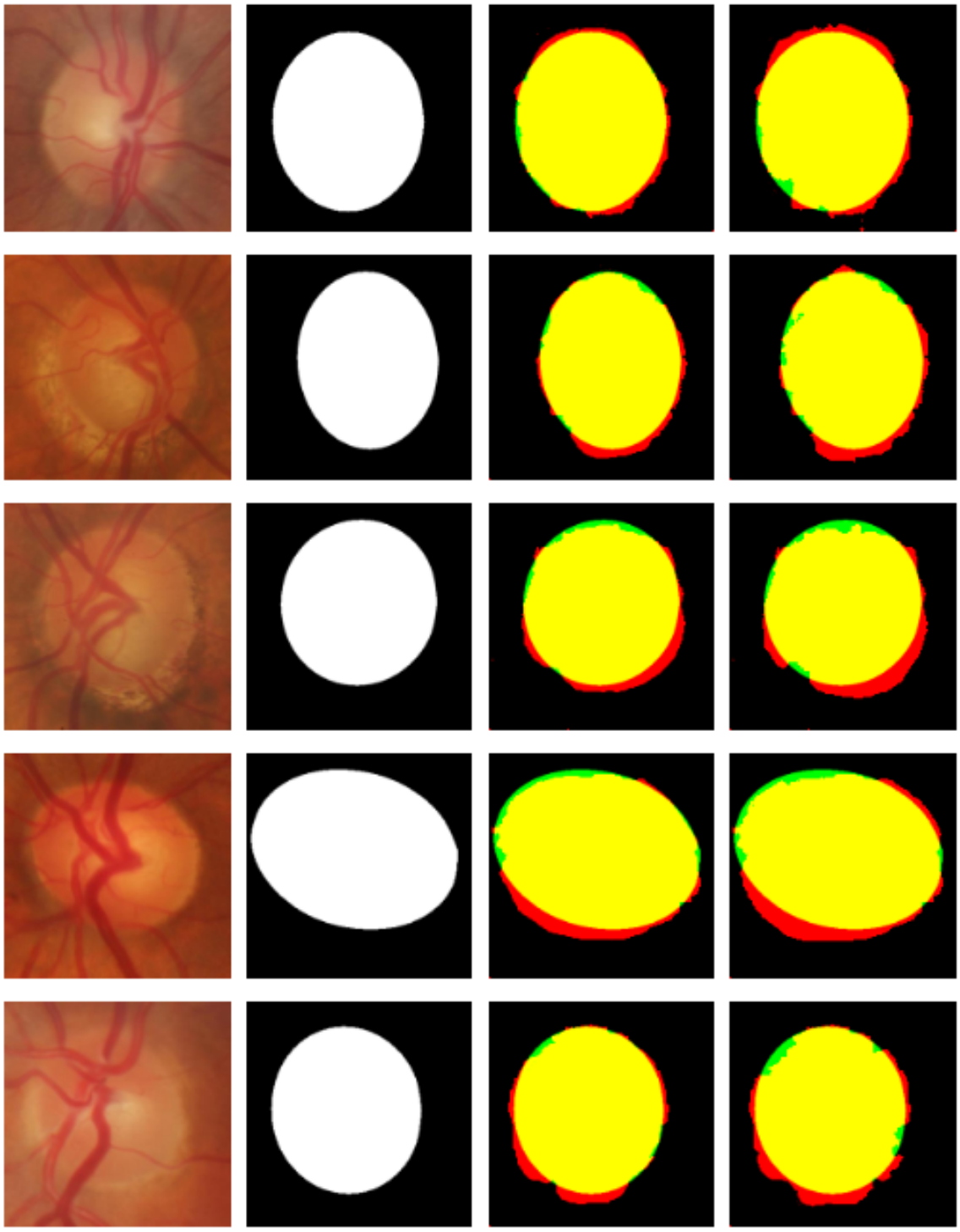


Figure 4.6: Comparison of alternating Adam and QNSTR algorithm on RIM_ONE_R2. Columns from left to right are: 1. original images, 2. manual segmentation, 3. yellow (correct); red (wrong); green (missing) of QNSTR, 4. yellow (correct); red (wrong); green (missing) of alternating Adam.

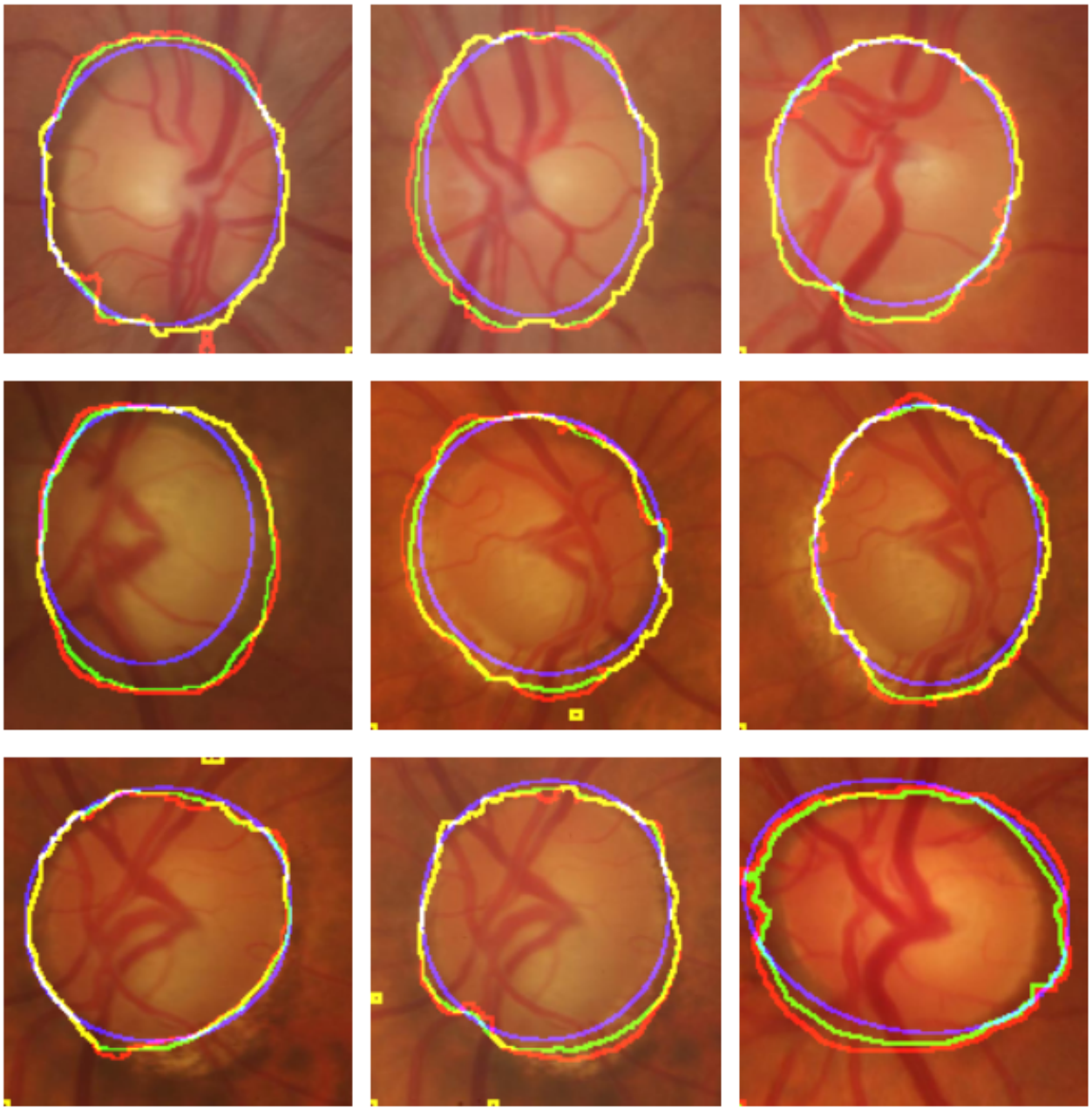


Figure 4.7: The segmentation results along with the ground truth. blue (ground truth); red (results of alternating Adam); green (results of QNSTR algorithm).

Chapter 5

Conclusions and future work

In this chapter, we conclude this thesis and give some remarks for further research.

5.1 Conclusions

In this thesis, we mainly study a class of nonmonotone VIs with box constraints, and investigate the following two research problems in detail.

1. We consider the residual function of the VI($\mathcal{Z}, H$) with box constraints in (1.1.4), which can be formulate as a nonsmooth least squares problem. We propose a QNSTR algorithm for solving VI($\mathcal{Z}, H$) via the merit function. In order to deal with the nonsmoothness of problem (1.2.7), we introduce a smoothing relaxation problem (2.1.10) and establish the relationship between problem (1.2.7) and problem (2.1.10). By exploiting the special structure of problem (2.1.10), we propose the QNSTR algorithm (Algorithm (1)). In the algorithm, we solve a low dimensional strongly convex with ellipse constraint subproblem in each step and we demonstrate that all computational processes involved in the QNSTR algorithm can be efficiently computed using simple techniques. We prove that there exist an accumulation point of $\{z_k\}_{k=0}^{\infty}$ generated by the QNSTR is a Clarke stationary point of (1.2.7). Moreover, if each elements in the Clarke generalized Jacobian at this accumulation point is

nonsingular, it is a solution of (1.1.1). We also implement inexact-QNSTR (Algorithm 2) with fixed smoothing parameter and show that any accumulation point of the sequence generated by QNSTR with fixed smoothing parameter is an ϵ-first order stationary point of (2.1.10). If the Jacobian matrix at the accumulation point is nonsingular, it is an ϵ solution of (1.1.1). Additionally, we establish that the complexity of QNSTR with fixed smoothing parameters is $O(\ln(\epsilon^{-2})\epsilon^{-3})$.

2. We apply the QNSTR algorithm to solve a class of minimax problems with box constraints (3.1.1). We consider a discrete form (3.1.2) of the minimax problem (3.1.1) by using SAA and establish a relationship between optimal values, global minimax point, first-order stationary points and second-order stationary points of the problem (3.1.1) and (3.1.2) in Theorem 3.1. We apply the QNSTR algorithm to find a first-order stationary point of (3.1.2). Finally, we present some numerical results in image segmentation to show the effectiveness of the QNSTR algorithm. The results show that the QNSTR algorithm for solving problem (4.1.2) exhibits superior performance in segmentation, making it a more promising approach for these applications.

5.2 Future work

Related topics for the future research work are listed below.

1. In this thesis, we introduce the QNSTR algorithm and demonstrate that its computational process can be simplified using various techniques. Furthermore, we show that the QNSTR is well-suited for problems with high-dimensional parameters. However, a notable limitation of the proposed QNSTR algorithm is that it requires the use of all training data at each step, which can be computationally expensive and memory-intensive when dealing with large datasets.

Many practical problems in machine learning and artificial intelligence involve vast amounts of data, a promising direction for future research is the development of a stochastic version of QNSTR that leverages mini-batch data at each iteration, thereby addressing the scalability challenges associated with large datasets.

2. The selection of a searching subspace V_k is a crucial aspect of our algorithm, and the linear independence among the column vectors is a necessary condition. However, the searching subspace used in the numerical part of this work does not guarantee that V_k will always be of full column rank. Developing a strategy to ensure the linear independence of the column vectors in V_k would be a meaningful contribution to the QNSTR algorithm. Furthermore, our numerical results suggest that the choice of searching subspace can significantly impact the performance of the algorithm. Therefore, designing a strategy to guide the selection of the searching subspace V_k is an interesting and important problem that warrants further investigation.

Bibliography

[1] A. Antoniadis. Wavelets in statistics: a review. *Journal of the Italian Statistical Society*, 6:97–130, 1997.

[2] C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 894–942, 2010.

[3] M.Z. Alom, M. Hasan, C. Yakopcic, T.M. Taha, and V.K. Asari. Recurrent residual convolutional neural network based on U-net (R2U-Net) for medical image segmentation. *arXiv preprint arXiv:1802.06955*, 2018.

[4] Q. Jin, Z. Meng, T.D. Pham, Q. Chen, L. Wei, and R. Su. DUNet: A deformable network for retinal vessel segmentation. *Knowledge-Based Systems*, 178:149–162, 2019.

[5] L. Li, M. Verma, Y. Nakashima, H. Nagahara, and R. Kawasaki. Iternet: Retinal image segmentation utilizing structural redundancy in vessel networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3656–3665, 2020.

[6] R.T. Rockafellar and R. J.-B. Wets. *Variational Analysis*, volume 317. Springer Science & Business Media, New York, 2009.

[7] F. Facchinei and J.-S. Pang. *Finite-dimensional Variational Inequalities and Complementarity Problems*. Springer, New York, 2003.

[8] M. Sibony. Méthodes itératives pour les équations et inéquations aux dérivées partielles non linéaires de type monotone. *Calcolo*, 7:65–183, 1970.

[9] B. Martinet. Brève communication. régularisation d'inéquations variationnelles par approximations successives. *Revue française d'informatique et de recherche opérationnelle. Série rouge*, 4: 154–158, 1970.

[10] G. M. Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

[11] P. Tseng. On linear convergence of iterative methods for the variational inequality problem. *Journal of Computational and Applied Mathematics*, 60(1-2):237–252, 1995.

[12] S. J Wright. Numerical Optimization, 2006.

[13] P. Tseng. A modified forward-backward splitting method for maximal monotone mappings. *SIAM Journal on Control and Optimization*, 38:431–446, 2000.

[14] A. Nemirovski. Prox-method with rate of convergence O (1/t) for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15:229–251, 2004.

[15] Y. Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109:319–344, 2007.

[16] Y. Nesterov and L. Scrimali. Solving strongly monotone variational and quasi-variational inequalities. *Discrete and Continuous Dynamical Systems*, 31: 1383–1396, 2011.

[17] R.DC. Monteiro and B.F. Svaiter. On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean. *SIAM Journal on Optimization*, 20: 2755–2787, 2010.

[18] A. Mokhtari, A. Ozdaglar, and S. Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: proximal point approach. In *International Conference on Artificial Intelligence and Statistics*, 1497–1507, 2020.

[19] A. Mokhtari, A.E. Ozdaglar, and S. Pattathil. Convergence rate of $o(1/\mathrm{k})$ for optimistic gradient and extragradient methods in smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 30:3230–3251, 2020.

[20] K. Huang and S. Zhang. A unifying framework of accelerated first-order approach to strongly monotone variational inequalities. *arXiv preprint arXiv:2103.15270*, 2021.

[21] D. Adil, B. Bullins, A. Jambulapati, and S. Sachdeva. Optimal methods for higher-order smooth monotone variational inequalities. *arXiv preprint arXiv:2205.06167*, 2022.

[22] R. DC. Monteiro and B.F. Svaiter. Iteration-complexity of a newton proximal extragradient method for monotone variational inequalities and inclusion problems. *SIAM Journal on Optimization*, 22:914–935, 2012.

[23] Y. Nesterov. Cubic regularization of Newton's method for convex problems with constraints. 2006.

[24] W. Bian and X. Chen. Nonsmooth convex–concave saddle point problems with cardinality penalties. *Mathematical Programming*, 1–47, 2024.

[25] Z. Xu, H. Zhang, Y. Xu, and G. Lan. A unified single-loop alternating gradient projection algorithm for nonconvex–concave and convex–nonconcave minimax problems. *Mathematical Programming*, 201(1):635–706, 2023.

[26] T. Lin and M. I. Jordan. Perseus: A simple and optimal high-order method for variational inequalities. *Mathematical Programming*, 1–42, 2024.

[27] G. Cai, Q.-L. Dong, and Y. Peng. Strong convergence theorems for solving variational inequality problems with pseudo-monotone and non-lipschitz operators. *Journal of Optimization Theory and Applications*, 188:447–472, 2021.

[28] M. Lei and Y. He. An extragradient method for solving variational inequalities without monotonicity. *Journal of Optimization Theory and Applications*, 188: 432–446, 2021.

[29] D. V. Thong and A. Gibali. Extragradient methods for solving non-Lipschitzian pseudo-monotone variational inequalities. *Journal of Fixed Point Theory and Applications*, 21: 1–19, 2019.

[30] D. V. Thong, Y. Shehu, and O. S. Iyiola. Weak and strong convergence theorems for solving pseudo-monotone variational inequalities with non-Lipschitz mappings. *Numerical Algorithms*, 84: 795–823, 2020.

[31] W. Y. Wang and B. B.Ma. Extragradient type projection algorithm for solving quasimonotone variational inequalities. *Optimization*, 73: 2639–2655, 2024.

[32] M. Ye and Y. He. A double projection method for solving variational inequalities without monotonicity. *Computational Optimization and Applications*, 60: 141–150, 2015.

[33] T. N. Hai. Two modified extragradient algorithms for solving variational inequalities. *Journal of Global Optimization*, 78: 91–106, 2020.

[34] X. Chen. Superlinear convergence of smoothing quasi-Newton methods for nonsmooth equations. *Journal of Computational and Applied Mathematics*, 80: 105–126, 1997.

[35] X. Chen, L. Qi, and D. Sun. Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities. *Mathematics of Computation*, 67: 519–540, 1998.

[36] J. Martínez and L. Qi. Inexact Newton methods for solving nonsmooth equations. *Journal of Computational and Applied Mathematics*, 60: 127–145, 1995.

[37] Y.-X. Yuan. *Trust Region Algorithms for Nonlinear Equations*. Citeseer, Princeton, 1994.

[38] S. A. Gabriel and J. J. Moré. Smoothing of mixed complementarity problems. *Complementarity and Variational Problems: State of the Art*, 92: 105–116, 1997.

[39] X. Chen. Smoothing methods for nonsmooth, nonconvex minimization. *Mathematical Programming*, 134: 71–99, 2012.

[40] F. H. Clarke. *Optimization and Nonsmooth Analysis*. SIAM, Philadelphia, 1990.

[41] W. Zhou and X. Chen. Global convergence of a new hybrid Gauss–Newton structured BFGS method for nonlinear least squares problems. *SIAM Journal on Optimization*, 20: 2422–2441, 2010.

[42] R. B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, Cambridge, 1991.

[43] P. M. Esfahani and D. Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and tractable reformulations. *Mathematical Programming*, 171: 115–166, 2018.

[44] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[45] I. Goodfellow, J. P. Abadie, M. Mirza, B. Xu, D. W. Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in Neural Informtion Processing Systems*, 27, 2014.

[46] B. Dai, A. Shaw, L. Li, L. Xiao, N. He, Z. Liu, J. Chen, and L. Song. SBEED: Convergent reinforcement learning with nonlinear function approximation. In *Proceedings of the International Conference on Machine Learning*, 1125–1134, 2018.

[47] M. Rabbat and R. Nowak. Distributed optimization in sensor networks. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, 20–27, 2004.

[48] C. Jin, P. Netrapalli, and M. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *Proceedings of International Conference on Machine Learning*, 4880–4889, 2020.

[49] Y.-H. Dai and L. Zhang. Optimality conditions for constrained minimax optimization. *CSIAM Transactions on Applied Mathematics*, 1: 296–315, 2020.

[50] J. Jiang and X. Chen. Optimality conditions for nonsmooth nonconvex-nonconcave min-max problems and generative adversarial networks. *SIAM Journal on Mathematics of Data Science*, 5: 693–722, 2023.

[51] R. DC. Monteiro and Benar Fux Svaiter. Complexity of variants of Tseng's modified FB splitting and Korpelevich's methods for hemivariational inequalities with applications to saddle-point and convex optimization problems. *SIAM Journal on Optimization*, 21: 1688–1720, 2011.

[52] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. *submitted to SIAM Journal on Optimization*, 2, 2008.

[53] H. Rafique, M. Liu, Q. Lin, and T. Yang. Weakly-convex–concave min–max optimization: provable algorithms and applications in machine learning. *Optimization Method and Software*, 37: 1087–1121, 2022.

[54] T. Lin, C. Jin, and M. I. Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *Proceedings of the International Conference on Machine Learning*, 6083–6093, 2020.

[55] T. Lin, C. Jin, and M. I. Jordan. Near-optimal algorithms for minimax optimization. In *Proceedings of the Conference on Learning Theory*, 2738–2779. PMLR, 2020.

[56] J. Diakonikolas, C. Daskalakis, and M. I. Jordan. Efficient methods for structured nonconvex-nonconcave min-max optimization. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2746–2754, 2021.

[57] J. Yang, N. Kiyavash, and N. He. Global convergence and variance-reduced optimization for a class of nonconvex-nonconcave minimax problems. *arXiv preprint arXiv:2002.09621*, 2020.

[58] A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures on Stochastic Programming: Modeling and Theory.* SIAM, Philadelphia, 2021.

[59] A. Shapiro. Monte Carlo sampling methods. *In: Ruszczyński, A., Shapiro, A. (eds.) Handbooks in Operations Research and Management Science,*, 353–425, Elsevier, Amsterdam, 2003.

[60] H. Xu. Uniform exponential convergence of sample average random functions under general sampling with applications in stochastic programming. *Journal of Mathematical Analysis and Applications*, 368: 692–710, 2010.

[61] D. Hendrycks and K. Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[62] G. Zhang, Y. Wang, L. Lessard, and R. B. Grosse. Near-optimal local convergence of alternating gradient descent-ascent for minimax optimization. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 7659–7679, 2022.

[63] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng. Training gans with optimism. *arXiv preprint arXiv:1711.00141*, 2017.

[64] M. Liu, H. Rafique, Q. Lin, and T. Yang. First-order convergence theory for weakly-convex-weakly-concave min-max problems. *Journal of Machine Learning Research*, 22: 7651–7684, 2021.

[65] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer Nature, Switzerland, 2022.

[66] D. A. Forsyth and Jean Ponce. *Computer Vision: a Modern Approach*. Prentice Hall Professional Technical Reference, Upper Saddle River, 2002.

[67] Y. Zhang, L. Yang, J. Chen, M. Fredericksen, D. P. Hughes, and D. Z. Chen. Deep adversarial networks for biomedical image segmentation utilizing unannotated images. In *Proceedings of the Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference*, 408–416. Springer, 2017.

[68] J. Son, S. J. Park, and K.-H. Jung. Towards accurate segmentation of retinal vessels and the optic disc in fundoscopic images with generative adversarial networks. *Journal of Digital Imaging*, 32: 499–512, 2019.

[69] J.-VB. Soares, J. JG. Leandro, R. M. Cesar, H. F. Jelinek, and M. J. Cree. Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification. *IEEE Transactions on medical Imaging*, 25:1214–1222, 2006.

[70] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. Haar. Romeny, J. B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39:355–368, 1987.

[71] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference*, 234–241. Springer, 2015.

[72] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit.*, 30: 1145–1159, 1997.

[73] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13: 600–612, 2004.