Chapter 1

Introduction

Any mathematical representation associated with a real-life problems can be categorized as either an equation or an equality or an inclusion depending on the nature of representation. The present thesis is concentrated to solve inclusion problems. Monotone inclusion problems have attracted the attention of mathematics researchers over the globe because of their importance in solving optimization problems for the last three decades. In fact, these are highly used in engineering problems, economic problems and problems arising in different branches of basic sciences. Consider a d-dimentional vector $\mathbf{x} = (x_1, \dots, x_d)$ and functions $f_0, \dots, f_m$ defined on nonempty convex set $S \subseteq \mathbb{R}^d$ where f_i is a finite convex function, $i = 1 \dots r$ and f_i is an affine function on S, $i = r + 1, \dots, m$. Then the general form of the minimization problem can be defined as follows:

$$\begin{cases} \min_{x\in S} f_0(x), \\ \text{subject to } f_1(x) \le 0, \dots, f_r(x) \le 0, \\ f_{r+1}(x) = 0, \dots, f_m(x) = 0, \end{cases} \tag{1.1}$$

where S is the basic feasible set. The optimization problems can be categorized into following:

- Constrained problems: $S \subset \mathbb{R}^d$.
- Unconstrained problem: $S \equiv \mathbb{R}^d$.
- Smooth problems: all f_i are differentiable, i=1,$\dots, m$.
- Nonsmooth problems: there is a nondifferentiable function component $f_j(x)$, $j \in \{1, \dots, m\}$.

Definition 1.0.1. ***(Karush-Kuhn-Tucker (KKT) Optimality Conditions)*** *A point $\bar{x}$ be the solution of the minimization problem if there exist $\lambda_1, \dots, \lambda_m \ge 0$ and $\bar{x}$ satisfy the following conditions*

(i) $f_i(\bar{x}) \le 0$ *and* $\lambda_i f_i(\bar{x}) = 0, i = 1, \dots, r$.

(ii) $f_i(\bar{x}) = 0$ *for* $i = r+1, \dots, m$,

(iii) $0 \in \partial f_0(\bar{x}) + \lambda_1 \partial f_1(\bar{x}) + \cdots + \lambda_m \partial f_m(\bar{x})$.

In case of optimization problem (1.1) having no constraints, the abstract form of the above KKT optimality conditions can be seen as a problem to search a point x in the Hilbert space $\mathcal{H}$ such that $y \in T(x)$ for a given $y \in \mathcal{H}$ and an operator $T : \mathcal{H} \to 2^{\mathcal{H}}$. This is called the inclusion problem. In this thesis, the author has focussed to solve the inclusion problem of monotone operators for $y = 0$, then the inclusion problem can be written as follows:

Problem 1.0.1. *Find $x \in \mathcal{H}$ such that $0 \in \mathrm{T}(x)$.*

The evolution equation associated with Problem 1.0.1 is given as

$$\begin{cases} 0 \in \frac{\partial x}{\partial t} + T(x) \\ x(0) = x_0. \end{cases} \tag{1.2}$$

Suppose $T = \nabla f$, where $f : \mathbb{R}^d \to \mathbb{R}_\infty$ is a convex function and differentiable on $\mathbb{R}^d$. The simplest method to solve the Problem 1.0.1 is given by

$$x_{n+1} = (Id - \lambda_n \nabla f)(x_n), \tag{1.3}$$

where $\lambda_n > 0$ is the step-size parameter and the operator $(Id - \lambda_n T)$ is called the forward operator. This method is known as the gradient descent method. In case, f is nondifferentiable, algorithm (1.3) is generalized into the subgradient method, which is given by

$$x_{n+1} = (Id - \lambda_n \partial f)(x_n), \tag{1.4}$$

where ∂f is the subdifferential of f. Gradient and subgradient methods are considered as the methods of finding zeros of gradient or subgradient of a function f. The monotone inclusion problems deal with a general class of operators, the monotone operators.

1.1 Proximal Point Algorithm

Monotone operator was first defined by Kachurosvkii in [49]. It plays an important role in functional analysis. It has significant history, the initial contribution can be found in the works [22, 23, 24, 50, 67, 66, 68, 80]. Consider the Hilbert space $\mathcal{H}$ equipped with inner product $\langle\cdot,\cdot\rangle$ and induced norm $\|\cdot\|$.

Definition 1.1.1. *Consider a set-valued function $T:\mathcal{H}\to 2^{\mathcal{H}}$. T is said to be monotone if satisfies* $\forall x,y\in\mathcal{H}$

$$\langle x-y,u-v\rangle\geq 0 \textit{ whenever } u\in T(x),v\in T(y). \tag{1.5}$$

T is said to be maximally monotone if there does not exist any monotone operator $S:\mathcal{H}\to 2^{\mathcal{H}}$ such that graph of S contains the graph of T. In the last almost 40 years, the monotone operators have turned out to be an important tool in the study of various problems arising in the domain of optimization, nonlinear analysis, differential equations and other related fields. Among those operators, it seems that the class of maximal monotone ones contains the mappings that possess the most desirable properties, such as, for example, local boundedness, perturbation surjectivity in reflexive spaces, generic single-valuedness and continuity in appropriate classes of Banach spaces, and others. In case of monotone operators, Proximal Point Algorithm (PPA) [81] is proposed to solve the inclusion Problem 1.0.1, which is given by,

$$x_{n+1}=x_n-\lambda_n v_n, \text{ where } v_n\in T(x_{n+1}), \tag{1.6}$$

where λ_k is a sequence of positive regularization parameter. Equivalentally, it can be written as

$$x_{n+1}=J_{\lambda_n T}(x_n), \tag{1.7}$$

where $J_{\lambda T} = (Id + \lambda T)^{-1} : \mathcal{H} \to \mathcal{H}$ is the resolvent operator of T with parameter λ. The PPA may be viewed as one step time-discretization of the dynamical system

$$0 \in \dot{x}(t) + \lambda_n T(x(t)), \text{ a.e. } t > 0,$$

where λ_n is interpreted as a step-size parameter. Denote the solution set $T^{-1}\{0\}$ as $\mathcal{S}$. The sequence generated by PPA converges strongly to a point in the solution set $\mathcal{S}$ provided $\mathcal{S} \neq \emptyset$. PPA and its dual version in the context of convex programming, the method of multipliers of Hesteness and Powel, have been extensively studied ([13, 44, 53]) and are known to yield as special cases of decomposition methods such as a method of partial inverses([84, 85]), the Douglas-Rachford splitting method and alternating direction method of multipliers ([40, 41, 42]).

1.2 Splitting Methods

Since the resolvent of an operator is not present in any closed form, the resolvent of an operator is hard to find. This difficulty is reduced by splitting the operator as the sum of two operators and evaluating separately either via resolvent in case of the set-valued operator or the operator itself when it is single-valued. The operator T splits as the sum of A and B such that the resolvent of A, $(Id + A)^{-1}$ and B, $(Id + B)^{-1}$ is easy to compute. For $T = A + B$, the monotone inclusion Problem 1.0.1 can be written as

$$\text{Find } x \in \mathcal{H} \text{ such that } 0 \in (A + B)(x). \tag{1.8}$$

Using the idea of splitting the operator, the first two algorithms are respectively known as Paceman-Rachford and Douglas-Rachford algorithm, which are originally proposed for linear operators. The Paceman-Rachford algorithm [77] proposed by Paceman and Rachford for linear operators is given by

$$x_{n+1} = (Id + \lambda B)^{-1}(Id - \lambda A)(Id + \lambda A)^{-1}(Id - \lambda B)(x_n),$$

which is not unconditionally stable, but converges to the solution of the stationary problem for λ sufficiently small if B is Lipschitz continuous. The Douglas-Rachford algorithm [39] is given by

$$x_{n+1} = (Id + \lambda B)^{-1}[(Id + \lambda A)^{-1}(Id - \lambda A)^{-1}(Id - \lambda B) + \lambda B](x_n). \tag{1.9}$$

Lions and Mercier [60] have studied the Paceman-Rachford and Douglas-Rachford algorithms and improved the algorithms which are equivalent to the Paceman-Rachford and Douglas-Rachford algorithm upto the variable $x_n = (I + \lambda B)^{-1}(v_n)$. These improved algorithms show the convergences for set-valued monotone operators also. Eckstein [40] have rewritten the Douglas-Rachford algorithm in the form of proximal point algorithm. Moreover, Eckstein has applied the Douglas–Rachford algorithm to solve the dual of a certain structured convex optimization problem, which coincides with the socalled alternating direction method of multipliers.

The Douglas-Rachford algorithm proposed by Lions and Mercier [60] is given as below:

$$x_{n+1} = \frac{1}{2}(Id + R_A \circ R_B)x_n, \tag{1.10}$$

where A and B are set-valued maximal monotone operators. The convergence analysis of the Douglas-Rachford algorithm is studied in [58, 60] and it is proved that the sequence generated by the algorithm converges weakly to a fixed point, which lies in the solution set of inclusion problem (1.8). The result was further reinforced by Svaiter [87] by studying the weak convergence of the shadow sequences. Recently, the Douglas-Rachford algorithm has drawn the attention of researchers due to its applicability to solve inclusion problems both in convex as well as in non-convex settings [10, 36, 56]. This algorithm is used to solve large scale optimization problems arising in machine learning, finance, control, image processing, and PDEs (see [6, 10, 20, 36, 56, 62, 78]).

Problem solving strategy for the case, when one of the operators A or B is single-valued, has been evolved separately. Consider A and B to be maximal monotone operators on $\mathcal{H}$ such that B is single-valued and $domB \supset domA$, Lions and Mercier [60] have proposed the forward-backward algorithm, which is given by

$$x_{n+1} = (Id + \lambda_n A)^{-1}(Id - \lambda_n B)(x_n), \tag{1.11}$$

where $\lambda_n > 0$. Gabay and Mercier ([43, 65]) have proved the weak convergence of proximal point algorithm for B^{-1}, a strongly monotone operator with modulus α such that $\lambda_n \in (0, \alpha)$ is kept constant. It shows strong convergence if A is also strongly monotone. Han and Lou [45] have proposed the dual form of forward-backward splitting algorithm for convex programming. The forward-backward method has a nice property that we can put the dense part of operator T as B which facilitates the problem decomposition. Tseng [94] has ruled out the requirement of B^{-1} or T to be strongly monotone by including an additional forward step. He has proposed the forward-backward-forward algorithm with the only assumption that forward operator B be Lipschitz continuous on some closed convex subset of its domain. For $x_0 \in \mathcal{H}$, the forward-backward-forward algorithm is given by,

$$\begin{cases} p_n = (Id + \lambda_n A)^{-1}(Id - \lambda_n B)(x_n) \\ x_{n+1} = (Id - \lambda_n B)(p_n) + \lambda_n B(x_n), \end{cases} \tag{1.12}$$

where $\lambda_n \in [\epsilon, (1-\epsilon)/L]$ with $\epsilon \in (0, \frac{1}{L+1})$ and L is a Lipschitz constant of B. The convergence of the algorithm (1.12) is guaranteed when the solution set of inclusion problem (1.8) is nonempty.

1.3 Inertial Methods

In the present world of high-dimensional datasets, we do not just need the methods to solve the problem but the speed of the convergence of the methods is also equally important. To achieve the fast convergence speed, different algorithms having faster convergence speed and fewer restrictions on the parameters had been proposed. In this direction, the inertial term has been introduced which increases the convergence speed without increasing the computing cost of the algorithm. The inertial term can be seen as the discretized form of the time dynamical system

$$\ddot{x}(t) + \alpha_1 \dot{x} + \alpha_2 f(x(t)) = 0,$$

where $\alpha_1, \alpha_2 > 0$ are free model parameters of the equation. The inertial term is first introduced in Heavy Ball method proposed by Polyak [79] to minimize a smooth convex function f and has the following algorithm:

$$\begin{cases} y_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = y_n - \lambda_n \nabla f(x_n) \quad \text{for all} \;\; n \in \mathbb{N}, \end{cases} \tag{1.13}$$

where $\alpha_n \in [0, 1)$ is an extrapolation factor and λ_n is a step-size parameter. In 2001, Alvarez and Attouch [4] used the idea of Polyak to improve the performance of proximal point algorithm for the monotone operators and proposed the inertial proximal point algorithm is given by

$$\begin{cases} y_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = (Id + \lambda_n T)^{-1}(y_n) \quad \text{for all } n \in \mathbb{N}, \end{cases} \tag{1.14}$$

where T is a monotone operator and $\{\lambda_n\}$ is a nondecreasing sequence. He has also proved in [4] that the algorithm (1.14) converges weakly to a zero of T if $\{\alpha_n\}$ is in $[0, 1)$ and satisfies the condition:

$$\sum_{n=1}^{\infty} \alpha_n \|x_n - x_{n-1}\|^2 < \infty. \tag{1.15}$$

In order to improve the convergence rate of the Heavy Ball method, Nesterov [72] has proposed a modification in the above iteration method (1.17) and proposed the following algorithm:

$$\begin{cases} y_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = y_n - \lambda_n \nabla f(y_n) \quad \text{for all } n \in \mathbb{N}, \end{cases} \tag{1.16}$$

where $\lambda_n = 1/L$, L is the Lipschitz constant of f.

The inertial extrapolation based algorithms have been studied by researchers and been implemented in several directions (see [15, 31, 73] and references therein). Recently, using inertial extrapolation researchers have constructed many iterative algorithms, such as fast iterative shrinkage thresholding algorithm (FISTA) [11], inertial forward-backward algorithm [61] and inertial Douglas-Rachford splitting algorithm [16]. Beck and Teboulle [11] had proposed the fast iterative shrinkage thresholding algorithms, which is popularly known as FISTA. This algorithm has dominated the optimization world in the last decade. Consider $g : \mathbb{R}^d \to \mathbb{R}$ as a continuous convex function and $f : \mathbb{R}^d \to \mathbb{R}$ is a smooth convex function with continuously differentiable with Lipschitz continuous gradient L. Then, for $y_0 = x_0 = u_0$, FISTA is given by

$$\begin{cases} x_n = \operatorname{prox}_{\gamma g}(Id - \gamma \nabla f)(y_{n-1}) \\ u_n = x_{n-1} + t_n(x_n - x_{n-1}) \\ y_n = \left(1 - \frac{1}{t_{n+1}}\right) x_n + \frac{1}{t_{n+1}} u_n \quad n \in \mathbb{N}_0, \end{cases} \tag{1.17}$$

where $\{t_n\}$ is a sequence of real numbers greater than 1 and $\gamma > 0$.For more information regarding on inertial methods, we refer to [12, 15, 29, 31, 73].

1.4 Fixed Point Methods

The monotone inclusion problems can also be solved by formulating them into the fixed point problem. Consider an operator $T : \mathcal{H} \to \mathcal{H}$. The fixed point problem is to find $x \in \mathcal{H}$ such that

$$(Id - T)x = 0 \tag{1.18}$$

The solution set of fixed point problem is denoted by $\operatorname{Fix}(T)$. The monotone inclusion problem can be reduced into the fixed point problem by considering $T = (Id + \lambda A)^{-1}(Id - \lambda B)$. The fixed point approach is largely used to solve problems in information theory, game theory, optimization, etc. by formulating them into fixed point problems. One of the most used iterative techniques has been introduced by Mann[64] in 1953, which is given as follows:

$$\begin{cases} x_1 \in C \\ x_{n+1} = (1 - \alpha_n)x_n + \alpha_n T x_n, \qquad \text{for all } n \in \mathbb{N}, \end{cases} \tag{1.19}$$

where $\{\alpha_n\}$ is a real sequence in $(0,1)$. The sequence $\{x_n\}$ defined by (1.19) converges weakly to a fixed point of T if the iteration parameter $\{\alpha_n\}$ satisfies the condition $\sum_{n=1}^{\infty} \alpha_n(1-\alpha_n) = \infty$.

It is well known that the Mann iteration method for the approximation of fixed points of pseudo contractive mappings may not well behave (see [33]). To get rid of this problem, in 1974 Ishikawa [47] has introduced an iterative technique, which is extensively studied for the approximation of fixed points of pseudo contractive and nonexpansive mappings by many authors in different spaces (see for example [1, 38, 88]).

In 2007, Agarwal et al. [3] have introduced an iteration method which is called S-iteration method. Its convergence rate is faster than both Mann and Ishikawa iteration methods for contraction mappings. The S-iteration is defined by

$$\begin{cases} x_{n+1} = (1-\alpha_n)Tx_n + \alpha_n Ty_n, \\ y_n = (1-\beta_n)x_n + \beta_n Tx_n \quad \text{for all } n \in \mathbb{N}, \end{cases} \tag{1.20}$$

where $\{\alpha_n\}$ and $\{\beta_n\}$ are sequences in $(0,1)$ with $\sum_{n=1}^{\infty} \alpha_n\beta_n(1-\beta_n) = \infty$. The algorithmic design of S-iteration method is comparatively different and independent of Mann and Ishikawa iteration methods, i.e. neither Mann nor Ishikawa iterative technique can be reduced into S-iteration and vice versa. In 2011, Sahu [82] came up with another form of S-iteration, named as normal S-iteration method which is defined by

$$x_{n+1} = T[(1-\beta_n)x_n + \beta_n Tx_n] \quad \text{for all } n \in \mathbb{N}, \tag{1.21}$$

where $\{\beta_n\}$ is a sequence in $(0,1)$. Normal S-iteration is also known as Hybrid-Picard Mann iteration method [51]. The performance of normal S-iteration method is better than Picard method for contraction mappings. In sense of effective performance, S-iteration method and normal S-iteration method have attracted many researchers as alternative iteration method for common fixed point problems (see [30, 34, 86, 97, 98]).

On the other hand, in 2014, in order to accelerate Halpern fixed point algorithm, Sakurai and Liduka [83] have discussed the importance to find fixed point quickly for the class of steady and authentic practical systems. So there are increasing interests in the study of fast algorithms for approximating fixed points of nonexpansive mappings. In 2008, Mainge [63] has combined the inertial type extrapolation algorithm with the classical Mann algorithm and named as inertial Mann iteration method for finding the fixed points of nonexpansive mappings in a real Hilbert space. Inertial Mann algorithm is defined as follows:

$$\begin{cases} y_n = x_n + \alpha_n(x_n - x_{n-1}), \\ x_{n+1} = (1-\lambda_n)y_n + \lambda_n T y_n \quad \text{for all } n \in \mathbb{N}. \end{cases} \tag{1.22}$$

Mainge [63] has further studied weak convergence of inertial Mann algorithm (1.22) under the following conditions:

(C_0) $\alpha_n \in [0, \alpha]$ for all $n \in \mathbb{N}$ and $\alpha \in [0, 1)$;

(C_1) $\sum_{n=1}^{\infty} \alpha_n \|x_n - x_{n-1}\|^2 < \infty$;

(C_2) $\inf_{n\geq 1} \lambda_n \geq 0$ and $\sup_{n\geq 1} \lambda_n \leq 1$.

Moreover, we consider the inclusion problems when it contains the composition of linear operator or composition of parallel-sum operators as one of the operators. Since in general resolvent of the composition of operators and resolvent of parallel-sum operators are not present in closed form, Douglas-Rachford algorithm is not applicable to solve the structured monotone inclusion problem. In recent studies [16, 18, 35, 95], several primal-dual methods have been proposed to solve such problems. Combettes et al. [35] have proposed an inexact Tseng algorithm to solve structured problems containing set-valued operators in composition with bounded linear operator and Lipschitz operator. Vũ [95] has replaced Lipschitz operators with cocoerceive operators and proposed an inexact forward-backward algorithm to solve this problem. Bot et al. [16, 18] proposed Douglas-Rachford type primal-dual method and inertial Douglas-Rachford type primal-dual method to solve the structured monotone inclusion problem.

1.5 Outline of the Thesis

In this subsection, the author describes the outline of the work done in each chapter. The thesis has possibly made independent by describing the definition and results useful for better understanding of the thesis in each chapter.

The next chapter is dedicated to the study of an inertial fixed point algorithm to find the fixed point of a nonexpansive operator. We also study the application of the proposed algorithm to solve the convex optimization problem. The numerical experiments have been conducted on the high-dimensional dataset to solve the regression problem.

In Chapter 3, we propose a novel accelerated preconditioning forward-backward algorithm to obtain the vanishing point of the sum of two operators in which one is maximal monotone and another is M-cocoercive, where M is a linear bounded operator on underlying spaces. Our proposed algorithm is more general than previously known algorithms. We study the convergence behavior of the proposed algorithm under mild assumptions in the framework of real Hilbert spaces. We employ our model to solve regression problems and link prediction problems for high-dimensional datasets and conduct numerical experiments to support our results. This model improves convergence speed and accuracy in respective problems.

In Chapter 4, we propose a novel two-step inertial Douglas-Rachford algorithm to solve the monotone inclusion problem of the sum of two maximally monotone operators. We study the convergence behavior of the proposed algorithm. Based on the proposed method, we develop an inertial primal-dual algorithm to solve highly structured monotone inclusions containing the mixture of linearly composed and parallel-sum type operators. Finally, we apply the proposed inertial primal-dual algorithm to solve a highly structured minimization problem. We also perform a numerical experiment to solve the generalized Heron problem and compare the performance of the proposed inertial primal-dual algorithm to those of already known algorithms.

In Chapter 5, we propose a fixed point algorithm based on normal-S iteration to find common fixed point of nonexpansive operators and prove the strong convergence of the generated sequence to the set of common fixed points without assuming strong convexity and strong monotonicity. Based on the proposed fixed point algorithm, we develop a new forward-backward algorithm and a Douglas-Rachford algorithm in connection with Tikhonov regularization to find the solution of splitting monotone inclusion problem. We also propose a strongly convergent forward-backward type primal-dual algorithm and a Douglas-Rachford type primal-dual algorithm such that they solve the monotone inclusion problems containing the mixture of linearly composed and parallel-sum operators. Finally, we have conducted a numerical experiment to solve image deblurring problems.

****