STABILITY PROPERTIES OF THE GRADIENT PROJECTION METHOD WITH APPLICATIONS TO THE BACKPROPAGATION ALGORITHM[*]

M. V. Solodov[†] and S. K. Zavriev[‡]

June 6, 1994

ABSTRACT

Convergence properties of the generalized gradient projection algorithm in the presence of data perturbations are investigated. It is shown that every trajectory of the method is attracted, in a certain sense, to an ε-stationary set of the problem, where ε depends on the magnitude of the perturbations. Estimates for the attraction sets of the iterates are given in the general (nonsmooth and nonconvex) case. In the convex case, our results imply convergence to an ϵ-optimal set. The results are further strengthened for weakly sharp and strongly convex problems. Convergence of the parallel algorithm in the case of the additive objective function is established. One of the principal applications of our results is the stability analysis of the classical backpropagation algorithm for training artificial neural networks.

KEY WORDS. gradient projection, error-stability, parallelization, backpropagation convergence.

[*] The first author is supported by Air Force Office of Scientific Research Grant F49620-94-1-0036 and National Science Foundation Grant CCR-9101801.

[†] Computer Sciences Department, University of Wisconsin, 1210 West Dayton Street, Madison, WI 53706, U.S.A. Email : *solodov@cs.wisc.edu.*

[‡] Center for the Mathematical Sciences, University of Wisconsin, Madison, WI 53715. Fulbright Scholar, on leave from Operations Research Department, Faculty of Computational Mathematics and Cybernetics, Moscow State University, Moscow, Russia, 119899.

1 Introduction

We consider the following general optimization problem

$$\min_{x \in X} f(x), \tag{1.1}$$

where X is a convex compact set in $\Re^n$, and the objective function $f : X \to \Re$ is at least Lipschitz continuous on X and regular (in the sense of Clarke, [2]).

Let X_{opt} and X_{stat} denote the optimal and stationary sets of problem (1.1) respectively, that is

$$X_{opt} = \{x \in X \mid f(x) = \min_{y \in X} f(y)\},$$

$$X_{stat} = \{x \in X \mid 0 \in \partial f(x) + N_X(x)\},$$

where $\partial f(x)$ is the set of all generalized gradients (in the sense of Clarke, [2]) of $f(\cdot)$ at x, and $N_X(x) \subset \Re^n$ is the normal cone to the set X at the point $x \in X$:

$$N_X(x) = \{y \in \Re^n \mid \forall z \in X \ \langle y, z - x\rangle \leq 0\}.$$

The following notions will play an important role in our analysis. Let $\varepsilon : X \to \Re^+$ be any nonnegative upper semicontinuous function. We introduce the ***$\varepsilon(\cdot)$ -stationary set*** of the problem (1.1) as follows :

$$X_{stat}(\varepsilon(\cdot)) = \{x \in X \mid 0 \in \partial f(x) + N_X(x) + \varepsilon(x)B\},$$

where B is the closed unit ball in $\Re^n$, that is $B = \{x \in \Re^n \mid \|x\| \leq 1\}$.

In this paper we establish convergence properties of the generalized gradient projection method and its modifications (see Algorithms 3.1,3.2,4.1) in the presence of nonvanishing noise. In particular, we show that the iterates of the algorithm are, in a certain sense, attracted to an $\varepsilon(\cdot)$-stationary set of the problem (Theorem 3.1). We give a precise estimate for $\varepsilon(\cdot)$ in terms of asymptotic behavior of the perturbations. Our analysis is based on the novel technique developed in [26]. This approach allows us to treat essentially perturbed problems (i.e. problems with nonvanishing noise), as well as analyze the algorithms that are inherently nonmonotone (see Algorithm 3.1).

For every $x \in X$ we define a nonnegative scalar function $r : X \to \Re^+$ by the following relation :

$$r(x) := \{\min \|h\| \mid h \in \partial f(x) + N_X(x)\}. \tag{1.2}$$

It is clear that $r(\cdot)$ is an optimality function for problem (1.1) in the sense that

$$r(x) \begin{cases} = 0 & \text{if } x \in X_{stat} \\ > 0 & \text{otherwise} \end{cases}$$

From the definitions of $X_{stat}(\varepsilon(\cdot))$ and $r(x)$, we immediately obtain the following useful relation

$$X_{stat}(\varepsilon(\cdot)) = \{x \in X \mid r(x) \leq \varepsilon(x)\}. \tag{1.3}$$

For any nonnegative upper semicontinuous function $\epsilon : X \to \Re^+$, we define the $\epsilon(\cdot)$-optimal set of (1.1) as follows :

$$X_{opt}(\epsilon(\cdot)) = \{x \in X \mid f(x) \leq \min_{y \in X} f(y) + \epsilon(x)\}.$$

Obviously, $X_{opt}(0) = X_{opt}$. In the convex case the sets $X_{stat}(\varepsilon(\cdot))$ and $X_{opt}(\epsilon(\cdot))$ are related in a certain way (see Lemma 4.2). In that case many of our general results can be considerably strengthened (see Section 4).

Let $\mathcal{F}(\cdot,\cdot) : \mathcal{N} \times X \to \mathcal{M}(\Re^m)$ be a point-to-set mapping (or a multifunction), where $\mathcal{M}(C)$ denotes the set of all subsets of a set C, and $\mathcal{N}$ denotes the nonnegative integers. We define the *upper topological limit* of $\mathcal{F}(\cdot,\cdot)$ at $x \in \Re^n$ by

$$\bar{\text{lt}}_{\substack{x'(\in X)\to x \\ i\to\infty}} \mathcal{F}(i, x') := \left\{ y \in \Re^m \;\middle|\; \begin{array}{l} \text{there exist sequences } \{m_i\} \to \infty, \{x_i'\} \to x \text{ as } i \to \infty, x_i' \in X, \\ \text{and } \{y_i\}, y_i \in \mathcal{F}(m_i, x_i'), i = 1, 2, \dots\,, \text{ such that } y = \lim_{i\to\infty} y_i \end{array} \right\}$$

In particular, for a bounded sequence $\{x_i\}$, $x_i \in X$, $\bar{\text{lt}}_{i\to\infty}\{x_i\}$ denotes the set of all limit points of $\{x_i\}$. We say that a sequence $\{x_i\}$ converges to a set C, if $\bar{\text{lt}}_{i\to\infty}\{x_i\} \subset C$. Note that under our assumptions,

$$\bar{\text{lt}}_{x'(\in X)\to x} N_X(x') = N_X(x) \;\; \forall x \in X. \tag{1.4}$$

Of particular interest for us will be an extension of problem (1.1) to the case where the objective function $f(\cdot)$ is given by a summation of a finite number of functions $f_j(\cdot, \alpha_0)$, $j = 1, \ldots, K$. Note that we further allow the dependence of f_j on a parameter. We thus consider the problem

$$\min_{x \in X} f(x, \alpha_0) := \sum_{j=1}^{K} f_j(x, \alpha_0). \tag{1.5}$$

For every $j = 1, \ldots, K$ the function $f_j : \Re^n \times A \to \Re$ involves a parameter $\alpha \in A \subset \Re$ that may vary during the optimization process. We assume that A is bounded. Problems of the form (1.5) arise, for example, in least-norm minimization, neural networks applications, and approximation theory. Among some important practical applications that involve parameters in the objective function, we note the adaptive smoothing techniques [12], and the neural network training [18, 10, 9]. We assume that each function $f_j(\cdot, \alpha)$ is Lipschitz continuous with modulus $L > 0$ and regular on an open neighborhood of X for every $\alpha \in A$. We also assume that the map $\partial f_j(\cdot, \cdot)$ is upper semicontinuous. That is, for all j

$$\bar{\mathrm{l}}\mathrm{t}_{\substack{x'(\in X) \to x \\ \alpha(\in A) \to \alpha_0}} \partial f_j(x', \alpha) \subset \partial f_j(x, \alpha_0) \;\; \forall x \in X, \tag{1.6}$$

where $\partial f_j(x, \alpha)$ denotes the set of all generalized gradients of $f_j(\cdot, \alpha)$ at $x \in X$.

The rest of the paper is organized as follows. In section 2 we outline the Generalized Lyapunov Direct Method for stability analysis. In Section 3 we establish convergence properties of the generalized gradient projection method and its modifications in the presence of data perturbations. Section 4 contains the results that are strengthened for the case of weakly sharp and convex problems. We relate our work to the neural networks applications in Section 5. Section 6 contains some concluding remarks.

One more word about our notation. All the vectors are column-vectors. $\langle \cdot, \cdot \rangle$ denotes the usual Euclidean inner product. Throughout the paper, $\| \cdot \|$ denotes the two-norm, that is $\|x\| = \sqrt{\langle x, x \rangle}$. By *conv* C we shall denote the convex hull of a set C, and by *int* C its interior. $P_X(\cdot)$ will stand for the orthogonal projection map onto a closed convex set X.

2 Generalized Lyapunov Direct Method

In this Section we outline the novel convergence analysis technique that was first proposed in [26]. This technique can be viewed as generalization of the Lyapunov Direct Method for convergence analysis of nonlinear iterative processes. The Lyapunov Direct Method has proved to be a powerful tool for stability analysis of both continuous and discrete time processes [17, 23, 15, 16]. Roughly, this approach reduces the analysis of the stability properties of a process to the analysis of the local improvement of this process with respect to some scalar criterion $V(\cdot)$ (usually called the *Lyapunov function*). According to the classical approach, $V(\cdot)$ is assumed to be a descent function of the process [16]. The key difference of the presented technique is that we relax this monotonicity assumption. We thus refer to $V(\cdot)$ as the ***pseudo-Lyapunov function***. This generalization makes our approach applicable to a wider class of algorithms.

We now state the Generalized Lyapunov Direct Method. The convergence (attraction) properties of the process are expressed in terms of pseudo-Lyapunov function $V(\cdot)$. For each specific algorithm, the results allow further interpretation depending on the choice of $V(\cdot)$.

We consider the following iterative process

$$x^{i+1} \in x^i - \eta_i G(i, x^i) - \xi_i, \;\; i = 0, 1, \dots, \;\; x^0 \in X', \tag{2.1}$$

$$\eta_i \to \infty, \;\; \sum_{i=0}^{\infty} \eta_i = \infty, \;\; \sum_{i=0}^{\infty} \xi_i \text{ is (component-wise) convergent,} \tag{2.2}$$

where $G(\cdot,\cdot) : X' \to \mathcal{M}(X')$, and X' is an open set in $\Re^n$. In applications, ξ_i usually corresponds to random noise. We further assume that

$$\sup_{x \in X'} \limsup_{\substack{x' \to x \\ i \to \infty}} \sup_{y \in G(i,x')} \|y\| < +\infty \; .$$

Thus the upper topological limit of $G(\cdot,\cdot)$, denoted by $G_0(\cdot)$,

$$G_0(x) := \overline{\mathrm{lt}}_{\substack{x' \to x \\ i \to \infty}} G(i, x')$$

is bounded and upper semicontinuous on a neighborhood of a compact set $X \subset X'$.

Let the iterates generated by (2.1)-(2.2) satisfy the condition

$$\overline{\mathrm{lt}}\{x^i\} \subset X. \tag{2.3}$$

Suppose we have chosen a pseudo-Lyapunov function $V(\cdot)$ that is Lipschitz continuous and regular on a neighborhood of X. For the Lyapunov function $V(\cdot)$, the set X, and the map $G_0(\cdot)$, we define the following set which is crucial for our analysis :

$$\begin{aligned} \mathcal{A}_0 &:= \{x \in X \mid \max_{h \in H(x)} \min_{g \in G_0(x)} \langle h, g\rangle \leq 0\} \\ H(x) &= conv\{\partial V(x) \cup N_X(x)\}. \end{aligned} \tag{2.4}$$

Roughly speaking, the set $\mathcal{A}_0$ is comprised of all the points in X for which $-G_0(x)$ does not contain feasible directions that are of descent for the pseudo-Lyapunov function $V(\cdot)$.

The following result shows that the sequences of (2.1)-(2.2) that satisfy (2.3) are, in a certain sense, attracted to the components of the set $\mathcal{A}_0$. We first have to introduce the notion of $V(\cdot)$-connected components of $\mathcal{A}_0$ (recall that $\mathcal{A}_0$ is compact). We say that a set $C \subset \Re^n$ is $V(\cdot)$-connected, if the set $V(C) = \{v \in \Re \mid \exists x \in X, \;\; v = V(x)\} \subset \Re$ is connected. Let $\{\mathcal{A}^{(\gamma)}\}, \;\; \gamma \in \Gamma$ be the (unique) decomposition of $\mathcal{A}_0$ into $V(\cdot)$-connected components [24], that is

$$\mathcal{A}_0 = \cup_{\gamma \in \Gamma} \mathcal{A}^{(\gamma)}, \;\; \mathcal{A}^{(\gamma')} \neq \mathcal{A}^{(\gamma'')}, \; \gamma' \neq \gamma'', \;\; \gamma', \gamma'' \in \Gamma.$$

The following theorem will play an important role in the subsequent analysis.

Theorem 2.1 *[26] For every sequence $\{x^i\}$ generated by the process (2.1)-(2.2), and satisfying (2.3), there exists a $\gamma \in \Gamma$ such that the following properties hold :*

$$\overline{lt}_{i \to \infty} V(x^i) = V\left(\overline{lt}_{i \to \infty}\{x^i\} \cap \mathcal{A}^{(\gamma)}\right),$$

and every subsequence $\{x^{i_m}\}$ of $\{x^i\}$ satisfying

$$\lim_{m \to \infty} V(x^{i_m}) = \liminf_{i \to \infty} V(x^i) \quad or \quad \lim_{m \to \infty} V(x^{i_m}) = \limsup_{i \to \infty} V(x^i)$$

converges to $\mathcal{A}^{(\gamma)}$.

Corollary 2.1 *[26] Let the set $V(\mathcal{A}_0)$ be nowhere dense in $\Re$. Then every sequence $\{x^i\}$ generated by the process (2.1)-(2.2), and satisfying (2.3), converges to a connected component of $\mathcal{A}_0$.*

3 Convergence Properties of Parallel Generalized Gradient Projection Algorithm in the Presence of Data Perturbations

In this Section we consider the problem (1.5) with an additive parametric objective function. We first describe our notation for stating and establishing convergence of the parallel perturbed generalized gradient projection method (GGPM) for solving (1.5) and its modifications. The type of parallelization considered here is primarily motivated by neural network training (see [11, 14, 4]). Another related work is [21]. We first consider the most general case. Our results can be then specialized by removing parallelism and/or considering the standard (nonadditive) objective function.

$\boldsymbol{i = 1, 2, \ldots}$: Index number of major iterations of GGPM, each of which consists of going through the entire set of functions $f_1(x, \alpha_i), \ldots, f_K(x, \alpha_i)$. This is achieved serially or in parallel by k processors with processor l handling at the i-th iteration the functions $f_j(x, \alpha_i)$, $j \in J_l$. Recall that $\alpha_i \in A$ is the (smoothing) parameter, and $\lim_{i \to \infty} \alpha_i = \alpha_0$. For simplicity, we assume that the sets J_l, $l = 1, \ldots, k$ are ordered as follows

$$\begin{aligned}
&J_1 = \{1, \ldots, K_1\},\\
&J_2 = \{K_1 + 1, \ldots, K_1 + K_2\},\\
&\cdots\ \cdots\ \cdots\ \cdots\ \cdots\ \cdots\\
&J_k = \{K_1 + \cdots + K_{k-1} + 1, \ldots, K\},
\end{aligned}$$

i.e.

$$J_l = \{\bar{K}_l + 1, \ldots, \bar{K}_l + K_l\},\ l = 1, \ldots, k,$$

where

$$\bar{K}_l = \sum_{t=1}^{l-1} K_t,\ l = 2, \ldots, k,\ \bar{K}_1 = 0.$$

$\boldsymbol{j = 1, \ldots, K_l}$: Index of minor iterations performed by parallel processor l, $l = 1, \ldots, k$. Each minor iteration j consists of a step in the direction of a negative generalized gradient

$-\tilde{g}^l_{i,j}$ of the function $f_{\bar{K}_l+j}(\cdot,\alpha_i)$ at $z_l^{i,j}$ that is calculated with some error $\delta_l^{i,j}$:

$$\begin{array}{l} \tilde{g}^l_{i,j} = g^l_{i,j} + \delta_l^{i,j}, \\ g^l_{i,j} \in \partial f_{\bar{K}_l+j}(z_l^{i,j},\alpha_i), \\ \delta_l^{i,j} = \delta_{\bar{K}_l+j}(z_l^{i,j},\alpha_i,i). \end{array}$$

Note that $\delta_j(z,\alpha,i)$ is a perturbation of the generalized gradient of $f(\cdot,\alpha)$ at the point $z \in X$ at the i-th major iteration of the algorithm. With respect to those perturbations we make the following fairly mild assumption :

$$\sum_{j=1}^{K} \sup_i \sup_{z\in X} \sup_{\alpha\in A} \|\delta_j(z,\alpha,i)\| < +\infty.$$

$\boldsymbol{x^i}$: Iterate in $\Re^n$ of major iteration $i = 1, 2, \ldots$.

$\boldsymbol{z_l^{i,j}}$: Iterate in $\Re^n$ of minor iteration $j = 1, \ldots, K_l$, within major iteration $i = 1, 2, \ldots$, computed by processor $l = 1, \ldots, k$.

By $\boldsymbol{\eta_i}$ we shall denote the stepsize, i.e the coefficient multiplying the generalized gradients at the i-th major iteration. For simplicity we shall assume that η_i remains fixed within each major iteration. We consider the process with stepsizes decreasing subject to the following condition

$$\eta_i > 0,\ i = 0, 1, \ldots,\quad \eta_i \to 0,\ \sum_{i=0}^{\infty} \eta_i = \infty. \tag{3.1}$$

Note that under our assumptions, there exists $M > 0$ such that

$$\|y\| \le M \quad \forall y \in \partial f_j(x,\alpha_i) + \delta_j(x,\alpha_i,i),\ j = 1, \ldots, K,\ \ i = 0, 1, \ldots,\ \ \forall x \in X. \tag{3.2}$$

We are now ready to state and prove convergence of the parallel GGPM.

Algorithm 3.1 ***(Parallel GGPM)*** *Start with any $x^0 \in X$. Having x^i, compute x^{i+1} as follows :*

(i) Parallelization: for each processor $l \in \{1, \ldots, k\}$ do

$$z_l^{i,j+1} = P_X(z_l^{i,j} - \eta_i \tilde{g}^l_{i,j}),\ \ j = 1, \ldots, K_l \tag{3.3}$$

(ii) Synchronization

$$x^{i+1} = P_X\left(x^i + \sum_{l=1}^{k}(z_l^{i,K_l+1} - x^i)\right) \tag{3.4}$$

Note that for $K = k = 1$ Algorithm 3.1 becomes the standard perturbed generalized gradient projection method. There are two sources of nonmonotonicity that are present in Algorithm 3.1. First of all, each direction is associated with a generalized gradient of a partial objective function f_j. Thus even if this direction is that of descent for f_j, there is no guarantee that it is also of descent for the full objective function f given by (1.5). The other source of nonmonotonicity is induced by the perturbations of the generalized gradients.

To analyze the influence of computational errors $\delta_l^{i,j}$ on the convergence properties of the algorithm, we need to estimate the level of perturbations in the limit. We say that $\varepsilon(x)$ is the ***exact asymptotic level of perturbations*** at a point $x \in X$, if

$$\varepsilon(x) = \limsup_{\substack{z_j(\in X) \to x \\ i \to \infty}} \| \sum_{j=1}^{K} \delta_j(z_j, \alpha_i, i) \|. \tag{3.5}$$

It is easy to see that the function $\varepsilon(\cdot) : X \to \Re^+$ is upper semicontinuous.

The following simple lemma proves to be very useful.

Lemma 3.1 *For every $x \in X$, $g \in \Re^n$, and $\eta > 0$ the following property holds*

$$y = P_X(x - \eta g) \Longrightarrow \exists h \in N_X(y),\ \|h\| \le \|g\|,\ \ y = x - \eta(g + h). \tag{3.6}$$

The proof requires only elementary arguments, and is thus omitted.

Taking into account (3.6), we introduce the following map $G(\cdot,\cdot) : \mathcal{N} \times X \to \Re^n$ that is associated with major iterates of Algorithm 3.1 :

(3.7)

$$G(i,x) = \left\{ y \in \Re^n \;\middle|\; \begin{array}{l} \exists \bar{x} \in X,\ z_l^j \in X,\ g_l^j \in \partial f_{\bar{K}_{l-1}+j}(z_l^j, \alpha_i), \text{ and } \bar{h} \in N_X(\bar{x}) \text{ such that} \\ \|\bar{h}\| \le 2MK,\ h_l^j \in N_X(z_l^{j+1}),\ \|h_l^j\| \le M, \\ j = 1, \dots, K_l + 1,\ l = 1, \dots, k, \\ y = \sum_{l=1}^k \sum_{j \in J_l} (g_l^j + h_l^j + \delta_l^j) + \bar{h},\ \ z_l^{j+1} = z_l^j - \eta_i (g_l^j + h_l^j + \delta_l^j), \\ \delta_l^j = \delta_{\bar{K}_l+j}(z_l^j, \alpha_i, i),\ j = 1, \dots, K_l,\ z_l^1 = x,\ l = 1, \dots, k, \\ \bar{x} = x + \sum_{l=1}^k (z_l^{K_l+1} - x) + \bar{h} \end{array} \right\}$$

Obviously, by (3.2),

$$\|y\| \le 4KM \ \ \forall y \in G(i,x),\ \ i = 0, 1, \dots,\ \ \forall x \in X.$$

Hence the map $G(\cdot,\cdot)$ is bounded, and so is its upper topological limit. Comparing (3.7) with (3.3) and (3.4), and taking into account (3.6), it is easily seen that every sequence $\{x^i\}$ generated by Algorithm 3.1 is a trajectory of the iterative process

$$x^{i+1} \in x^i - \eta_i G(i, x^i), \quad i = 0, 1, \dots , \quad x^0 \in X.$$

We are now ready to apply the Generalized Lyapunov Direct Method of Section 2 to establish the properties of Algorithm 3.1.

Applying (1.4), (1.6), and the definition (3.5) of $\varepsilon(\cdot)$, we obtain

$$G_0(x) := \bar{\mathrm{l}}\mathrm{t}_{\substack{x' \to x \\ i\to\infty}} G(i, x') \subset \partial f(x) + N_X(x) + \varepsilon(x) B. \tag{3.8}$$

Consider the decomposition of of the set $X_{stat}(\varepsilon(\cdot))$ into the union of $f(\cdot)$-connected components

$$X_{stat}(\varepsilon(\cdot)) = \cup_{\gamma\in\Gamma} X_{stat}(\varepsilon(\cdot))^{(\gamma)}$$

(see Section 2). Our main result is the following

Theorem 3.1 *For every sequence $\{x^i\}$ generated by Algorithm 3.1, there exists $\gamma \in \Gamma$ such that the following properties hold :*

$$\bar{l}t_{i\to\infty} f(x^i) = f\left(\bar{l}t_{i\to\infty}\{x^i\} \cap X_{stat}(\varepsilon(\cdot))^{(\gamma)}\right),$$

and every subsequence $\{x^{i_m}\}$ of $\{x^i\}$ satisfying

$$\lim_{m\to\infty} f(x^{i_m}) = \liminf_{i\to\infty} f(x^i) \quad or \quad \lim_{m\to\infty} f(x^{i_m}) = \limsup_{i\to\infty} f(x^i) \tag{3.9}$$

converges to $X_{stat}(\varepsilon(\cdot))^{(\gamma)}$.

In particular, if $\varepsilon(\cdot) \equiv 0$ and the set $f(X_{stat})$ is nowhere dense in $\Re$, then every sequence $\{x^i\}$ generated by Algorithm 3.1 converges to a connected component of X_{stat}.

Proof. We choose

$$V(x) := f(x),$$

where $f(x)$ is given by (1.5), as the pseudo-Lyapunov function of the iterative process. Following the approach outlined in Section 2, we introduce the set

$$\mathcal{A}_0 := \{x \in X \mid \max_{h \in H(x)} \min_{g \in G_0(x)} \langle h, g \rangle \leq 0\},$$

where $H(x) := conv\{\partial f(x) \cup N_X(x)\}$. Our proof is by virtue of showing that

$$\mathcal{A}_0 \subset X_{stat}(\varepsilon(\cdot)),$$

and then applying Theorem 2.1 and Corollary 2.1.

For every $x \in X$ we define

$$h_0(x) = \arg\min\{\|h\| \mid h \in \partial f(x) + N_X(x)\}.$$

Note that $\|h_0(x)\| = r(x)$ (see (1.2)). Since $h_0(x)$ is the orthogonal projection of the origin onto the set $\{\partial f(x) + N_X(x)\}$, it follows that

$$\langle h_0(x), h \rangle \geq \|h_0(x)\|^2 \ \ \forall h \in \partial f(x) + N_X(x). \tag{3.10}$$

Since $h_0(x) \in \partial f(x) + N_X(x)$, it follows that

$$\frac{1}{2} h_0(x) \in H(x). \tag{3.11}$$

Fix an arbitrary $x \notin X_{stat}(\varepsilon(\cdot))$. By (1.3), we have

$$\|h_0(x)\| = r(x) > \varepsilon(x). \tag{3.12}$$

We further obtain

$$\begin{aligned}
\max_{h \in H(x)} \min_{g \in G_0(x)} \langle h, g \rangle &\geq \frac{1}{2} \min_{g \in G_0(x)} \langle h_0(x), g \rangle \\
&\geq \frac{1}{2} \min_{g \in \partial f(x) + N_X(x) + \varepsilon(x) B} \langle h_0(x), g \rangle \\
&\geq \frac{1}{2} \min_{\delta \in \varepsilon(x) B} \ \min_{h \in \partial f(x) + N_X(x)} \langle h_0(x), h + \delta \rangle \\
&\geq \frac{1}{2} \min_{\delta \in \varepsilon(x) B} \langle h_0(x), h + \delta \rangle \\
&\geq \frac{1}{2} \min_{\delta \in \varepsilon(x) B} (\|h_0(x)\|^2 - \|\delta\| \|h_0(x)\|) \\
&\geq \frac{1}{2} \|h_0(x)\| (\|h_0(x)\| - \varepsilon(x)) > 0
\end{aligned}$$

where the first inequality follows from (3.11), the second inequality follows from (3.8), the fifth inequality follows from (3.10), and the last inequality follows from (3.12). Hence $x \notin \mathcal{A}_0$, and it follows that $\mathcal{A}_0 \subset X_{stat}(\varepsilon(\cdot))$. Now applying Theorem 2.1 and Corollary 2.1, we immediately obtain the desired results. ■

Adding the "heavy ball" term [16] in Algorithm 3.1, we arrive at the following modification of the parallel GGPM. In neural network literature, methods of this type are usually referred to as backpropagation with momentum term [7, 10].

Algorithm 3.2 ***(Parallel GGPM with Momentum term).*** *Start with any $x^0 \in X$. Having x^i, compute x^{i+1} as follows :*
i) Parallelization: for each processor $l \in \{1, \dots, k\}$ do

$$z_l^{i,j+1} = P_X(z_l^{i,j} - \eta_i \tilde{g}_{i,j}^l), \quad j = 1, \dots, K_l,$$

where $z_l^{i,1} = x^i$.
(ii) Synchronization with momentum term:

$$x^{i+1} = P_X\left(x^i + \sum_{l=1}^{k}(z_l^{i,K_l+1} - x^i) + \beta_i(x^i - x^{\max\{i-s,0\}})\right),$$

where $s \in \mathcal{N}$ is some positive integer.

With respect to coefficients multiplying the momentum term, we assume that

$$\beta_i \geq 0, \ i = 0, 1, \dots, \quad \beta_i \to 0. \tag{3.13}$$

We also make the following assumption on the stepsizes (in addition to (3.1))

$$\limsup_{i \to \infty} \frac{\eta_{i-1}}{\eta_i} < +\infty. \tag{3.14}$$

We have the following

Theorem 3.2 *For every sequence $\{x^i\}$ generated by Algorithm 3.2, all the conclusions of Theorem 3.1 hold.*

Proof. We first define the following quantity

$$\mu_i = 2\beta_i M \sum_{t=\max\{i-s,0\}}^{i-1} \frac{\eta_{t-1}}{\eta_t}, \quad i = 1, 2, \dots, \quad \mu_0 = 0.$$

Note that by (3.13),(3.14),

$$\mu_i \geq 0, \ i = 0, 1, \dots, \quad \lim_{i \to \infty} \mu_i = 0.$$

Similarly to the case of Algorithm 3.1, it follows that every sequence $\{x^i\}$ generated by Algorithm 3.2 is a trajectory of the following process

$$x^{i+1} \in x^i - \eta_i \left(G(i, x^i) + \mu_i B \right), \quad i = 0, 1, \dots, \quad x^0 \in X,$$

where the mapping $G(\cdot, \cdot)$ is defined by (3.7). Now taking into account that $\mu_i \to 0$, we obtain

$$G_0(x) := \overline{\mathrm{l}}\mathrm{t}_{\substack{x' \to x \\ i \to \infty}} (G(i, x') + \mu_i B) \subset \partial f(x) + N_X(x) + \varepsilon(x) B.$$

The rest of the proof is analogous to that of Theorem 3.1, and is thus omitted. ∎

Remark 3.1. Theorems 3.1,3.2 generalize the results on convergence properties of the generalized gradient projection method obtained in [13, 3, 25].

4 Important Special Cases

In this section we consider the standard optimization problem (1.1), and establish stronger convergence properties of GGPM in a number of important special cases. These include convex and strongly convex problems, and problems with weak sharp minima [16, 1].

We start with the following lemma.

Lemma 4.1 *Let*

$$\varepsilon(x) \leq \max\{\bar{\varepsilon}, \theta r(x)\} \quad \forall x \in X,$$

where $\bar{\varepsilon} \geq 0, \ 1 > \theta \geq 0$. *Then*

$$X_{stat}(\varepsilon(\cdot)) \subset X_{stat}(\bar{\varepsilon}).$$

In particular, if $\bar{\varepsilon} = 0$, then

$$X_{stat}(\varepsilon(\cdot)) = X_{stat}.$$

Proof. Suppose $x \in X_{stat}(\varepsilon(x))$. Then, by (1.3) and the assumption of the lemma,

$$r(x) \leq \varepsilon(x) \leq \max\{\bar{\varepsilon}, \theta r(x)\}.$$

If $\theta r(x) \geq \bar{\varepsilon}$, then $r(x) \leq \theta r(x)$ and $1 > \theta \geq 0$ imply that $r(x) = 0$. Since $X_{stat}(0) \subset X_{stat}(\bar{\varepsilon})$, we have that $x \in X_{stat}(\bar{\varepsilon})$. If $\theta r(x) \leq \bar{\varepsilon}$, then $r(x) \leq \varepsilon(x) \leq \bar{\varepsilon}$, and hence $x \in X_{stat}(\bar{\varepsilon})$. ■

Let $d(\cdot, C)$ be the distance function to the set $C \subset \Re^n$, that is

$$d(x, C) = \inf_{y \in C} \|x - y\|.$$

Define $\bar{\varepsilon} = \sup_{x \in X} \varepsilon(x)$, and $D = \sup_{x,y \in X} \|x - y\|$. The following lemma relates the ε-stationary sets to the ϵ-optimal sets for the case when $f(\cdot)$ is convex.

Lemma 4.2 *Let $f(\cdot)$ be convex on X. Then*

$$X_{stat}(\varepsilon(x)) \subset X_{opt}\left(\varepsilon(x) d(x, X_{opt})\right).$$

In particular,

$$X_{stat}(\varepsilon(\cdot)) \subset X_{opt}(\bar{\varepsilon} D).$$

If, in addition, $f(\cdot)$ is differentiable and strongly convex on X with modulus l, and $X_{stat}(\bar{\varepsilon}) \subset intX$, then

$$X_{stat}(\bar{\varepsilon}) \subset X_{opt}(\bar{\varepsilon}^2/2l).$$

Proof. Let $x \in X_{stat}(\varepsilon(x))$. By definition of $X_{stat}(\varepsilon(\cdot))$, there exist $g \in \partial f(x)$, $h_1 \in N_X(x)$, and $h_2 \in \varepsilon(x)B$ such that $0 = g + h_1 + h_2$. Let $x^* = P_{X_{opt}}(x)$, i.e. x^* is the closest point to x in X_{opt}. By convexity of $f(\cdot)$, it follows that

$$\begin{aligned} f(x) - f(x^*) &\leq \langle -g, x^* - x\rangle = \langle h_1 + h_2, x^* - x\rangle \\ &\leq \langle h_2, x^* - x\rangle \leq \|h_2\| \|x^* - x\| \\ &\leq \varepsilon(x) d(x, X_{opt}), \end{aligned}$$

where the second inequality follows from definition of the normal cone. This establishes the first two assertions of the lemma.

For the last assertion, just note that (Lemma 1.4.3, [16]) for any $x \in X$

$$2l(f(x) - \min_{y \in X} f(y)) \leq \|\partial f(x)\|^2.$$

$\blacksquare$

Definition 4.1 *[1] We say that problem (1.1) is weakly sharp with parameter $\rho > 0$ if*

$$f(x) - \min_{y \in X} f(y) \geq \rho d(x, X_{opt}) \;\; \forall x \in X.$$

We have the following important corollary.

Corollary 4.1 *Let $f(\cdot)$ be convex on X. Assume that problem (1.1) is weakly sharp with parameter $\rho > 0$. Then if*

$$\varepsilon(x) \leq \max\{\nu, \theta r(x)\} \;\; \forall x \in X, \;\; 0 \leq \theta < 1, \nu < \rho,$$

it follows that

$$X_{stat}(\varepsilon(\cdot)) = X_{opt}.$$

Proof. Obviously, $X_{opt} \subset X_{stat}(\varepsilon(\cdot))$. Take any $x \in X_{stat}(\varepsilon(\cdot))$. By Lemmas 4.1,4.2, and our assumption, we have

$$x \in X_{stat}(\varepsilon(\cdot)) \subset X_{stat}(\nu) \subset X_{opt}(\nu d(x, X_{opt})).$$

Hence

$$\nu d(x, X_{opt}) \geq f(x) - \min_{y \in X} f(y) \geq \rho d(x, X_{opt}),$$

where the last inequality follows from Definition 4.1. Now $\nu < \rho$ implies that $d(x, X_{opt}) = 0$, $x \in X_{opt}$. $\blacksquare$

When in Algorithms 3.1,3.2 the parameter $K = 1$, those algorithms reduce to the following standard GGPM with the "heavy ball" term :

Algorithm 4.1 ***(GGPM with heavy ball term).*** *Start with any $x^0 \in X$. Having x^i, compute x^{i+1} as follows :*

$$x^{i+1} = P_X\left(x^i - \eta_i(g_i + \delta(x^i, \alpha_i, i)) + \beta_i(x^i - x^{\max\{i-s,0\}})\right)$$

$$g_i \in \partial f(x^i, \alpha_i),\ i = 0, 1, \dots ,$$

where parameters $\{\eta_i\}$, $\{\alpha_i\}$, $\{\beta_i\}$, and $s \in \mathcal{N}$ are the same as in Algorithms 3.1,3.2.

From Theorems 3.1,3.2, and Lemmas 4.1,4.2, we immediately get the following results :

Theorem 4.1 *Every sequence $\{x^i\}$ generated by Algorithm 4.1 possesses the following properties :*

1. *there exists an $f(\cdot)$-connected component $X_{stat}(\varepsilon(\cdot))^{(\gamma)}$ of $X_{stat}(\varepsilon(\cdot))$ such that*

   $$\bar{l}t_{i\to\infty}\{f(x^i)\} = f\left(\bar{l}t\{x^i\} \cap X_{stat}(\varepsilon(\cdot))^{(\gamma)}\right);$$

2. *every subsequence $\{x^{i_m}\}$ of $\{x^i\}$ satisfying (3.9) converges to $X_{stat}(\varepsilon(\cdot))^{(\gamma)}$;*

3. *if $f(\cdot)$ is convex, then $\{x^i\}$ converges to the set*

   $$X_{opt}\left(\varepsilon(x)d(x, X_{opt})\right) \subset X_{opt}(\tilde{\varepsilon}D);$$

4. *if, in addition, problem (1.1) is weakly sharp with parameter $\rho > 0$, and*

   $$\varepsilon(x) < \rho \quad \forall x \in X,$$

   then $\{x^i\}$ converges to X_{opt}.

Furthermore, Lemma 4.1 together with the last assertion of Theorem 2.1 yield the following result :

Theorem 4.2 *Let the exact asymptotic level of perturbations satisfy the following condition*

$$\varepsilon(x) \le \theta r(x) \quad \forall x \in X, \quad 0 \le \theta < 1,$$

(i.e. perturbations are relatively small). Suppose that

$$\textit{the set } f(X_{stat}) \textit{ is nowhere dense in } \Re\,.$$

Then every sequence $\{x^i\}$ generated by Algorithm 4.1 converges to X_{stat}.

Apllying Lemma 4.2 we can significantly sharpen the assertion 3 of Theorem 4.1 for the unconstrained strongly convex case.

Theorem 4.3 *Let $f(\cdot)$ be strongly convex with modulus $l > 0$, and $X_{stat}(\bar{\varepsilon}) \subset intX$. Then every sequence $\{x^i\}$ generated by Algorithm 4.1 converges to $X_{opt}(\bar{\varepsilon}^2/2l)$.*

5 Backpropagation With Noise

In this Section we apply the results of Section 3 to reveal some important properties of the backpropagation (BP) algorithm for training artificial neural networks [18, 7]. Due to numerous successful applications, a lot of empirical knowledge has been accumulated in the neural networks field. It is therefore important to provide rigorous mathematical foundation to neural networks theory and algorithms. Stochastic analysis of BP is given in [22]. The first deterministic convergence results (without data perturbations) were recently obtained in [11, 8]. An interesting new training method is proposed in [5].

In this Section we give a precise characterization to empirically observed stability of neural networks [19, 6]. We also discuss BP modifications with varying smoothing parameter.

We regard training artificial neural network as minimization of the following error function (see [9]) :

$$\min_{x \in X \subset \Re^n} f(x, \alpha) = \sum_{j=1}^{K} f_j(w, \theta, v, \tau, \alpha)$$

$$(5.1) \qquad f_j(w, \theta, v, \tau, \alpha) := \left(s \left(\sum_{i=1}^{h} s(\xi^j w^i - \theta^i) v^i - \tau \right) - t^j \right)^2 ,$$

where

h = fixed integer number of hidden units

K = fixed integer number of given training samples ξ^j in $\Re^m$

$t^j = 0$ or 1 target value for ξ^j, $j = 1, \dots, K$

τ = real number threshold of output unit

v^i = real number weights of outgoing arcs from hidden units, $i = 1, \ldots, h$

θ^i = real number thresholds of hidden units, $i = 1, \ldots, h$

w^i = m-vector weights of incoming arcs to hidden units, $i = 1, \ldots, h$

ξ^j = given m-dimensional vector samples, $j = 1, \ldots, K$

$s(\zeta) = 1$ if $\zeta > 0$ else 0

$s(\zeta) \cong \sigma(\zeta, \alpha) = \dfrac{1}{1 + e^{-\alpha\zeta}}$ for some $\alpha > 0$.

Here α is the smoothing parameter of the *sigmoid* approximation $\sigma(\zeta, \alpha)$ of the discontinuous step function $s(\zeta)$. Note that $f(x, \alpha)$ is precisely of the form (1.5). X is typically either $\Re^n$ or a set of simple box-constraints.

Each iteration of the ***serial online*** BP consists of a step in the direction of negative gradient $-\nabla f_j$ of a partial error function f_j associated with the j-th training example. Thus BP is a special case of Algorithm 3.1. Many other computationally important BP modifications, such as ***parallel*** BP [11, 14, 4], BP with ***momentum term*** [7], and BP with varying smoothing parameter [20] all fall within the framework of Section 3.

We now discuss stability issues in neural network training. It is quite common that for a sample ξ^j in the training set some of its attributes (i.e. the components of the m-dimensional vector) are computed (or supplied) with an error that we shall denote Δ_j. Obviously, this induces certain perturbation in values of the corresponding error function f_j and its gradient. We can then write

$$\tilde{f}_j(w, \theta, v, \tau, \alpha) := \left(\sigma \left(\sum_{i=1}^{h} \sigma((\xi^j + \Delta_j)w^i - \theta^i)v^i - \tau \right) - t^j \right)^2 ,$$

and

$$\nabla \tilde{f}_j(x, \alpha) = \nabla f_j(x, \alpha) + \delta_j(x, \alpha).$$

Note that it is fairly straightforward to estimate the dependence of δ_j on Δ_j. We can then introduce the exact asymptotic level of perturbations (3.6) by

$$\varepsilon(x) = \limsup_{\substack{z_j (\in X) \to x \\ i \to \infty}} \| \sum_{j \in Q} \delta_j(z_j, \alpha_i, i) \|,$$

where Q is the set of training examples with noise. If some upper bound on Δ_j, $j \in Q$ is known then the corresponding perturbations δ_j, $j \in Q$ and their asymptotic level $\varepsilon(\cdot)$ can be estimated. This in turn yields the guaranteed $\varepsilon(\cdot)$-stationarity of all the accumulation points of the BP iterates.

As another source of perturbations in the neural network training, we note the technique presented in [6]. To simplify the network topology and improve the network generalization properties, it is proposed in [6] to eliminate at the late stages of training the arcs with sufficiently small weights. The latter is equivalent to setting the corresponding weights to zero, and can also be treated as induced perturbations.

Another possible application of our analysis is devising new algorithms with varying smoothing parameter α. There exists empirical evidence that changing α during training can significantly speed up the learning process [20]. Unfortunately, most applications that take advantage of this idea usually employ some kind of heuristic to control the parameter. It will be interesting to develop a more rigorous algorithmic approach. This however is beyond the scope of this paper.

6 Concluding Remarks

We have analyzed convergence of the generalized gradient projection method in the presence of data perturbations. The parametric and "heavy ball" modifications, as well as the extension to the parallel method with additive objective function were also considered. It is shown that every trajectory of the algorithms is attracted to an $\varepsilon(\cdot)$-*stationary* set of the problem, where $\varepsilon(\cdot)$ depends on the magnitude of perturbations. In the convex case, the iterates are attracted to a certain ϵ-*optimal* set. Furthermore, if the problem has weak sharp minima, then convergence to the optimal set is established. Stability issues of the fundamental backpropagation algorithm for neural network training are discussed.

References

[1] J.V. Burke and M.C. Ferris. Weak sharp minima in mathematical programming. *SIAM Journal on Control and Optimization*, 31(5):1340–1359, 1993.

[2] F.H. Clarke. *Optimization and Nonsmooth Analysis.* John Wiley & Sons, New York, 1983.

[3] P. A. Dorofeev. On some properties of quasi-gradient method. *USSR Computational Mathematics and Mathematical Physics, Pergamon Press*, 25:181–189, 1985.

[4] D. Girard and H. Paugam-Moisy. Strategies for weight updating for parallel backpropagation. Technical Report 93-39, Laboratoire de l'Informatique du Parallélisme, Ecole Normale Supérieure de Lyon, 46, Allée d'Italie, 69364 Lyon Cedex 07, France, 1993.

[5] L. Grippo. A class of unconstrained minimization methods for neural network training. *Optimization Methods and Software*, 1994. to appear.

[6] B. Hassibi and D.G. Stork. Optimal brain sergeon. In G. Tesauro J.D. Cowan and J. Alspector, editors, *Advances in Neural Information Processing Systems 5*, San Francisco, CA, 1993. Morgan Kaufmann Publishers.

[7] T. Khanna. *Foundations of neural networks.* Addison–Wesley, New Jersey, 1989.

[8] Z.-Q. Luo and P. Tseng. Analysis of an approximate gradient projection method with applications to the backpropagation algorithm. *Optimization Methods and Software*, 1994. to appear.

[9] O.L. Mangasarian. Mathematical programming in neural networks. *ORSA Journal on Computing*, 5(4):349–360, 1993.

[10] O.L. Mangasarian and M.V. Solodov. Backpropagation convergence via deterministic nonmonotone perturbed minimization. In G. Tesauro J.D. Cowan and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, San Francisco, CA, 1994. Morgan Kaufmann Publishers.

[11] O.L. Mangasarian and M.V. Solodov. Serial and parallel backpropagation convergence via nonmonotone perturbed minimization. *Optimization Methods and Software*, 4:103–116, 1994.

[12] D.Q. Mayne and E. Polak. Nondifferentiable optimization via adaptive smoothing. *Journal of Optimization Theory and Applications*, 43(4):601–614, 1984.

[13] V. S. Mikhalevitch, A. M. Gupal, and V. I. Norkin. *Methods of nonconvex optimization.* Nauka, Moscow, 1987. (in Russian).

[14] H. Paugam-Moisy. On parallel algorithm for backpropagation by partitioning the training set. In *Neural Networks and Their Applications.* Proceedings of Fifth International Conference, Nimes, France, November 2-6, 1992.

[15] E. Polak. *Computational methods in optimization: A unified approach.* Academic Press, New York, New York, 1971.

[16] B.T. Polyak. *Introduction to Optimization.* Optimization Software, Inc., Publications Division, New York, 1987.

[17] N. Rouche, P. Habets, and M Laloy. *Stability Theory by Liapunov's Direct Method.* Springer–Verlag, New York, 1977.

[18] D.E. Rumelhart, G.E. Hinton, and R.J. Williams. Learning internal representations by error propagation. In D.E. Rumelhart and J.L. McClelland, editors, *Parallel Distributed Processing*, pages 318–362, Cambridge, Massachusetts, 1986. MIT Press.

[19] T.J. Sejnowski and C.R. Rosenberg. Paralel networks that learn to pronounce english text. *Complex Systems*, 1:145–168, 1987.

[20] A. Sperduti and A. Starita. Speed up learning and network optimization with extended backpropagation. *Neural Networks*, 6:365–383, 1993.

[21] J.N. Tsitsiklis, D.P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, AC-31(9):803–812, 1986.

[22] H. White. Some asymptotic results for learning in single hidden-layer feedforward network models. *Journal of the American Statistical Association*, 84(408):1003–1013, 1989.

[23] W.I. Zangwill. *Nonlinear Programming: A Unified Approach.* Prentice–Hall, Inc, Englewood Cliffs, New Jersey, 1969.

[24] S. K. Zavriev. Stochastic subgradient methods for Minmax problems. Izdatelstvo MGU, Moscow, 1984. (in Russian).

[25] S.K. Zavriev and A.G. Perevozchikov. Attraction of trajectories of finite-difference inclusions and stability of numerical methods of stochastic nonsmooth optimization. *Soviet Phys. Doklady*, 313:1373–1376, 1990.

[26] S.K. Zavriev and A.G. Perevozchikov. Direct Lyapunov's method in attraction analysis of finite-difference inclusions. *USSR Computational Mathematics and Mathematical Physics, Pergamon Press*, 30(1):22–32, 1990.