Computational Optimization and Applications 11, 23–35 (1998)

Incremental Gradient Algorithms with Stepsizes Bounded Away from Zero

M.V. SOLODOV solodov@impa.br
Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro, RJ 22460-320, Brazil

Received December 10, 1996; Revised July 10, 1997; Accepted August 8, 1997

Abstract. We consider the class of incremental gradient methods for minimizing a sum of continuously differentiable functions. An important novel feature of our analysis is that the stepsizes are kept bounded away from zero. We derive the first convergence results of any kind for this computationally important case. In particular, we show that a certain ε-approximate solution can be obtained and establish the linear dependence of ε on the stepsize limit. Incremental gradient methods are particularly well-suited for large neural network training problems where obtaining an approximate solution is typically sufficient and is often preferable to computing an exact solution. Thus, in the context of neural networks, the approach presented here is related to the principle of tolerant training. Our results justify numerous stepsize rules that were derived on the basis of extensive numerical experimentation but for which no theoretical analysis was previously available. In addition, convergence to (exact) stationary points is established when the gradient satisfies a certain growth property.

Keywords: incremental gradient methods, perturbed gradient methods, approximate solutions, backpropagation, neural network training

1. Introduction

We consider the problem

$$\min_{x\in\Re^n} f(x) := \sum_{j=1}^{K} f_j(x) \tag{1.1}$$

of minimizing a finite summation of continuously differentiable (partial) objective functions $f_j : \Re^n \to \Re,\ j = 1, \ldots, K$, where the number K is typically large. Our analysis is primarily motivated by machine learning (in particular, neural network) applications, where weights and thresholds of the network comprise the problem variable $x \in \Re^n$, K is the number of training samples, and $f_j(\cdot)$ represents the error associated with the jth sample, $j = 1, \ldots, K$ (see [11] for a detailed description).

In applications where K is large, the following *incremental* gradient algorithm (IGA) proved to be very useful (see also [2, Section 1.5.2]).

Algorithm 1.1 (IGA). *Choose any $x^0 \in \Re^n$. Having x^i, check a stopping criterion. If not satisfied, compute $x^{i+1} = T(x^i, \eta_i)$, where $T : \Re^n \times \Re_+ \to \Re^n$ is given by*

$$T(x, \eta) := x - \eta \sum_{j=1}^{K} \nabla f_j(z^j),$$

with

$$z^1 = x, \quad z^{j+1} = z^j - \eta \nabla f_j(z^j), \qquad j = 1, \ldots, K-1.$$

Algorithm 1.1 processes partial objective functions one at a time and immediately updates the variables (hence the name "incremental"). On the domain of large neural network training problems, this algorithm is known to be often superior to standard optimization techniques which process all the partial objective functions before adjusting the variables (see [5, 7] for a discussion of this issue). In particular, it is typically more effective than the standard gradient descent method given by

$$\tilde{T}(x, \eta) := x - \eta \nabla f(x) = x - \eta \sum_{j=1}^{K} \nabla f_j(x).$$

Moreover, in some applications, the cost of computing the full gradient $\nabla f(\cdot)$ at every iteration can be essentially cost prohibitive. Naturally, in that case more sophisticated techniques, such as conjugate gradient and quasi-Newton methods, are also inapplicable. Unfortunately, this often seems to be the case in machine learning. For many practical neural network systems, standard optimization methods require storage and/or computational cost which can become unmanageable even for a moderate network size, provided the training set (i.e., the number K) is large enough [17]. For problems of this class, incremental methods have to be used. In artificial intelligence literature IGA is usually referred to as on-line backpropagation training [16]. In addition to being faster, IGA has some other advantages over standard optimization methods when considered in the machine learning context. For example, it can be used in real-time on-chip operation [5].

Despite the popularity of incremental methods within the artificial intelligence community and their wide use in practice, until very recently there existed no rigorous convergence analysis for this class of algorithms. It is clear that IGA generates a sequence of iterates which need not be monotone with respect to the objective function values (this is easy to see because $-\nabla f_j$ need not be a direction of descent for the objective function $f(\cdot)$). This fact makes it difficult to apply standard Lyapunov-type techniques [14, 15] to the analysis of IGA. Thus a new approach had to be developed. This problem has recently attracted a lot of interest. The first deterministic results were obtained in [10, 13] (see also [2, 4]), where the stepsizes are chosen in order to satisfy the following condition

$$\sum_{i=0}^{\infty} \eta_i = \infty, \quad \sum_{i=0}^{\infty} \eta_i^2 < \infty. \tag{1.2}$$

In particular, it was shown that if (1.2) is satisfied then the sequence $\{f(x^i)\}$ converges and the sequence $\{\nabla f(x^i)\}$ converges to zero (in [13], furthermore, the use of a momentum

term and a parallel version of IGA were considered, while in [10] a constrained version of IGA was studied). Stochastic analysis under conditions similar to (1.2) can be found in [6, 22]. Error-stability properties of a very general class of algorithms, which includes IGA, are analyzed in [19]. In [3] a related least-squares incremental method is considered.

The results just cited, though significant, still left a certain gap between theoretical convergence analysis of incremental algorithms and computational practice. In particular, (1.2) implies that the stepsizes tend to zero in the limit, while many heuristic rules used by practitioners keep them bounded away from zero. It is therefore important to study the behavior of IGA when

$$\lim_{i\to\infty} \eta_i = \bar{\eta} > 0. \tag{1.3}$$

An example in [9] shows that in general, under the condition of (1.3), one cannot expect convergence to an exact solution even in the simple case when $f(\cdot)$ is given by a sum of two strongly convex quadratic (not identical) functions. Fortunately, in neural network applications one is typically *not* interested in computing an exact solution of (1.1) (more on this later). The results in this paper provide theoretical foundation for a number of heuristic stepsize rules that satisfy (1.3) but not (1.2) (see [5]).

We finally mention some interesting recent work on incremental algorithms. In [21] new adaptive stepsize rules are proposed and analyzed. These rules are much in the spirit of heuristics that are used in practice. However, the rules in [21] are designed in order to find an exact solution (stationary point) of the problem. This requirement usually drives the stepsizes to zero, unless some additional assumptions are satisfied (we emphasize that we do not make these assumptions for the main result of this paper). In [1] a hybrid algorithm is proposed which is aimed at accelerating (local) convergence of IGA-type methods. This new algorithm essentially works just like IGA when far from the eventual limit, and it gradually transforms into the steepest descent as the iterates approach a stationary point of the problem. Because asymptotically this algorithm behaves as the standard steepest descent method, it admits the use of a fixed stepsize.

While in some applications convergence to an exact solution (and fast *local* rate of convergence) may be important, it should be noted that in typical neural network problems it is not. In fact, it can be argued that training a neural network till an exact minimum of the error function is achieved leads to "memorizing" the training data and deterioration of the network generalization ability (incidentally, generalization on unseen data is the ultimate goal of training). This phenomenon is known as *overtraining* or *overfitting*. Fitting the data very accurately can be particularly harmful in the presence of noise. It is a widely accepted heuristic in machine learning that *tolerant training* should be employed to avoid overfitting [20]. Thus state-of-the-art neural network training systems almost always use some kind of *early stopping* criteria that terminate training *before* an exact solution to (1.1) is attained. For example, the use of tuning sets is popular [8]. We refer the reader to artificial intelligence literature for a discussion of overfitting and other related issues (see [20] and references therein).

Motivated by the above considerations, we adopt a slightly different point of view on the issues related to convergence of IGA-type techniques than that in [1, 10, 13, 21]. We

note that tolerant training permits certain errors in fitting the training data which, in the context of this paper, can be viewed as solving the problem (1.1) inexactly. Having this in mind, we are not concerned here with convergence of IGA iterates to an exact solution (or exact stationary point) of the minimization problem or with fast local convergence to such a point. The questions we ask are the following. (1) What are the properties of IGA when the stepsizes are bounded away from zero ? (2) Is it possible to compute a reasonably good approximate solution under this condition ? If so, how it can be characterized ?

In Section 2 we show that we can indeed compute a certain ε-approximate solution while keeping the stepsizes bounded away from zero. We say that a point $\bar{x} \in \Re^n$ is an ε- approximate solution of (1.1) if

$$\|\nabla f(\bar{x})\| \leq \varepsilon.$$

Of course, the above estimate is useful only if we can predict and control the value of ε as a function of algorithm parameters. In this paper, we establish that ε depends on the stepsize limit $\bar{\eta}$ at least linearly. Thus, by decreasing the stepsize to a sufficiently small (but bounded away from zero) value, it is possible to achieve any desired accuracy (Theorem 2.2). We point out that Theorem 2.2 is the first convergence result of any kind for incremental algorithms with stepsizes bounded away from zero. Our analysis is by virtue of characterizing IGA as a special perturbed gradient method (Proposition 2.1), and it makes use of some of the ideas employed in [18] where general perturbed feasible descent algorithms are studied. It is worth to point out that the main results of this paper cannot be obtained using the approach of [13]. As an aside, we show that under a certain additional assumption on the growth property on the gradients (similar to the one used in [21]), IGA converges to (exact) stationary points.

We briefly describe our notation. The usual inner product of two vectors $x \in \Re^n$, $y \in \Re^n$ is denoted by $\langle x, y \rangle$. The Euclidean 2-norm of $x \in \Re^n$ is given by $\|x\|^2 = \langle x, x \rangle$. For a differentiable function $\psi : \Re^n \to \Re$, $\nabla\psi$ will denote the n-dimensional vector of partial derivatives with respect to x. If a function $\psi(\cdot)$ has Lipschitz continuous partial derivatives on a set $D \subset \Re^n$ with some constant $L > 0$, that is

$$\|\nabla\psi(y) - \nabla\psi(x)\| \leq L\|y - x\|, \quad \forall x, y \in D,$$

we write $\psi(\cdot) \in C^1_L(D)$.

We finally state a well-known result that will be used later.

Lemma 1.1 ([15, p. 6]). *Let $\psi(\cdot) \in C^1_L(D)$, then*

$$|\psi(y) - \psi(x) - \langle \nabla\psi(x), y - x \rangle| \leq \frac{L}{2}\|y - x\|^2 \quad \forall x, y \in D.$$

2. Convergence analysis

In this section we establish the properties of incremental gradient algorithms when the stepsizes are bounded away from zero. In particular, we show that at least one accumulation

point of the sequence of iterates generated by IGA is an ε-approximate solution of the problem. Furthermore, we establish at least linear dependence of ε on the limiting value $\bar{\eta}$ of the sequence of stepsizes. It can be argued that computing an approximate solution falls within the tolerant training principle [20] of machine learning and, consequently, that having stepsizes bounded away from zero is, in some sense, sufficient for solving a neural network training problem.

We first show that IGA can be regarded as a perturbed gradient algorithm with a certain special structure.

Proposition 2.1. *The mapping T defined in Algorithm* 1.1 *satisfies the following properties*

$$T(x, \eta) = x - \eta \nabla f(x) + \eta^2 \delta(x, \eta),$$

where

$$\|\delta(x, \eta)\| \leq B$$

for some constant $B > 0$ (independent of η), provided $f_j(\cdot) \in C^1_L$, and $\|\nabla f_j(x)\| \leq M$, $j = 1, \ldots, K$ for all x and some $M > 0$.

Proof: By the construction of Algorithm 1.1, we have

$$\begin{aligned} T(x, \eta) &= x - \eta \sum_{j=1}^{K} \nabla f_j(z^j) \\ &= x - \eta \left(\sum_{j=1}^{K} (\nabla f_j(z^j) - \nabla f_j(x) + \nabla f_j(x)) \right) \\ &= x - \eta \left(\sum_{j=1}^{K} \nabla f_j(x) + \sum_{j=1}^{K} (\nabla f_j(z^j) - \nabla f_j(x)) \right) \\ &= x - \eta \nabla f(x) + \eta^2 \delta(x, \eta), \end{aligned} \tag{2.1}$$

where

$$\delta(x, \eta) := \eta^{-1} \sum_{j=1}^{K} (\nabla f_j(x) - \nabla f_j(z^j)). \tag{2.2}$$

It is easy to see that δ is a function of both x and η because so are z^j, $j = 2, \ldots, K$. Furthermore, $\delta(\cdot, \cdot)$ is continuous in both variables.

Define

$$\delta_j := \|\nabla f_j(z^j) - \nabla f_j(x)\|, \quad j = 1, \ldots, K. \tag{2.3}$$

First, note that $\delta_1 = 0$ because $z^1 = x$. We next show, by induction, that

$$\delta_j \leq \eta L \sum_{t=1}^{j-1} (1 + L\eta)^{j-1-t} \|\nabla f_t(x)\|, \quad j = 2, \ldots, K. \tag{2.4}$$

For $j = 2$, we have

$$\begin{aligned} \delta_2 &= \|\nabla f_2(z^2) - \nabla f_2(x)\| \\ &\leq L\|z^2 - z^1\| \\ &= \eta L \|\nabla f_1(x)\|, \end{aligned}$$

where the inequality follows from $x = z^1$ and the Lipschitz continuity of $\nabla f_2(\cdot)$. Hence, (2.4) is valid for $j = 2$. Suppose that (2.4) holds for $j = 2, \ldots, m$ where $m < K$. By the triangle inequality, from (2.3) it follows that

$$\|\nabla f_j(z^j)\| \leq \|\nabla f_j(x)\| + \delta_j.$$

Combining the latter inequality with (2.4) (for $j \leq m$) we obtain

$$\|\nabla f_j(z^j)\| \leq \|\nabla f_j(x)\| + \eta L \sum_{t=1}^{j-1} (1 + L\eta)^{j-1-t} \|\nabla f_t(x)\|, \quad j = 2, \ldots, m. \tag{2.5}$$

Now consider $j = m + 1$. We have

$$\begin{aligned} \delta_{m+1} &= \|\nabla f_{m+1}(z^{m+1}) - \nabla f_{m+1}(x)\| \\ &\leq L\|z^{m+1} - x\| \\ &= L\left\| \sum_{t=1}^{m} (z^{t+1} - z^t) \right\| \\ &\leq L \sum_{t=1}^{m} \|z^{t+1} - z^t\| \\ &= \eta L \sum_{t=1}^{m} \|\nabla f_t(z^t)\|, \end{aligned}$$

where the first inequality follows from the Lipschitz continuity of $\nabla f_{m+1}(\cdot)$ and the second from the triangle inequality. Combining the latter relation with (2.5), we further obtain

$$\delta_{m+1} \leq \eta L \sum_{t=1}^{m} \left(\|\nabla f_t(x)\| + \eta L \sum_{s=1}^{t-1} (1 + L\eta)^{t-1-s} \|\nabla f_s(x)\| \right).$$

Combining the corresponding terms gives

$$\delta_{m+1} \leq \eta L \sum_{t=1}^{m} (1 + L\eta)^{m-t} \|\nabla f_t(x)\|.$$

The latter inequality is precisely (2.4) with $j = m + 1$. The induction step is complete and (2.4) is proven.

By (2.2)–(2.4) and the triangle inequality, it follows that

$$\begin{aligned}\|\delta(x,\eta)\| &\le \eta^{-1}\sum_{j=2}^{K}\delta_j \\ &\le L\sum_{j=2}^{K}\sum_{t=1}^{j-1}(1+L\eta)^{j-1-t}\|\nabla f_t(x)\| \\ &\le c_1\sum_{j=1}^{K}\|\nabla f_j(x)\| \\ &\le c_1 K M =: B, \end{aligned} \tag{2.6}$$

for some constant $c_1 > 0$ (c_1 can be taken independent of η). □

We next prove a convergence result for a class of perturbed gradient algorithms with a special structure given in Proposition 2.1. The principal application of this result is to establish convergence properties of IGA with stepsize bounded away from zero.

Theorem 2.1. *Let $f(\cdot) \in C^1_L(D)$, where D is a bounded set in $\Re^n$. Let $\{x^i\} \subset D$ be a sequence generated by $x^{i+1} = T(x^i, \eta_i)$ where*

$$T(x,\eta) = x - \eta\nabla f(x) + \eta^2\delta(x,\eta).$$

Suppose

$$\lim_{i\to\infty}\eta_i = \bar\eta > 0 \quad \textit{and} \quad \|\delta(x^i,\eta_i)\| \le B,$$

where $\eta_i \in (\theta, 2/L - \theta)$ with $\theta \in (0, 1/L]$, and $B > 0$. Then there exist a constant $C > 0$ (independent of $\bar\eta$) and an accumulation point $\bar x$ of the sequence $\{x^i\}$ such that

$$\|\nabla f(\bar x)\| \le C\bar\eta. \tag{2.7}$$

Furthermore, if the sequence $\{f(x^i)\}$ converges then every accumulation point $\bar x$ of the sequence $\{x^i\}$ satisfies (2.7).

Proof: By Lemma 1.1, we have

$$\begin{aligned} f(x) - f(T(x,\eta)) &\ge -\langle\nabla f(x), T(x,\eta) - x\rangle - \frac{L}{2}\|T(x,\eta) - x\|^2 \\ &= \eta\langle\nabla f(x), \nabla f(x) - \eta\delta(x,\eta)\rangle - \frac{L}{2}\eta^2\|\nabla f(x) + \eta\delta(x,\eta)\|^2 \\ &\ge \eta\left(1 - \frac{L}{2}\eta\right)\|\nabla f(x)\|^2 - \eta^2(1+L\eta)B\|\nabla f(x)\| - \frac{L}{2}\eta^4 B^2, \end{aligned}$$

where the last relation follows from the Cauchy-Schwarz inequality and the fact that $\|\delta(x^i, \eta_i)\| \leq B$. Define

$$\varphi(x, \eta) := \left(1 - \frac{L}{2}\eta\right) \|\nabla f(x)\|^2 - \eta(1 + L\eta)B\|\nabla f(x)\| - \frac{L}{2}\eta^3 B^2. \tag{2.8}$$

Note that $\varphi(\cdot, \cdot)$ is continuous in both variables. With definition (2.8), we have

$$f(x) - f(T(x, \eta)) \geq \eta\, \varphi(x, \eta). \tag{2.9}$$

Let $\{x^i\}$ be any sequence generated by the process under consideration. Suppose

$$\liminf_{i\to\infty} \varphi(x^i, \eta_i) > 0.$$

Then there exist an index i_1 and a number $\epsilon > 0$ such that $\varphi(x^i, \eta_i) \geq \epsilon$ for all $i \geq i_1$. Since $\eta_i \to \bar{\eta}$, it follows that for some i sufficiently large, say $i \geq i_2$, we also have $\eta_i \geq \bar{\eta}/2$. Then for all $i \geq i_3 := \max\{i_1, i_2\}$ it follows from (2.9) that

$$f(x^i) - f(x^{i+1}) \geq \bar{\eta}\epsilon/2.$$

Hence, for any $i > i_3$, we have

$$\begin{aligned} f\left(x^{i_3}\right) - f(x^i) &= \sum_{t=i_3}^{i-1} (f(x^t) - f(x^{t+1})) \\ &\geq \sum_{t=i_3}^{i-1} \bar{\eta}\epsilon/2 \\ &= (i - i_3)\bar{\eta}\epsilon/2. \end{aligned}$$

Letting $i \to \infty$ in the above relation we have that $\{f(x^i)\} \to -\infty$ which contradicts the fact that $f(\cdot)$ is continuous and D is bounded. It follows that

$$\liminf_{i\to\infty} \varphi(x^i, \eta_i) \leq 0.$$

Thus, by boundedness of the sequence $\{x^i\}$ and continuity of $\varphi(\cdot, \cdot)$, there exists an accumulation point $\bar{x}$ of $\{x^i\}$ such that

$$\varphi(\bar{x}, \bar{\eta}) \leq 0. \tag{2.10}$$

Denote $u := \|\nabla f(\bar{x})\|$. Then (2.10) gives the following quadratic inequality in u (via (2.8))

$$\left(1 - \frac{L}{2}\bar{\eta}\right) u^2 - \bar{\eta}(1 + L\bar{\eta})Bu - \frac{L}{2}\bar{\eta}^3 B^2 \leq 0.$$

Note that $1 - L\bar{\eta}/2 > 0$. Resolving this inequality yields

$$u \leq \frac{\bar{\eta}}{2 - L\bar{\eta}} \left((1 + L\bar{\eta})B + \sqrt{(1 + L\bar{\eta})^2 B^2 + 2L\bar{\eta}B^2(1 - L\bar{\eta}/2)} \right).$$

Therefore,

$$\|\nabla f(\bar{x})\| \leq C\bar{\eta},$$

for some constant $C > 0$ (C can be taken independent of $\bar{\eta}$).

The last assertion of the theorem follows from the observation that if the sequence $\{f(x^i)\}$ converges, the left-hand side of (2.9) tends to zero. This, in turn, implies that

$$\limsup_{i \to \infty} \varphi(x^i, \eta_i) \leq 0.$$

Hence, (2.10) and the subsequent analysis hold for *every* accumulation point $\bar{x}$ of the sequence $\{x^i\}$. The proof is complete. □

Remark 2.1. Instead of $\|\delta(x^i, \eta_i)\| \leq B$ we can more generally consider the assumption $\|\delta(x^i, \eta_i)\| \leq B_1 + B_2 \|\nabla f(x^i)\|$ with much of the above analysis still applying (a quadratic inequality will have to be replaced by a higher order one). However, we prefer to keep focus on IGA, so we will not pursue this extension.

We are now ready to state our convergence results for IGA. A remark about the assumptions of Theorem 2.2 below is in order. In this theorem, we explicitly assume that the sequence $\{x^i\}$ generated by IGA is bounded. We note that this is not restrictive since it can be shown (see [21]) that the iterates are contained in some set $\{x \mid f(x) \leq \rho_1\} + \{x \mid \|x\| \leq \rho_2\}$ which is bounded if the level set $\{x \mid f(x) \leq \rho_1\}$ is bounded for some $\rho_1 > f(x^0)$, as is the typical case with neural network training (see [10, Section 3]). It is also easy to see that the gradient of the neural network training function is Lipschitz continuous and bounded on any bounded set.

Theorem 2.2. *Let $\{x^i\}$ be any sequence generated by IGA such that all iterates, including "minor" iterates z^j's, belong to some bounded set D in $\Re^n$. Suppose $\eta_i \in (\theta, 2/L - \theta)$, where $\theta \in (0, 1/L]$, and*

$$\lim_{i \to \infty} \eta_i = \bar{\eta} > 0.$$

Let $f(\cdot) \in C_L^1(D)$, $f_j(\cdot) \in C_L^1(D)$, $j = 1, \ldots, K$, and $\|\nabla f_j(x)\| \leq M$, $j = 1, \ldots, K$ for all $x \in D$ and some $M > 0$. Then there exist a constant $C > 0$ (independent of $\bar{\eta}$) and an accumulation point $\bar{x}$ of the sequence $\{x^i\}$ such that

$$\|\nabla f(\bar{x})\| \leq C\bar{\eta}.$$

Furthermore, if the sequence $\{f(x^i)\}$ *converges then every accumulation point* $\bar{x}$ *of the sequence* $\{x^i\}$ *has the above property.*

Proof: The result follows from combining Proposition 2.1 and Theorem 2.1. □

The analysis presented here can be applied to a variety of modifications and extensions of Algorithm 1.1. For example, using the approach of [18], we could treat the projection version of IGA described in [10, 21]. At the expense of introducing considerably more notation and some technical details, we could also consider the parallel and momentum term modifications given in [13, 21], as well as algorithms with noisy data along the lines of [19].

Although specific stepsize rules are not a subject of this paper, we shall make a few remarks concerning this issue. In practice, one usually starts with a fixed intuitively reasonable stepsize value and uses it as long as the algorithm makes sufficient progress according to some chosen criterion. When sufficient progress is not being made, and if the current approximate solution is not satisfactory, the stepsize is decreased. From (2.7) we can see that decreasing the stepsize will, indeed, yield a better approximate solution (in some sense). As a practical matter, we would suggest using a stepsize rule similar to that proposed in [21] with a slight modification consisting of imposing a lower bound on the stepsize. On one hand, this modification will prevent the stepsize from becoming too small and, on the other hand, it may also help to avoid overfitting by computing an approximate rather than an exact solution. This approach would be very close to heuristics used in practice.

As an alternative to decreasing the stepsize, one could dynamically aggregate partial objective functions into (larger) groups (as mentioned in [12]) which is also likely to reduce the right-hand side of (2.7).

As a side result, we now establish convergence of IGA to exact stationary points of (1.1) under a growth condition on the gradients very similar to that used in [21]. We repeat, however, that convergence to exact solutions is not among primary concerns of this paper.

Theorem 2.3. *Let* $f(\cdot) \in C^1_L(\Re^n)$ *and let* $\{x^i\}$ *be any sequence generated by IGA such that for some index* i_0 *the level set* $\{x \in \Re^n \mid f(x) \leq f(x^{i_0})\}$ *is contained in the set*

$$P := \left\{ x \in \Re^n \;\middle|\; \sum_{j=1}^{K} \|\nabla f_j(x)\| \leq c_2 \|\nabla f(x)\| \right\} \quad \text{for some } c_2 > 0.$$

Let η_i *satisfy* (1.3) *and, in addition,*

$$\eta_i \leq \min\{1/(3L), 1/(2c_1c_2)\}, \quad \forall i \geq i_0,$$

where c_1 *is the constant from* (2.6).

Then the sequence $\{f(x^i)\}$ *converges, the sequence* $\{\nabla f(x^i)\}$ *converges to zero, and every accumulation point of the sequence* $\{x^i\}$ *is a stationary point of* (1.1).

Proof: Whenever $x \in P$, it follows from (2.6) that

$$\begin{aligned}\|\delta(x,\eta)\| &\le c_1 \sum_{j=1}^{K} \|\nabla f_j(x)\| \\ &\le c_1 c_2 \|\nabla f(x)\|. \end{aligned} \tag{2.11}$$

We further obtain

$$\begin{aligned} f(x) - f(T(x,\eta)) &\ge -\langle \nabla f(x), T(x,\eta) - x\rangle - \frac{L}{2}\|T(x,\eta) - x\|^2 \\ &= \eta\langle \nabla f(x), \nabla f(x) - \eta\delta(x,\eta)\rangle - \frac{L}{2}\eta^2\|\nabla f(x) + \eta\delta(x,\eta)\|^2 \\ &\ge \eta\left(1 - \frac{L}{2}\eta\right)\|\nabla f(x)\|^2 - \eta^2(1 + L\eta)c_1c_2\|\nabla f(x)\|^2 \\ &\quad - \frac{L}{2}(\eta^2 c_1 c_2)^2\|\nabla f(x)\|^2 \\ &= \eta(1 - L\eta/2 - \eta(1 + L\eta)c_1c_2 - \eta(\eta c_1 c_2)^2 L/2)\|\nabla f(x)\|^2. \end{aligned}$$

For $i = i_0$, by the choice of η_i, it follows that $L\eta_i/2 \le 1/6$, $1 + L\eta_i \le 4/3$ and $\eta_i c_1 c_3 \le 1/2$. Hence,

$$\begin{aligned} f(x^{i_0}) - f(x^{i_0+1}) &\ge \eta_{i_0}\big(1 - 1/6 - 4/6 - 1/24\big)\|\nabla f(x^{i_0})\|^2 \\ &= \frac{\eta_{i_0}}{8}\|\nabla f(x^{i_0})\|^2 \ge 0. \end{aligned}$$

Since $f(x^{i_0+1}) \le f(x^{i_0})$, it follows that $x^{i_0+1} \in \{x \in \Re^n \mid f(x) \le f(x^{i_0})\}$. Then, by assumption, we also have that $x^{i_0+1} \in P$. Using (2.11) and repeating the preceding argument for $i = i_0 + 1, i_0 + 2, \ldots$, we have $x^i \in \{x \in \Re^n \mid f(x) \le f(x^{i_0})\} \subset P$ for all $i \ge i_0$. Therefore, for all $i \ge i_0$,

$$f(x^i) - f(x^{i+1}) \ge \frac{\eta_i}{8}\|\nabla f(x^i)\|^2.$$

By (1.3), for all i sufficiently large, say $i \ge i_1$, we have $\eta_i \ge \bar{\eta}/2$. Then for $i \ge i_2 := \max\{i_0, i_2\}$ we obtain

$$f(x^i) - f(x^{i+1}) \ge \frac{\bar{\eta}}{16}\|\nabla f(x^i)\|^2.$$

Hence, the sequence $\{f(x^i)\}$ is nonincreasing (at least starting with some index i_2). Since it is bounded, it converges. Then the last relation also implies that the sequence $\{\nabla f(x^i)\}$ converges to zero. Therefore, by continuity of $\nabla f(\cdot)$, for every accumulation point $\bar{x}$ of the sequence $\{x^i\}$ it follows that $\nabla f(\bar{x}) = 0$. □

3. Concluding remarks

The first convergence results for the class of incremental gradient algorithms with stepsizes bounded away from zero were presented. In particular, it was shown that a certain ε-approximate solution can be obtained. Furthermore, the linear dependence of ε on the stepsize values was established. Applications on neural network training were also discussed. For example, solving the original problem inexactly conforms to the widely accepted tolerant training principle in machine learning.

Acknowledgments

This research is supported in part by CNPq grant number 300734/95-6. I am grateful to the two anonymous referees for constructive review and helpful suggestions which led to improvements in the paper.

References

1. D.P. Bertsekas, "A new class of incremental gradient methods for least squares problems," SIAM J. on Optimization, vol. 7, pp. 913–926, 1997.
2. D.P. Bertsekas, Nonlinear Programming, Athena Scientific: Belmont, MA, 1995.
3. D.P. Bertsekas, "Incremental least squares methods and the extended Kalman filter," SIAM Journal on Optimization, vol. 6, pp. 807–822, 1996.
4. D.P. Bertsekas and J.N. Tsitsiklis, Neuro-Dynamic Programming, Athena Scientific: Belmont, MA, 1996.
5. A. Cichocki and R. Unbehauen, Neural Networks for Optimization and Signal Processing, John Wiley & Sons: New York, 1994.
6. A.A. Gaivoronski, "Convergence properties of backpropagation for neural networks via theory of stochastic gradient methods. Part I," Optimization Methods and Software, vol. 4, pp. 117–134, 1994.
7. T. Khanna, Foundations of Neural Networks, Addison-Wesley: NJ, 1989.
8. K. Lang, A. Waibel, and G. Hinton, "A time-delay neural network architecture for isolated word recognition," Neural Networks, vol. 3, pp. 23–43, 1990.
9. Z.-Q. Luo, "On the convergence of the LMS algorithm with adaptive learning rate for linear feedforward networks," Neural Computation, vol. 3, pp. 226–245, 1991.
10. Z.-Q. Luo and P. Tseng, "Analysis of an approximate gradient projection method with applications to the backpropagation algorithm," Optimization Methods and Software, vol. 4, pp. 85–101, 1994.
11. O.L. Mangasarian, "Mathematical programming in neural networks," ORSA Journal on Computing, vol. 5, no. 4, pp. 349–360, 1993.
12. O.L. Mangasarian and M.V. Solodov, "Backpropagation convergence via deterministic nonmonotone perturbed minimization," in Advances in Neural Information Processing Systems 6, G. Tesauro, J.D. Cowan, and J. Alspector (Eds.), Morgan Kaufmann: San Francisco, CA, 1994, pp. 383–390.
13. O.L. Mangasarian and M.V. Solodov, "Serial and parallel backpropagation convergence via nonmonotone perturbed minimization," Optimization Methods and Software, vol. 4, pp. 103–116, 1994.
14. E. Polak, Computational Methods in Optimization: A Unified Approach, Academic Press: New York, 1971.
15. B.T. Polyak, Introduction to Optimization, Optimization Software, Inc. Publications Division: New York, 1987.
16. D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning internal representations by error propagation," in Parallel Distributed Processing, D.E. Rumelhart and J.L. McClelland (Eds.), MIT Press: Cambridge, MA, 1986, pp. 318–362.
17. S. Shah, F. Palmieri, and M. Datum, "Optimal filtering algorithms for fast learning in feedforward neural networks," Neural Networks, vol. 5, pp. 779–787, 1992.

18. M.V. Solodov, “Convergence analysis of perturbed feasible descent methods,” Journal of Optimization Theory and Applications, vol. 93, no. 2, pp. 337–353, May 1997.
19. M.V. Solodov and S.K. Zavriev, “Error-stabilty properties of generalized gradient-type algorithms,” Technical Report Mathematical Programming 94-05, Computer Science Department, University of Wisconsin, 1210 West Dayton Street, Madison, Wisconsin 53706, USA, June 1994. Journal of Optimization Theory and Applications, vol. 98, no. 3, September 1998.
20. W.N. Street and O.L. Mangasarian, “Improved generalization via tolerant training,” Technical Report 95-11, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin 53706, USA, July 1995. Journal of Optimization Theory and Applications, vol. 96, pp. 259–279, 1998.
21. P. Tseng, “Incremental gradient(-projection) method with momentum term and adaptive stepsize rule,” SIAM J. on Optimization, vol. 8, pp. 506–531, 1998.
22. H. White, “Some asymptotic results for learning in single hidden-layer feedforward network models,” Journal of the American Statistical Association, vol. 84, no. 408, pp. 1003–1013, 1989.