Optimization on Solution Sets of Common Fixed Point Problems

Alexander J. Zaslavski

November 15, 2021

We study optimization on solution sets of common fixed point problems. Our goal is to obtain a good approximate solution of the problem in the presence of computational errors. We show that an algorithm generates a good approximate solution, if the sequence of computational errors is bounded from above by a small constant. Moreover, if we know computational errors for our algorithm, we find out what an approximate solution can be obtained and how many iterates one needs for this. The talk is based on the recent book titled "Optimization on Solution Sets of Common Fixed Point Problems".

For every $z \in R^1$ denote by $\lfloor z \rfloor$ the largest integer which does not exceed z:

$$\lfloor z \rfloor = \max\{i \in R^1 : i \text{ is an integer and } i \le z\}.$$

For every nonempty set D, every function $f : D \to R^1$ and every nonempty set $C \subset D$ we set

$$\inf(f, C) = \inf\{f(x) : \ x \in C\}$$

and

$$\text{argmin}(f, C) = \text{argmin}\{f(x) : \ x \in C\}$$

$$= \{x \in C : \ f(x) = \inf(f, C)\}.$$

Let X be a Hilbert space equipped with an inner product denoted by $\langle \cdot, \cdot \rangle$ which induces a complete norm $\| \cdot \|$. For each $x \in X$ and each $r > 0$ set

$$B_X(x, r) = \{y \in X : \ \|x - y\| \le r\}$$

and set

$$B(x, r) = B_X(x, r)$$

if the space X is understood.

For each $x \in X$ and each nonempty set $E \subset X$ set

$$d(x, E) = \inf\{\|x - y\| : \ y \in E\}.$$

For each nonempty open convex set $U \subset X$ and each convex function $f : U \to R^1$, for every $x \in U$ set

$$\partial f(x) = \{l \in X :$$

$$f(y) - f(x) \ge \langle l, y - x \rangle \text{ for all } y \in U\}$$

which is called the subdifferential of the function f at the point x. Denote by Card(A) the cardinality of a set A. We suppose that the sum over an empty set is zero.

We study the subgradient algorithm and its modifications for minimization of convex functions, under the presence of computational errors.

Usually the problem, studied in the literature, is described by an objective function and a set of feasible points. For this algorithm each iteration consists of two steps. The first step is a calculation of a subgradient of the objective function while in the second one we calculate a projection on the feasible set. In each of these two steps there is a computational error. In general, these two computational errors are different.

In our recent research (see A. J. Zaslavski, Numerical Optimization with Computational Errors, Springer Optimization and Its Applications, Springer, 2016 (AZ16a), A. J. Zaslavski, Convex Optimization with Computational Errors, Springer Optimization and Its Applications, Springer, 2020 (AZ0a), A. J. Zaslavski, The Projected Subgradient Algorithm in Convex Optimization, SpringerBriefs in Optimization, 2020 (AZ20b)) we show that the algorithm generate a good approximate solution, if all the computational errors are bounded from above by a small positive constant. Moreover, if we know computational errors for the two steps of our algorithm, we find out what an approximate solution can be obtained and how many iterates one needs for this. It should be mentioned that in AZ16a, AZ20a analogous results were obtained for many others important algorithms in optimization and in the game theory.

We use the subgradient projection algorithm for constrained minimization problems in Hilbert spaces equipped with an inner product denoted by $\langle \cdot, \cdot \rangle$ which induces a complete norm $\| \cdot \|$.

Let C be a nonempty closed convex subset of X, U be an open convex subset of X such that $C \subset U$ and let $f : U \to R^1$ be a convex function.

Suppose that there exist $L > 0$, $M_0 > 0$ such that

$$C \subset B_X(0, M_0),$$

$$|f(x) - f(y)| \leq L\|x - y\| \text{ for all } x, y \in U.$$

It is not difficult to see that for each $x \in U$,

$$\emptyset \neq \partial f(x) \subset B_X(0, L).$$

For every nonempty closed convex set $D \subset X$ and every $x \in X$ there is a unique point $P_D(x) \in D$ satisfying

$$\|x - P_D(x)\| = \inf\{\|x - y\| : \ y \in D\}.$$

We consider the minimization problem

$$f(z) \to \min, \ z \in C.$$

Suppose that $\{a_k\}_{k=0}^{\infty} \subset (0, \infty)$. Let us describe our algorithm.

Subgradient projection algorithm

Initialization: select an arbitrary $x_0 \in U$.

Iterative step: given a current iteration vector $x_t \in U$ calculate

$$\xi_t \in \partial f(x_t)$$

and the next iteration vector $x_{t+1} = P_C(x_t - a_t \xi_t)$.

In AZ16a we study this algorithm under the presence of computational errors. We suppose that $\delta \in (0, 1]$ is a computational error produced by our computer system, and study the following algorithm.

Subgradient projection algorithm with computational errors

Initialization: select an arbitrary $x_0 \in U$.

Iterative step: given a current iteration vector $x_t \in U$ calculate

$$\xi_t \in \partial f(x_t) + B_X(0, \delta)$$

and the next iteration vector $x_{t+1} \in U$ such that

$$\|x_{t+1} - P_C(x_t - a_t \xi_t)\| \leq \delta.$$

In Chapter 2 of AZ20a we consider more complicated, but more realistic, version of this algorithm. Clearly, for the algorithm each iteration consists of two steps. The first step is a calculation of a subgradient of the objective function f while in the second one we calculate a projection on the set C. In each of these two steps there is a computational error produced by our computer system. In general, these two computational errors are different. This fact is taken into account in the following projection algorithm studied in Chapter 2 of AZ20a.

Suppose that $\{a_k\}_{k=0}^{\infty} \subset (0, \infty)$ and $\delta_f, \delta_C \in (0, 1]$.

Initialization: select an arbitrary $x_0 \in U$.

Iterative step: given a current iteration vector $x_t \in U$ calculate

$$\xi_t \in \partial f(x_t) + B_X(0, \delta_f)$$

and the next iteration vector $x_{t+1} \in U$ such that

$$\|x_{t+1} - P_C(x_t - a_t\xi_t)\| \leq \delta_C.$$

Note that in practice for some problems the set C is simple but the function f is complicated. In this case δ_C is essentially smaller than δ_f. On the other hand, there are cases when f is simple but the set C is complicated and therefore δ_f is much smaller than δ_C.

In Chapter 2 of AZ20a we proved the following result.

Theorem 1 *Let* $\delta_f, \delta_C \in (0, 1]$, $\{a_k\}_{k=0}^{\infty} \subset (0, \infty)$ *and let*

$$x_* \in C$$

satisfy

$$f(x_*) \le f(x) \text{ for all } x \in C.$$

Assume that $\{x_t\}_{t=0}^{\infty} \subset U$, $\{\xi_t\}_{t=0}^{\infty} \subset X$,

$$\|x_0\| \le M_0 + 1$$

and that for each integer $t \ge 0$,

$$\xi_t \in \partial f(x_t) + B_X(0, \delta_f)$$

and

$$\|x_{t+1} - P_C(x_t - a_t \xi_t)\| \le \delta_C.$$

Then for each natural number T,

$$\sum_{t=0}^{T} a_t(f(x_t) - f(x_*))$$

$$\leq 2^{-1}\|x_* - x_0\|^2 + \delta_C(T+1)(4M_0+1)$$

$$+\delta_f(2M_0+1)\sum_{t=0}^{T} a_t + 2^{-1}(L+1)^2 \sum_{t=0}^{T} a_t^2.$$

Moreover, for each natural number T,

$$f((\sum_{t=0}^{T} a_t)^{-1} \sum_{t=0}^{T} a_t x_t) - f(x_*),$$

$$\min\{f(x_t) : \ t = 0, \dots, T\} - f(x_*)$$

$$\leq 2^{-1}(\sum_{t=0}^{T} a_t)^{-1} \|x_* - x_0\|^2$$

$$+(\sum_{t=0}^{T} a_t)^{-1} \delta_C (T+1)(4M_0 + 1)$$

$$+\delta_f(2M_0 + 1) + 2^{-1}(\sum_{t=0}^{T} a_t)^{-1}(L+1)^2 \sum_{t=0}^{T} a_t^2.$$

We are interested in an optimal choice of a_t, $t = 0, 1, \ldots$. Let T be a natural number and $A_T = \sum_{t=0}^{T} a_t$ be given. It is shown in AZ20a that the best choice is $a_t = (T+1)^{-1} A_T$, $t = 0, \ldots, T$.

Let T be a natural number and $a_t = a > 0$, $t = 0, \ldots, T$. It is shown in AZ20a that the best choice of a is

$$a = (2\delta_C(4M_0 + 1))^{1/2}(L+1)^{-1}.$$

Now we can think about the best choice of T. It is not difficult to see that it should be at the same order as $\lfloor \delta_C^{-1} \rfloor$.

In AZ20b we generalize the results obtained in AZ20a for the subgradient projection algorithm in the case when instead of the projection operator on C it is used a quasi-nonexpansive retraction on C.

Fixed point subgradient algorithms

In our analysis it was used the fact that we can calculate a projection operator P_C with small computational errors. Of course, this is possible only when the C is simple, like a simplex or a half-space. In practice the situation is more complicated. In real world applications the set C is an intersection of a finite family of simple closed convex sets C_i, $i = 1, \dots, C_m$. To calculate the mapping P_C is impossible and instead of it one has to work with projections P_{C_i}, $i = 1, \dots, m$ on the simple sets $C_1, \dots, C_m$ considering the products $\prod_{i=1}^{m} P_{C_i}$ (the iterative algorithm), convex combination of P_{C_i}, $i = 1, \dots, m$ (the Cimmino algorithm) and a more recent and advanced dynamic string-averaging algorithm introduced by Y. Censor, T. Elfving, and G. T. Herman in (2001) for solving a convex feasibility problem, when a given collection of sets is divided into blocks and the algorithms operate in such a manner that all the blocks are processed in parallel.

In Chapter 2 of the book A. J. Zaslavski, Optimization on Solution Sets of Common Fixed Point Problems, Springer Optimization and Its Applications, 2021 (AZ21) we consider a minimization of a convex function on a common fixed point set of a finite family of quasinonexpansive mappings in a Hilbert space. Our goal is to obtain a good approximate solution of the problem in the presence of computational errors. We use the Cimmino subgradient algorithm, the iterative subgradient algorithm and the dynamic string-averaging subgradient algorithm and show that each of them generates a good approximate solution, if the sequence of computational errors is bounded from above by a small constant. Moreover, if we known computational errors for our algorithm, we find out what an approximate solution can be obtained and how many iterates one needs for this.

Let $(X, \langle \cdot, \cdot \rangle)$ be a Hilbert space with an inner product $\langle \cdot, \cdot \rangle$ which induces a complete norm $\| \cdot \|$.

Suppose that m is a natural number, $\bar{c} \in (0, 1]$, $P_i : X \to X$, $i = 1, \dots, m$, for every integer $i \in \{1, \dots, m\}$,

$$\mathrm{Fix}(P_i) := \{z \in X : \ P_i(z) = z\} \neq \emptyset$$

and that the inequality

$$\|z - x\|^2 \geq \|z - P_i(x)\|^2 + \bar{c}\|x - P_i(x)\|^2$$

holds for every for every integer $i \in \{1, \dots, m\}$, every point $x \in X$ and every point $z \in \mathrm{Fix}(P_i)$. Set

$$F = \cap_{i=1}^{m} \mathrm{Fix}(P_i).$$

For every positive number ϵ and every integer $i \in \{1, \dots, m\}$ set

$$F_\epsilon(P_i) = \{x \in X : \|x - P_i(x)\| \le \epsilon\},$$

$$\tilde{F}_\epsilon(P_i) = F_\epsilon(P_i) + B(0, \epsilon),$$

$$F_\epsilon = \cap_{i=1}^m F_\epsilon(P_i),$$

$$\tilde{F}_\epsilon = \cap_{i=1}^m \tilde{F}_\epsilon(P_i)$$

and

$$\widehat{F}_\epsilon = F_\epsilon + (0, \epsilon).$$

A point belonging to the set F is a solution of our common fixed point problem while a point which belongs to the set $\tilde{F}_\epsilon$ is its ϵ-approximate solution.

Let $M_* > 0$ satisfy

$$F \cap B(0, M_*).$$

and let $f : X \to R^1$ be a convex continuous function. In AZ21a we consider the minimization problem

$$f(x) \to \min, \; x \in F.$$

Assume that

$$\inf(f, F) = \inf(f, F \cap B(0, M_*)).$$

Fix $\alpha > 0$. Let us describe our first algorithm.

Cimmino subgradient algorithm

Initialization: select an arbitrary $x_0 \in X$.

Iterative step: given a current iteration vector $x_k \in X$ calculate

$$l_k \in \partial f(x_k),$$

pick $w_{k+1} = (w_{k+1}(1), \ldots, w_{k+1}(m)) \in R^m$ such that

$$w_{k+1}(i) \geq 0, \; i = 1, \ldots, m,$$

$$\sum_{i=1}^{m} w_{k+1}(i) = 1$$

and define the next iteration vector

$$x_{k+1} = \sum_{i=1}^{m} w_{k+1}(i) P_i(x_k - \alpha l_k).$$

In AZ21a this algorithm is studied under the presence of computational errors and two convergence results are obtained.

Fix

$$\Delta \in (0, m^{-1}).$$

We suppose that $\delta_f \in (0, 1]$ is a computational error produced by our computer system, when we calculate a subgradient of the objective function f while $\delta_p \in [0, 1]$ is a computational error produced by our computer system, when we calculate the operators P_i, $i = 1, \ldots, m$. Let $\alpha > 0$ be a step size.

Cimmino subgradient algorithm with computational errors

Initialization: select an arbitrary $x_0 \in X$.

Iterative step: given a current iteration vector $x_k \in X$ calculate

$$\xi_k \in \partial f(x_k) + B(0, \delta_f),$$

pick $w_{k+1} = (w_{k+1}(1), \dots, w_{k+1}(m)) \in R^m$ such that

$$w_{k+1}(i) \geq \Delta, \ i = 1, \dots, m,$$

$$\sum_{i=1}^{m} w_{k+1}(i) = 1,$$

calculate

$$y_{k,i} \in B(P_i(x_k - \alpha\xi_k), \delta_p), \ i = 1, \dots, m$$

and the next iteration vector and $x_{t+1} \in X$ such that

$$\|x_{t+1} - \sum_{i=1}^{m} w_{t+1}(i) y_{t,i}\| \leq \delta_p.$$

In this algorithm, as well for other algorithms considered in the book, we assume that the step size does not depend on the number of iterative step k. The same analysis can be done when step sizes depend on k. On the other hand, as it was shown in AZ16a, AZ20a, in the case of computational errors the best choice of step sizes is step sizes which do not depend on iterative step numbers.

In the following result obtained in AZ21a we assume the the objective function f satisfies a coercivity growth condition.

Theorem 2 *Let the function f be Lipschitz on bounded subsets of X, $\lim_{\|x\| \to \infty} f(x) = \infty$, $M \geq 2M_* + 8$, $L_0 \geq 1$,*

$$M_1 > \sup\{|f(u)| : u \in B(0, M_* + 4)\} + 4,$$

$$f(u) > M_1 + 4 \text{ for all } u \in X \setminus B(0, 2^{-1}M),$$

$$|f(z_1) - f(z_2)| \leq L_0 \|z_1 - z_2\|$$

$$\text{for all } z_1, z_2 \in B(0, 3M + 4),$$

$\delta_f, \delta_p \in [0, 1], \alpha > 0$ satisfy

$$\alpha \leq L_0^{-2},\ \alpha \geq \delta_f(6M + L_0 + 2),\ \alpha \geq 2\delta_p(6M + 2), \tag{1.1}$$

T be a natural number and let

$$\gamma_T = \max\{\alpha(L_0 + 1),$$

$$(\Delta\bar{c})^{-1/2}(4M^2T^{-1} + \alpha(L_0 + 1)(12M + 4))^{1/2} + \delta_p\}.$$

Assume that $\{x_t\}_{t=0}^{T} \subset X$, $\{\xi_t\}_{t=0}^{T-1} \subset X$,

$$(w_t(1), \dots, w_t(m)) \in R^m, \ t = 1, \dots, T,$$

$$\sum_{i=1}^{m} w_t(i) = 1, \ t = 1, \dots, T,$$

$$w_t(i) \geq \Delta, \ i = 1, \dots, m, \ t = 1, \dots, T,$$

$$x_0 \in B(0, M)$$

and that for all integers $t \in \{0, \dots, T-1\}$,

$$B(\xi_t, \delta_f) \cap \partial f(x_t) \neq \emptyset,$$

$$y_{t,i} \in B(P_i(x_t - \alpha \xi_t), \delta_p), \ i = 1, \dots, m,$$

$$\|x_{t+1} - \sum_{i=1}^{m} w_{t+1}(i) y_{t,i}\| \leq \delta_p.$$

Then

$$\|x_t\| \le 2M + M_*,\ t = 0, \dots, T$$

and

$$\min\{\max\{\Delta \bar{c} \sum_{i=0}^{m} \|x_t - \alpha \xi_t - y_{t,i}\|^2 - \alpha(L_0 + 1)(12M + 4),$$

$$2\alpha(f(x_t) - \inf(f, F)) - 4\delta_p(6M + 3)$$

$$-\alpha^2 L_0^2 - 2\alpha(6M + L_1 + 1)\} :$$

$$t = 0, \dots, T - 1\} \le 4M^2 T^{-1}.$$

Moreover, if $t \in \{0, \dots, T-1\}$ and

$$\max\{\Delta \bar{c} \sum_{i=0}^{m} \|x_t - \alpha\xi_t - y_{t,i}\|^2 - \alpha(L_0+1)(12M+4),$$

$$2\alpha(f(x_t) - \inf(f, F)) - 4\delta_p(6M+3)$$

$$-\alpha^2 L_0^2 - 2\alpha\delta_f(6M+L_0+1)\} \leq 4M^2T^{-1}, \quad (1.2)$$

then

$$f(x_t) \leq \inf(f, F) + 2M^2(T\alpha)^{-1}$$

$$+2\alpha^{-1}\delta_p(6M+3) + 2^{-1}\alpha L_0^2 + \delta_f(6M+L_0+3) \quad (1.3)$$

and

$$x_t \in \widehat{F}_{\gamma_T}.$$

In AZ21awe also obtain an extension of this result when instead of assuming that f satisfies the growth condition we suppose that there exists $r_0 \in (0,1]$ such that the set F_{r_0} is bounded.

In Theorem 2 the computational errors δ_f, δ_p are fixed. Assume that they are positive. Let us choose α, T. First, we choose α in order to minimize the right-hand side of (1.3). Since T can be an arbitrary large we need to minimize the function

$$2\alpha^{-1}\delta_p(6M+3) + 2^{-1}\alpha L_0^2,\ \alpha > 0.$$

Its minimizer is

$$\alpha = 2L_0^{-1}(\delta_p(6M+3))^{1/2}.$$

Since α satisfies (1.1) we obtain the following restrictions on δ_f, δ_p:

$$\delta_f \le 2L_0^{-1}\delta_p^{1/2}(6M+3)^{1/2}(6M+L_0+2)^{-1},$$

$$\delta_p \le 4^{-1}L_0^{-2}(6M+3)^{-1}.$$

In this case

$$\gamma_T = \max\{2L_0^{-1}(\delta_p(6M+3))^{1/2}(L_0+1),$$

$$(\Delta\bar{c})^{-1/2}(4M^2T^{-1}$$

$$+2L_0^{-1}(\delta_p(6M+3))^{1/2}(L_0+1)(12M+4))^{1/2}+\delta_p\}.$$

We choose T with the same order as δ_p^{-1} For example, $T = \lfloor \delta_p^{-1} \rfloor$. In this case in view of Theorem 2, there exists $t \in \{0, \dots, T-1\}$ such that then

$$f(x_t) \le \inf(f, F) + c_1\delta_p^{1/2} + \delta_f(6M+L_0+3)$$

and $x_t \in \widehat{F}_{c_2\delta_p^{1/4}}$ where c_1, c_2 are positive constants which depend on $M, L_0, \Delta, \bar{c}$.

Let us explain how we can obtain t satisfying (1.2). Set

$$E = \{t \in \{0, \dots, T-1\} : \ \Delta \bar{c} \sum_{i=0}^{m} \|x_t - \alpha \xi_t - y_{t,i}\|^2$$

$$\leq \alpha(L_0 + 1)(12M + 4) + 4M^2 T \alpha^{-1}\}$$

and find $t_* \in E$ such that $f(x_{t_*}) \leq f(x_t)$ for all $t \in E$. This t satisfies (1.2).

In AZ21a we also establish analogs of Theorem 1.2 for the iterative subgradient algorithm and the dynamic string-averaging subgradient algorithm.