2012 American Control Conference
Fairmont Queen Elizabeth, Montréal, Canada
June 27-June 29, 2012

# Randomized Smoothing for Near-Convex Functions in Context of Image Processing

I. Minin, A. Vakhitov

*Abstract*— **In this paper the problem of optimization of near-convex functions which can be represented as a sum of strongly convex and bounded functions is addressed. We have noted that in several optimization problems in area of image processing cost functions follow this model. Here we present an algorithm how to minimize such functions using randomized smoothing technique. The technique is an attractive theoretical justification of global optimization properties of SPSA-like algorithms. We present bounds on estimates error after finite number of steps, asymptotic bounds, bounds on estimates' variance and show how the algorithm presented can robustly optimize a function with many similar local minima.**

## I. Introduction

Research in the area of randomized algorithms of stochastic approximation started in second half of XX century in [1]–[4]. Randomization allowed to reduce the number of function measurements needed to achieve certain quality of minimum estimates [4] or even, in case of additional measurement noise, achieve optimal convergence rate in general class of zero-order stochastic approximation algorithms [2], cancel out unknown but bounded measurement noise [3] or smoothen non-differentiable functions [5].

Mostly, however, the research in the area was devoted to local optimization, where there is one minimum which needs to be found. Here we address a problem when there are many local minima, however the global one (the smallest) is of interest. If there is nothing known about the function, the problem of optimization reduces to simple and very resources consuming search in the whole area of parameters.

In this paper we assume that there exists a particular model of the function, namely it is a sum of a strongly convex and some another bounded function. We call such functions "near-convex". This assumption allows us to rely on global properties of strongly convex functions. To differentiate a "good" strongly convex function from "bad" corruption function, we use randomized smoothing technique. In the paper of Yin [6], another type of globally convergent stochastic approximation algorithms with noise injection resembling simulated annealing algorithm are studied. In the paper [7] the similar setting to ours is explored, namely global optimization with SPSA algorithms without noise injection. However, theoretical justification is not clear and the effects reported are hard to reproduce, so the method proposed seems to be less reliable than it could potentially be.

Smoothing is a result of convolution of the initial function with some kernel, which can be implemented with the help of randomization. We restrict ourselves to a class of functions $f(x)$ being a sum $f(x) = f_0(x) + v(x)$ of strongly convex function $f_0(x)$ and bounded function $v(x)$ (we call them near-convex). We show that this approach can be used in applications of optimization in the area of image processing.

In papers of Katkovnik [1] the possibility of smoothing a function in order to overcome local minima while minimizing with gradient-type procedures was mentioned but from our point of view analysis of convergence and applicability of the results to practical problems was up to now missing. Recently a lot of interest in global minimization of non-convex functions using averaging based on convolution arose in physics. The Lennard-Jones problem is to find a configuration of particles with global minimum of energy [8], [9]. It is shown that a way to deal with this function minimization is to smoothen it and then minimize a smooth version [9].

There is a number of challenging problems in image processing such as image restoration, registration, segmentation e.t.c. which often require large-scale nonsmooth, nonconvex optimization.

Image registration problem [10], [11] may be stated as: given two images taken, for example, at different times, from different devices or perspectives, the goal is to determine a reasonable transformation, such that a transformed version of the first image is similar to the second one. The applications to this are stereovision and optical flow calculation [10], reconstruction of objects and scenes from multiple views [11], [12], etc.

There are two general types of cost functions used in the context of image registration:sum of squared differences between pixel intensities $f_{SSD}(x) = \sum_{p \in W} w(p) \| I(x + p) - J(p) \|^2$ where $p \in \mathbb{R}^2$ is pixel of pattern $J$ and $I$ is an image where the search for patttern is performed, $w(p)$ is some weighting function [10], or pixelwise correlation $f_{COR} = \sum_{p \in W} w(p) I(x + p)^T J(p)$. In image registration mostly Newton-type procedures are used to optimize the cost functions [11]. It is important to increase convexity region of $f_{SSD}$ or $f_{COR}$ in order to improve convergence. Usually the method of pyramidal smoothing is used: images $I$ and $J$ are smoothed with some kernel and the minimum is found, then smoothing is repeated with smaller kernel support and optimization is started from the previous optimum point, and so on until convergence [11], [13]. However, this scheme has no theoretical justifications from the side of optimization: there is no closed-loop solution for the parameters of such an algorithm in order to make it convergent in some particular case, so that trial and error way is used.

I. Minin and A. Vakhitov are with the Saint-Petersburg State University, Saint-Petersburg, Russia alexander.vakhitov@gmail.com

978-1-4577-1096-4/12/$26.00 ©2012 AACC

The algorithm proposed here aims to fill this gap and offer theoretical ground to smoothing-based optimization of near-convex functions. We assume some model of a cost function and show how to get parameters of smoothing and gradient-descent algorithm in order to make it convergent to the small neighbourhood of a true minimum. We are going to present an algotihm based on randomized averaging which will help us to find a point close to true optimum of the initial function. Because of the "corruption" we cannot find the true minimum point anyhow. However, we will see that

- the lesser the "corruption", the closer is the solution to the true minimum;
- the higher strong convexity of the initial function, the higher is the tolerance of the algorithm to "corruption".

The article presents convergence conditions, analysis of the speed of convergence of the estimates, analysis of convergence of the estimates' variance and experiments with synthetic data and a real problem of image registration. We will do the analysis for a fixed step type of algorithm, because from our perspective it is more relevant to practice [14].

## II. Problem Setting

Assume that $f(x) : Q \subset \mathbb{R}^q \to \mathbb{R}$ is a non-convex function, but $Q$ is a convex set and

$$f(x) = f_0(x) + v(x),$$

where $f_0(x)$ is differentiable, strongly convex with parameter $\mu > 0$:

$$f_0(x) \geq f_0(y) + \langle \nabla f_0(y), x - y \rangle + \frac{\mu}{2}\|x - y\|^2,$$

for all $x, y \in \mathbb{R}^q$ and its gradient is Lipschitz-continuous with constant $L > 0$:

$$\|\nabla f_0(x) - \nabla f_0(y)\| \leq L\|x - y\|$$

and $v(x)$ is a continuous function with properties defined below.

The minimum of $f_0$ is located at the point $\theta_0$:

$$\theta_0 = argmin\ f_0(x).$$

The problem is to find a point $\theta_*$ in $\epsilon_\theta$ - neighbourhood of $\theta_0$:

$$\|\theta_* - \theta_0\| \leq \epsilon_\theta,$$

where we denote euclidean norm as $\|\cdot\|$.

### A. Averaging Kernel

We will use the averaging kernels to solve our problem. The averaging kernel definition is based on potential averaging operators from [1].

**Definition.** Averaging kernel of degree 1 is a function

$$h :\ Q \subset \mathbb{R}^q \to \mathbb{R},\ \forall u \in Q\, h(u) \geq 0,$$

for which the following properties hold:

$$\int_Q h(u)u^{(i)}du = 0;\ \int_Q h(u)du = 1,$$

and $Q$ is open set, or $Q$ is closed and a.s.

$$\forall u \in \partial Q\ h(u) = 0.$$

Averaging kernels have a useful for gradient-based algorithms property [1] holding for a very general class of functions $f$ and scalars $b > 0$:

$$\frac{\partial}{\partial x}\int_Q h(u)f(x - bu)du = \frac{1}{b}\int_Q h'(u)f(x - bu)du,$$

where $h'(u)$ is a gradient of $h$.

We will decompose the functions $h$ and $h'$ as

$$h(u) = c(u)p(u),\ h'(u) = d(u)p(u).$$

Examples of averaging kernels suitable for this paper taken from [1] are listed in the following table.

| $p(u)$ | $c(u)$ | $d(u)$ |
|---|---|---|
| $\frac{1}{(2\pi)^{q/2}}e^{-\|u\|^2/2}$ | $1$ | $-u$ |
| $\frac{1}{(2\pi)^{q/2}}e^{-\|u\|^2/2}$ | $1 + \frac{q}{2} - \frac{\|u\|^2}{2}$ | $-(2 + \frac{q}{2} - \frac{\|u\|^2}{2})u$ |

We denote

$$h_1 = \int_Q h(u)\|u\|,\ h_2 = \int_Q h(u)\|u\|^2.$$

### B. "Corruption" Properties

Let us assume that $v(x)$ is a differentable function as well. (This assumption can be relaxed).

We assume also that there exists finite scalar functions $C_1(b) > 0, C_2(b)$ such that:

$$\forall b > b_0\ \ |\int_Q h'(u)v(x - bu)du| < C_1(b),$$

$$\int_Q d^2(u)v^2(x \pm bu)du < C_2(b).$$

## III. Algorithm

Let us define parametric statistical gradient [1] as

$$\eta(x, N, b) = \frac{1}{Nb}\sum_{j=1}^{N}\eta_j(x, b),$$

$$\eta_j(x, b) = d(u_j)f(x - bu_j),\ d(u_j) = \frac{h'(u_j)}{p(u_j)}, \qquad (1)$$

where $p(\cdot)$ is some probability density function, and $u_j$ are sampled from corresponding distribution.

It is also possible to use another form of gradient estimate:

$$\eta_j(x, b) = \frac{1}{2}d(u_j)\big(f(x - bu_j) - f(x + bu_j)\big), \qquad (2)$$

which is better in practice as we will see from the theorems below and the simulation.

Starting with some initial point $\hat{\theta}_0$,

$$\hat{\theta}_n = \hat{\theta}_{n-1} - \alpha_n\eta(x, N, b_n), \qquad (3)$$

where $\alpha_n, b_n$ are some scalar sequences.

*A. Properties of the algorithm*

In the following section we denote $f_h = \int_Q h(u) f_0(x - bu)du$, $\theta_*$ is the minimum of $f_h$, $d_2 = E\|u\|^2 d^2(u)$. We will analyze the properties of the smoothed function, convergence of the algorithm, behavior of the variance of its estimates.

**Lemma 1.** When the properties for $h$ and $f_0$ listed above hold, $f_h$ is a strongly convex function with constant $\mu$ and its gradient has Lipschitz property with constant $L$.

**Proof.** Consider one of the equivalent formulations for a definition of a strongly convex function $f_0$ [15]:

$$\forall x, y \forall \lambda \in (0,1)\ f_0(\lambda x + (1-\lambda)y) \le \lambda f_0(x) +$$
$$+(1-\lambda) f_0(y) - \lambda(1-\lambda)\frac{\mu}{2}\|x - y\|.$$

This inequality is true for $f_0$. Let us substitute $x = x' - bu$, $y = y' - bu$, then multiply both sides of the inequality on $h(u) \ge 0$, and the integrate over $u \in Q$. We will get the same inequality for $f_h$.

The Lipschitz property can be checked directly using its definition.

**Theorem 1.**

$$\|\theta_* - \theta_0\| \le \frac{L}{\mu} b h_1.$$

**Proof.** Using the fact that $\nabla f_h(\theta_*) = 0$, we get

$$\int_Q h(u)\nabla f_0(\theta_* - bu) - \nabla f_0(\theta_*)du = -\nabla f_0(\theta_*).$$

From the strong convexity of $f_0$,

$$\|\nabla f_0(\theta_*)\| \ge \mu\|\theta_* - \theta_0\|;$$

from the Lipschitz property of $\nabla f_0$,

$$\| \int_Q h(u)\nabla f_0(\theta_* - bu)du - \nabla f_0(\theta_*)du\| \le$$

$$\int_Q h(u)\|\nabla f_0(\theta_* - bu)du - \nabla f_0(\theta_*)\|du \le L b h_1;$$

So, we have

$$\mu\|\theta_* - \theta_0\| \le L b h_1,$$

and

$$\|\theta_* - \theta_0\| \le \frac{L}{\mu} b h_1.$$

**Lemma 2**. Variance of differentiation operators.

For the operator (1), variance of gradient estimate at point $x$ has a bound $\frac{1}{N} E d^2(u) \frac{2}{b^2} f_0^2(x - bu) + \frac{2}{b^2} C_2(b)$. In case of operator (2),

$$E(g_n - Eg_n)^2 \le \frac{1}{N}\big(6 d_2 L^2 - \mu^2)\|x - \theta_0\|^2 +$$

$$+(2C(b)L + 6\frac{L^3}{\mu} b h_1 d_2)\|x - \theta_*\| + \frac{L^2}{\mu^2} b^2 h_1^2\big).$$

**Proof.** In case of all operators due to the definition of variance, for any sequence of independent variables $\xi_i$, $i = 1 \dots N$ with equal mean $E\xi$

$$E(\frac{1}{N}\sum \xi_i - E\xi)^2 = \frac{1}{N} E(\xi - E\xi)^2.$$

In the same time, for the operator (1)

$$E(d(u)\frac{1}{b} f(x - bu))^2 \le E d^2(u)\frac{2}{b^2} f_0^2(x - bu) + \frac{2}{b^2} C_2(b).$$

For the operator (2),

$$E(d(u)\frac{1}{2b}(f(x + bu) - f(x - bu)))^2 \le E d^2(u)\frac{3}{4b^2}$$
$$\big((f_0(x + bu) - f_0(x - bu)))^2 + v^2(x + bu) + v^2(x - bu))\big)$$

$$f_0(x + sbu) = f_0(x) + \int_0^1 \langle \nabla f_0(x + tsbu), sbu\rangle dt,$$

$$\| \int_0^1 \langle \nabla f_0(x + tsbu), sbu\rangle dt\| \le Lb\|x - \theta_0\|\|u\| + \frac{L}{2} b^2\|u\|^2,$$

$$(f_0(x + bu) - f_0(x - bu)))^2 \le 2(4L^2 b^2\|u\|^2\|x - \theta_0\|^2 +$$
$$+L^2 b^4\|u\|^4)$$

$$E(d(u)\frac{1}{2b}(f(x + bu) - f(x - bu)))^2 \le 6 d_2 L^2\|x - \theta_0\|^2 +$$

$$+\frac{3}{2b^2} C_2(b) \le 6 d_2 L^2\|x - \theta_*\|^2 + 6\frac{L^3}{\mu} b h_1 d_2\|x - \theta_*\| + \frac{L^2}{\mu^2} b^2 h_1^2.$$

Also,

$$\|\nabla \tilde{f}(x)\|^2 = \|\nabla f_h(x)\|^2 + 2\nabla f_h(x)\nabla v_h(x) +$$
$$+\|\nabla v_h(x)\|^2 \ge \mu^2\|x - \theta_*\|^2 - 2C(b)L\|x - \theta_*\|.$$

As a result,

$$E(g_n - Eg_n)^2 \le \frac{1}{N}\big(6 d_2 L^2 - \mu^2)\|x - \theta_0\|^2 +$$

$$+(2C(b)L + 6\frac{L^3}{\mu} b h_1 d_2)\|x - \theta_*\| + \frac{L^2}{\mu^2} b^2 h_1^2\big).$$

**Theorem 2.** Let $f, f_0, v, f_h$ be as defined above. When $b_0 > 0$, if $\alpha_n = \alpha$,

$$\exists \epsilon > 0 :\ \nu = 2\alpha\mu - \frac{\epsilon^2}{2} - \alpha^2(\frac{1}{N}(6 d_2 L^2 - \mu^2) + L) \in (0,1)$$

then the algorithm (3) with operator (2) converges to $\epsilon_\theta$-neighbourhood of $\theta_*$ in average, $\epsilon_\theta = \frac{\phi}{\nu}$, where

$$\phi = \alpha^2\frac{\epsilon^{-2}}{2}\Bigg(2\frac{C_1(b)}{b}(1 + 2\alpha L) +$$

$$+\alpha\frac{2C_1(b)L + 6\frac{L^3}{\mu} b h_1 d_2}{N}\Bigg)^2 + \frac{L^2}{\mu^2} b^2 h_1^2 + b^{-2} C_1^2(b).$$

**Note.** From the theorem statement we see that when $N$ grows, $\phi$ becomes less and the upper bound of estimation error becomes lower. Also, we see that when $b$ grows, terms like $\frac{C_1(b)}{b}$ become less, but terms like $\frac{L^2}{\mu^2} b^2 h_1^2$ will grow. This represents a tradeoff between accuracy and smoothing of corruption function. Also, when $\mu$ grows, $\nu$ grows, so the asymptotic bound also can be smaller when strong convexity of the function becomes higher.

**Proof.**

$$\|\hat{\theta}_{n+1} - \theta_0\|^2 = \|\hat{\theta}_n - \theta_0 - \alpha g_n(\hat{\theta}_n)\|^2 \le \|\hat{\theta}_n - \theta_0\|^2 - \quad (4)$$
$$-2\alpha\langle g_n(\hat{\theta}_n), \hat{\theta}_n - \theta_0\rangle + \alpha^2\|g_n(\hat{\theta}_n)\|^2,$$

$$E_n g_n(\hat{\theta}_n) = \nabla_{\hat{\theta}_n} f_h(\hat{\theta}_n) = \nabla_{\hat{\theta}_n} \int h(u) f_0(\hat{\theta}_n - bu) +$$

$$+\nabla_{\hat{\theta}_n} \int h(u) v(\hat{\theta}_n - bu)$$

$$\nabla \int h(u) v(\hat{\theta}_n - bu) = -\int_{\partial Q} h(u) v(x - bu) \cos(\widehat{x, n(u)}) du +$$

$$+\frac{1}{b} \int h'(u) v(\hat{\theta}_n - bu);$$

According to the definition of $h(u)$,

$$\int_{\partial Q} h(u) v(x - bu) \cos(\widehat{x, n(u)}) du = 0;$$

$$\| \int h'(u) v(\hat{\theta}_n - bu) \| \le C_1(b).$$

$$\langle \int h(u) \nabla f_0(\hat{\theta}_n - bu) du, \hat{\theta}_n - \theta_* \rangle =$$

$$= \int \langle h(u) \nabla f_0(\hat{\theta}_n - bu), \hat{\theta}_n - \theta_* \rangle du \ge$$

$$\ge \int h(u) \mu \|\hat{\theta}_n - \theta_*\|^2 du =$$

$$= \mu \|\hat{\theta}_n - \theta_*\|^2.$$

Then,

$$E_n\{-2\alpha \langle g_n(\hat{\theta}_n), \hat{\theta}_n - \theta_* \rangle\} \le$$

$$\le -2\alpha\mu \|\hat{\theta}_n - \theta_*\|^2 +$$

$$+2\alpha \frac{C}{b} \|\hat{\theta}_n - \theta_*\|.$$

We can bound the last term of inequality (4) as

$$E\alpha^2 \|g_n(\hat{\theta}_n)\|^2 = E\alpha^2 \| \frac{1}{Nb} \sum \eta_j(\hat{\theta}_n, b) \|^2 =$$

Adding and subtracting $\nabla \tilde{f}(\hat{\theta}_n) = \nabla \int_Q h(u) f(\hat{\theta}_n - bu) du$, we get

$$= E\alpha^2 \| \frac{1}{Nb} \sum \eta_j(\hat{\theta}_n, b) - \nabla \tilde{f}(\hat{\theta}_n) + \nabla \tilde{f}(\hat{\theta}_n) \|^2.$$

Using the fact that $E \frac{1}{Nb} \sum \eta_j(\hat{\theta}_n, b) = \nabla \tilde{f}(\hat{\theta}_n)$ we get

$$E\alpha^2 \|g_n(\hat{\theta}_n)\|^2 = \alpha^2 \|\nabla \tilde{f}(\hat{\theta}_n)\|^2 +$$

$$+E\alpha^2 \| \frac{1}{Nb} \sum \eta_j(\hat{\theta}_n, b) - \nabla \tilde{f}(\hat{\theta}_n) \|^2.$$

Considering the first term,

$$\|\nabla \tilde{f}(\hat{\theta}_n)\|^2 = \|\nabla f_h(\hat{\theta}_n)\|^2 + 2\langle \nabla f_h(\hat{\theta}_n), \nabla v_h(\hat{\theta}_n) \rangle +$$

$$+\|\nabla v_h(\hat{\theta}_n)\|^2,$$

where we denoted $f_h(x) = \int_Q h(u) f_0(x - bu) du$, $\nabla v_h = \frac{1}{b} \int_Q h'(u) v(x - bu) du$. From the properties of $f_0$ and $v$,

$$\|\nabla \tilde{f}(\hat{\theta}_n)\|^2 \le L\|\hat{\theta}_n - \theta_*\|^2 + 2LC_1(b) b^{-1} \|\hat{\theta}_n - \theta_*\| + b^{-2} C_1^2(b).$$

From the Lemma 2 we get the bound for the last term. Summarizing, we get

$$E\|g_n(\hat{\theta}_n)\|^2 \le (\frac{1}{N}(6d_2L^2 - \mu^2) + L)\|\hat{\theta}_n - \theta_*\|^2 +$$

$$+\alpha^2 (\frac{1}{N}(2C_1(b)L + 6\frac{L^3}{\mu} b h_1 d_2) + 2LC_1(b) b^{-1}) \|\hat{\theta}_n - \theta_*\| +$$

$$+\frac{L^2}{\mu^2} b^2 h_1^2 + b^{-2} C_1^2(b).$$

The term with first degree of estimation error using Cauchy-Bunyakovsky-Schwarz inequality with arbitrary $\epsilon > 0$ is bound as

$$\alpha\big(2\frac{C_1(b)}{b}(1 + 2\alpha L) + \alpha \frac{2C_1(b)L + 6\frac{L^3}{\mu} b h_1 d_2}{N}\big) \|\hat{\theta}_n - \theta_0\| \le$$

$$\le \frac{\epsilon^2}{2} \|\hat{\theta}_n - \theta_*\|^2 + \alpha^2 \frac{\epsilon^{-2}}{2} \Big( 2\frac{C_1(b)}{b}(1 + 2\alpha L) +$$

$$+\alpha \frac{2C_1(b)L + 6\frac{L^3}{\mu} b h_1 d_2}{N} \Big)^2.$$

Finally, we have an inequality

$$\|\hat{\theta}_{n+1} - \theta_0\|^2 \le (1 - 2\alpha\mu + \frac{\epsilon^2}{2} + \alpha^2 (\frac{1}{N}(6d_2L^2 - \mu^2) + L)) \cdot$$

$$\cdot \|\hat{\theta}_n - \theta_*\|^2 + \alpha^2 \frac{\epsilon^{-2}}{2} \Big( 2\frac{C_1(b)}{b}(1 + 2\alpha L) +$$

$$+\alpha \frac{2C_1(b)L + 6\frac{L^3}{\mu} b h_1 d_2}{N} \Big)^2 + \frac{L^2}{\mu^2} b^2 h_1^2 + b^{-2} C_1^2(b).$$

Denoting the coefficient of second degree of error as $1 - \nu_n$ and the rest term as $\phi_n$ and applying the unconditioned expectation we get

$$E\{\|\hat{\theta}_{n+1} - \theta_0\|^2\} \le (1 - \nu_n) E\{\|\hat{\theta}_n - \theta_0\|^2\} + \phi_n.$$

We have $\nu_n = \nu$, $\phi_n = \phi$ and

$$\lim_{n \to \infty} E\{\|\hat{\theta}_{n+1} - \theta_0\|^2\} \le \frac{\phi}{\nu},$$

while for the $n$-th estimate we have the inequality

$$E\{\|\hat{\theta}_{n+1} - \theta_0\|^2\} \le (1 - \nu)^n E\{\|\hat{\theta}_0 - \theta_0\|^2\} + \frac{\phi\big(1 - (1 - \nu)^n\big)}{\nu}.$$

Next we will analyze the variance of estimates.

**Theorem 2.** Let $f, f_0, v, f_h$ be as defined above. When $b_0 > 0$, if $\alpha_n = \alpha$,

$$\exists \epsilon > 0 : \ \nu = 2\alpha\mu - \frac{\alpha^2}{N}\big(6d_2L^2 - \mu^2\big) - \frac{\epsilon^2}{2} \in (0, 1)$$

then the variance of estimates of the algorithm (3) with operator (2) has asymptotic upper bound:

$$\lim_{n \to \infty} E\|\hat{\theta}_n - E\hat{\theta}_n\|^2 \le \frac{\phi}{\nu},$$

$$\phi = \frac{\epsilon^{-2}}{2} \alpha^2 (\alpha(2C_1(b)L + 6\frac{L^3}{\mu} b h_1 d_2) +$$

$$+4\frac{C_1(b)}{b}) + \frac{\alpha^2 L^2}{\mu^2} b^2 h_1^2.$$

and is bounded as

$$\sigma_{n+1}^2 \le (1 - \nu)^{n-1} \sigma_1^2 + (1 - (1 - \nu)^{n-1}) \frac{\phi}{\nu},$$

where $\sigma_1$ is the variance of $\hat{\theta}_1$.

**Proof.**

$$E\|\hat{\theta}_{n+1} - E\hat{\theta}_{n+1}\|^2 = E\|\hat{\theta}_n - \alpha g_n(\hat{\theta}_n) -$$
$$-E\hat{\theta}_n + \alpha E g_n(\hat{\theta}_n)\|^2 = E\|\hat{\theta}_n - E\hat{\theta}_n\|^2 -$$
$$-2\alpha E\langle g_n(\hat{\theta}_n) - E g_n(\hat{\theta}_n), \hat{\theta}_n - E\hat{\theta}_n\rangle +$$
$$+\alpha^2 E\|g_n(\hat{\theta}_n) - E g_n(\hat{\theta}_n)\|^2.$$

For the second term the following inequality holds:

$$-2\alpha E\langle g_n(\hat{\theta}_n) - E g_n(\hat{\theta}_n), \hat{\theta}_n - E\hat{\theta}_n\rangle =$$
$$-2\alpha\big(E\langle g_n(\hat{\theta}_n) - g_n(E\hat{\theta}_n), \hat{\theta}_n - E\hat{\theta}_n\rangle +$$
$$+E\langle g_n(E\hat{\theta}_n) - E g_n(\hat{\theta}_n), \hat{\theta}_n - E\hat{\theta}_n\rangle\big).$$

Since the last term is 0 because of independence of $\hat{\theta}_n$ and $g_n(E\hat{\theta}_n)$, we need only to bound the first term. We'll use then the well-known inequality for the strongly convex functions:

$$-2\alpha E\langle g_n(\hat{\theta}_n) - g_n(E\hat{\theta}_n), \hat{\theta}_n - E\hat{\theta}_n\rangle =$$
$$-2\alpha\langle\nabla\int_Q h(u)f_0(\hat{\theta}_n - bu)du - \nabla\int_Q h(u)f_0(E\hat{\theta}_n - bu)du,$$
$$\hat{\theta}_n - E\hat{\theta}_n\rangle - 2\alpha b^{-1}E\langle\int_Q h'(u)v(\hat{\theta}_n - bu)du -$$
$$-\int_Q h'(u)v(E\hat{\theta}_n - bu)du, \hat{\theta}_n - E\hat{\theta}_n\rangle \le$$
$$-2\alpha\mu E\|\hat{\theta}_n - E\hat{\theta}_n\|^2 + 4\alpha\frac{C_1(b)}{b}E\|\hat{\theta}_n - E\hat{\theta}_n\|.$$

We denote by $\sigma_n^2$ variance of estimates on the n-th step of estimation process. We get:

$$\sigma_{n+1}^2 \le (1 - 2\alpha_n\mu + \frac{\alpha^2}{N}\big(6d_2L^2 - \mu^2)\big)\sigma_n^2 +$$
$$+\alpha(\alpha(2C_1(b)L + 6\frac{L^3}{\mu}bh_1d_2) + 4\frac{C_1(b)}{b})\sigma_n + \frac{\alpha^2L^2}{\mu^2}b^2h_1^2.$$

Using the inequality with $\epsilon$ as in the previous theorem's proof, we get

$$\sigma_{n+1}^2 \le (1 - 2\alpha\mu + \frac{\alpha^2}{N}\big(6d_2L^2 - \mu^2\big) + \frac{\epsilon^2}{2})\sigma_n^2 +$$
$$+\frac{\epsilon^{-2}}{2}\alpha^2(\alpha(2C_1(b)L + 6\frac{L^3}{\mu}bh_1d_2) + 4\frac{C_1(b)}{b}) + \frac{\alpha^2L^2}{\mu^2}b^2h_1^2.$$

So, the analogous to theorem 1 result for the variances follows.

## IV. Experiments

### A. Real data

We have simulated our algorithm with an image from hubblesite.org representing a picture made by Hubble telescope (Fig. 1).

We have used $f_{SSD}$, $b = 2.0$, $\alpha_n = \frac{2}{(n+1)^{0.6}}$, $N = 4$, and stopping condition $||\hat{\theta}_{n+1} - \hat{\theta}_n|| < 0.01$ was satisfied after 200 iterations. The initial point was $\hat{\theta}_0 = (100, 30)^T$.

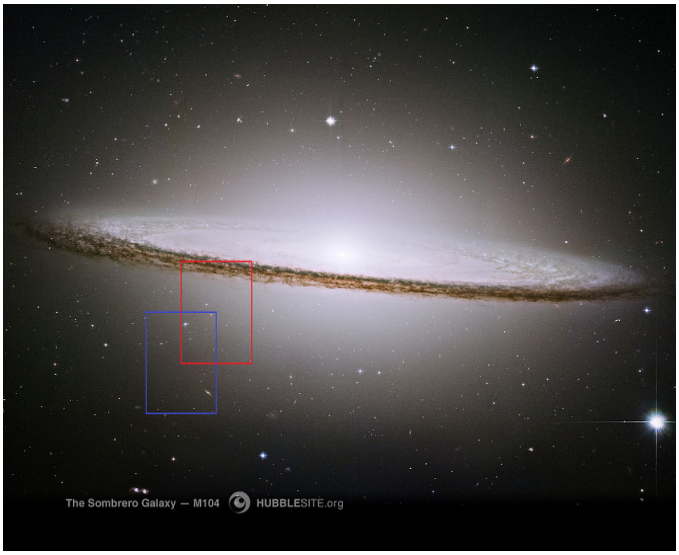


Fig. 1. Image of Sombrero galaxy, the pattern is marked by red square, initial estimate of the pattern position is marked by blue square.

### B. Simulated data

To test the algorithm we use a Griewank test function from [16] also used in [7]:

$$f(x) = cos(x^{(1)} - 100)cos(x^{(2)} - 100)/\sqrt{2} +$$
$$+\big((x^{(1)} - 100)^2 + (x^{(2)} - 100)^2\big)/4000 - 1.$$

The function is drawn at the Fig. 2. The strongly convex component has a very small slope towards a point $(100, 100)^T$ which can be noted by observing the height of the peaks while corruption function has comparatively big scale. The minimum is at the point $\theta = (100, 100)^T$.

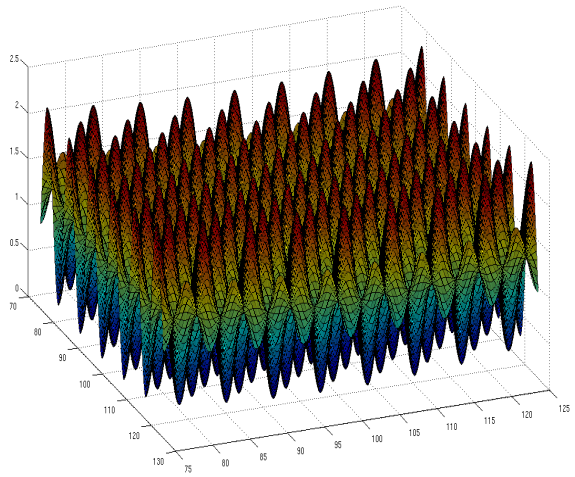

Fig. 2. Griewank function

In our experiments we used $b = 20$ and Gaussian kernel, as in the first line in the table above. The smoothed version of the function $f$ is drawn at the Fig. 3. The function is

convex and looks almost like strongly convex. The randomized algorithm presented in the paper on every step uses an estimate of the gradient of this smoothed function. The estimate converges to the true value as the number of function measurements per iteration $N$ grows.

Experimenting with this function, we found that the algorithm proposed in [7] (classical SPSA) diverges quite often even when we try to choose the best parameters of the algorithm. Algorithm used here for $N = 1$ resembles SPSA used by Maryak and Chin. Increasing $N$ we can benefit from better quality of the gradient estimate while increasing the number of function measurements which is impossible in classical SPSA formulation [3], [4]. When function measurements are too expensive it is not recommended, but nevertheless if reliable convergence is needed, some sacrifice in speed can be tolerated.

In the simulation we use the following parameters:

| | |
|---|---|
| $\alpha_n$ | $400/(n+1)^{0.3}$ |
| $b$ | 20 |
| $N$ | 500 |

We terminate the iterates when

$$\|\hat{\theta}_n - \hat{\theta}_{n-1}\| < 0.02.$$

The results of 50 runs from initial point randomly chosen from the interval $[-200; 400]x[-200; 400]$ are presented in the following table, where the final minimum estimate is denoted as $\hat{\theta}$.

| | |
|---|---|
| average $\|\hat{\theta} - \theta\|$ | 1.471 |
| max $\|\hat{\theta} - \theta\|$ | 4.403 |
| average number of iterations | 96.1 |

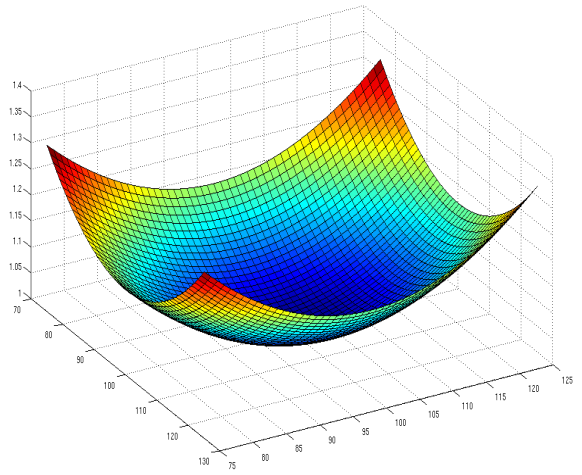

Fig. 3. Smoothed Griewank function with $b = 20$

## V. Conclusion

We have shown simple theoretical justification for global optimization of non-convex functions using SPSA-type algorithm and provided the simulation that shows flexibility and robustness of our approach comparing to classical SPSA. Smoothing (averaging) allows to get rid of corruption function and optimize only the most important strongly convex component of the function. From the theorem proved in the paper follows that absolute value of corruption can be higher for functions with higher degree of strong convexity; in the same time, when there is less additive corruption, minimum of strongly convex component is found more accurately. New algorithm allows to benefit from the tradeoff between amount of function measurements and better convergence due to higher quality of gradien estimates by adjusting the parameter $N$.

We would like to continue the chosen research direction with analysis of Newton-type procedures, smoothing of second derivatives, and also compare performance of new and traditional algorithms for smoothing in the context of image registration. Important problems not covered here are also optimal choice of averaging kernel and $N$. We'd like to thank the anonymous Reviewers for valuable comments and suggestions.

## References


[1] *Katkovnik V. Ya.* Method of Averaging Operators in Iterative Algorithms for Stochastic Extremum Problems Resolution // Cybernetics. 1972. No 4. Pp. 123–131.

[2] *Polyak B.T., Tsybakov A.B.* Optimal Accuracy Order for Stochastic Search Algorithms// Problemy Peredachi Informacii. 1990. No 26. Pp. 126–133.

[3] *Granichin O.N.* About One Stochastic Recursive Procedure with Dependent Noise in Observations, Using Input Perturbations // Vestnik LGU. Ser. 1. 1989. No 1(4). Pp. 19–21.

[4] *Spall J.C.* A Stochastic aproximation technique for generating maximum likelihood parameter estimates, // In Proc. of the American Control Conference. 1987. P. 1161–1167.

[5] *Yousefian F., Nedich A., and Shanbhag U. V.* On stochastic gradient and subgradient methods with adaptive steplength sequences, to appear in Automatica, 2011.

[6] *G. Yin.* Rates of Convergence for a Class of Global Stochastic Optimization Algorithms // SIAM J. Optim. Vol. 20, No. 1, pp. 99 - 120, 1999.

[7] *J.L. Maryak, D.C. Chin.* Global Random Optimization by Simultaneous Perturbation Stochastic Approximation // In Proc. of 2001 Winter Simulation Conference, eds. B.A. Peters, J.S. Smith, D.J. Medeiros, M.W. Rohrer, pp. 307 - 312, 2001.

[8] *Shao C.-S., Byrd R. H., Eskow E., Schnabel R. B.* Global optimization for molecular clusters using a new smoothing approach // Institute for Mathematics and Its Applications, Vol. 94, p.163, 1997.

[9] *Pappu R.V., Hart R.K., Ponde J.W.* Analysis and Application of Potential Energy Smoothing and Search Methods for Global Optimization // J. Phys. Chem. B, 1998, 102 (48), pp 97259742

[10] *Lucas N., Kanade T.*An Iterative Image Registration Technique with an Application to Stereo Vision // IJCAI (1981)

[11] J.Modersitzki *FAIR: flexible algorithms for image registration (Fundamentals of Algorithms)* SIAM, 2009.

[12] *Lieberknecht S., Benhimane S., Ilic S.* Simultaneous Reconstruction and Tracking of Non-planar Templates // Lecture Notes in Computer Science, 2011, Volume 6835/2011.

[13] *Bouguet J.* Pyramidal implementation of the Lucas Kanade feature tracker // Intel Corporation, Microprocessor Research Labs, 2000.

[14] *Granichin O., Gurevich L., and Vakhitov A.* Discrete-time minimum tracking based on stochastic approximation algorithm with randomized differences // In Proc. of the 48th IEEE Conf. on Decision and Control and 28th Chinese Control Conf. 2009. P. 5763–5767.

[15] Polyak B. T., *Introduction to Optimization.* New York: Optimization Software; 1987.

[16] J. Haataja. Using Genetic Algorithms for Optimization: Technology Transfer in Action, Chapter 1, pp. 3 - 22 in *Evolutionary Algorithms in Engineering and Computer Science* ed. M.M. Makela, P. Neitaanmaki, J. Periaux. Wiley; 1999.