International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

Random Search Method. Using Some Operators of Smoothing

George Filatov

Professor, Doctor of Techn. Sciences, Ukrainian State University of Chemical Technology, Ukraine

Absrtact **− In the paper are proposed two pseudo-gradient algorithms of search extremum of nonlinear functions with application of some operators of smoothing: an algorithm with the approximating spherical hyper-paraboloid and an algorithm with the approximating elliptical hyper-paraboloid. Are adduced: the statement of the problem, the algorithm of construction of operator of smoothing, the general scheme of search. The verification of the operation of the algorithms is illustrated in test examples. Are researched the problems of convergence of algorithms using operators of smoothing.**

Keywords**: Random Search Method, Operator of Smoothing, Pseudo-Gradient Algorithm.**

I. Inroduction

Many well-known search algorithms of minimum of functions and solving problems of non-linear programming [1, 2], which are based on the replacement of analytical expressions for gradient and second partial derivatives by the respective approximate formulas based on their difference approximation. This approach imposes certain requirements for the minimum number of calculations of functions at each step of search. It is not always convenient when calculating values of rather laborious functions. More efficient is using of pseudo-gradient algorithms [2] of smoothing type, using the procedure of multivariate approximation [3], convergence of which is proved in [2]. Introducing the approximation of increment of optimized function at each step as adaptation, and the random search as a means of gathering information about the behavior of the objective function, it is possible to construct algorithms that can be attributed to a class of pseudo-gradient [4].

II. The Statement Of The Problem

Let in *n*-dimensional Euclidean space E_n are given continuously differentiable functions $F(\mathbf{X}), g_i(\mathbf{X})$, $\mathbf{X} \in E_n$, $(i = 1,2,....,m)$.

The task of mathematical programming is reduced to minimizing the function $F(\mathbf{X})$ at the performance of restrictions

$$g_i(\mathbf{X}) = 0, \qquad \mathbf{X} \in E_n \quad (i = 1,2,....,m) \tag{1}$$

At now quite are well developed techniques for the bring the problem of nonlinear programming to the problem of calculating of unconditional extremum of some penalty function $f(\mathbf{X})$. In this case, if the original functions of the non-linear programming problem are continuously differentiable, then we can construct a continuously differentiable penalty function [5]. In this regard, speaking about of unconditional extremum of function $f(\mathbf{X})$, we assume that the task of finding its minimum in a certain sense, is equivalent to solving the original problem of nonlinear programming.

Thus let in *n*-dimensional Euclidean space E_n, is given the unimodal continuous, bounded below, differentiable function $F(\mathbf{X})$ whose gradient ∇F satisfies the Lipschitz condition:

$$\left\|\nabla f(\mathbf{X}_1) - \nabla f(\mathbf{X}_2)\right\| < L\left\|\mathbf{X}_1 - \mathbf{X}_2\right\| \tag{2}$$

It is also assumed that the sets

$$\mathbf{X}(C^1) = \{\mathbf{X} : f(\mathbf{X}) \le C^1\} \tag{3}$$

are limited. Here □ C^1 – an arbitrary constant.

The task is to find the point $\mathbf{X}^* \in E_n$, at which the function $f(\mathbf{X})$ reaches its minimum value:

$$f(\mathbf{X}^*) = \min_{X \in E_n} f(\mathbf{X}) \tag{4}$$

III. Building The Operator Of Smoothing

We approximate the increment of the function in the neighborhood of a point $\mathbf{X}_0 \in E_n$:

$$\Delta f(\mathbf{X}) = f(\mathbf{X}) - f(\mathbf{X}_0) = \sum_{j=1}^{N_1} \lambda_j(\mathbf{X}_0, R, \beta)\varphi_j(\mathbf{X} - \mathbf{X}_0) + \omega(\mathbf{X}) \tag{5}$$

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

At this $\omega(\mathbf{X}_0)=0$, and $\varphi(\mathbf{X})=\{\varphi_1(\mathbf{X}),\varphi_2(\mathbf{X}),...,\varphi_{N_1}(\mathbf{X})\}$, – vector of linearly independent functions of *n* variables.

The coefficients $\lambda_j(\mathbf{X}_0,R,\beta)$ can be found from the condition of the minimum of a quadratic functional [3, 6]:

$$\Phi(R,\beta)=1/2\int_{-\infty}^{\infty}\rho(\eta-\mathbf{X}_0,R,\beta)[\Delta f(\eta)-\delta(\eta,R,\beta)]^2 d\eta \ , \qquad (6)$$

Where $\delta(\eta,R,\beta)=\sum_{j=1}^{N_1}\lambda_j(\mathbf{X}_0,R,\beta)\varphi_j(\eta-\mathbf{X}_0)$; $\rho(\eta-\mathbf{X}_0,R,\beta)$ – the weighting function of the form:

$$\rho(\eta-\mathbf{X}_0,R,\beta)=\begin{cases} M(R,\beta) & \text{for} \quad \|\eta-\mathbf{X}_0\|\le R \\ \left(\dfrac{R}{\|\eta-\mathbf{X}_0\|}\right) & \text{for} \quad R<\|\eta-\mathbf{X}_0\|\le A_1 \\ 0 & \text{for} \quad \|\eta-\mathbf{X}_0\|>A_1 \end{cases} \qquad (7)$$

$\beta\ge 0$, $A_1>R$, $M(R,\beta)$ – □ normalizing factor selected from the condition:

$$\int_{-\infty}^{\infty}\rho(\eta-\mathbf{X}_0,R,\beta)d\eta=1 \ .$$

The parameters R and β have the following meaning: □ R – the radius of the neighborhood of point $\mathbf{X}_0$, in which is fulfilled the collection of information about the behavior $\Delta f(\mathbf{X})$ and the approximation is carried out with high accuracy, □ β – parameter characterizing the degree of utilization of previously collected information that is outside □ R – neighborhood of point $\mathbf{X}_0$.

According to [3,6] for $\delta(\mathbf{X},R,\beta)$ we can get an idea about of smoothing operator:

$$\delta(\mathbf{X},R,\beta)=\int_{-\infty}^{\infty}h(\mathbf{X}-\mathbf{X}_0,\eta-\mathbf{X}_0,R,\beta)\Delta f(\eta)d\eta$$

where:

$$h(\mathbf{X}-\mathbf{X}_0,\boldsymbol{\eta}-\mathbf{X}_0,R,\beta)=\rho(\boldsymbol{\eta}-\mathbf{X}_0,R,\beta)\varphi^T(\mathbf{X})D_1^{-1}\varphi(\boldsymbol{\eta})$$

$$D_1=h(\mathbf{X}-\mathbf{X}_0,\boldsymbol{\eta}-\mathbf{X}_0,R,\beta)=\int_{-\infty}^{\infty}\rho(\boldsymbol{\eta}-\mathbf{X}_0,R,\beta)\Delta f(\boldsymbol{\eta})d\boldsymbol{\eta},$$

from which we find parametric gradient at the point $\mathbf{X}_0$:

$$\delta'_X(\mathbf{X},R,\beta)=\int_{-\infty}^{\infty}h'_X(\mathbf{X}-\mathbf{X}_0,\eta-\mathbf{X}_0,R,\beta)\Delta f(\eta)d\eta \qquad (8)$$

The quality of approximation of increments $\Delta f(\mathbf{X})$ in the vicinity of the point $\mathbf{X}_0$ will be to assess by the value of:

$$u(R,\beta)=\left[\Phi(R,\beta)/\Phi_f(R,\beta)\right]^{1/2} \ ,$$

which we'll call as the level of approximation error. The value $\Phi_f(R,\beta)$ is defined as follows:

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

$$\Phi_f(R,\beta) = 1/2\int_{-\infty}^{\infty}\rho(\eta - \mathbf{X}_0, R, \beta)[\Delta f(\eta)]^2 d\eta$$

For smooth functions the level of approximation error tends to zero R and $\beta \to 0$. Valid out of (5): $\nabla\omega(\mathbf{X}) = \nabla f(\mathbf{X}) - \delta'_X(\mathbf{X}, R, \beta)$, at this at R and $\beta \to 0$ $\|\nabla\omega(\mathbf{X})\|^2 \to 0$, and, therefore, $\lim_{R,\beta\to 0}\chi(R,\beta) = 0$.

Thus, inputting the level of error υ^*, we can specify R and β such in order

$$\upsilon(R,\beta) \le \upsilon^* \text{ and } \left[\nabla f(\mathbf{X}_0)^T, \delta'_X(\mathbf{X}_0, R, \beta)\right] \ge 0.$$

IV. General Search Scheme

Using the parametric gradient of operator of smoothing (8) at each step, we'll describe the scheme of random search :

$$\mathbf{X}_N = \mathbf{X}_{N-1} - \gamma_{N-1}\Gamma(\mathbf{X}_{N-1})\delta'_X(\mathbf{X}_{N-1}, R_{N-1}, \beta_{N-1})$$

□ where $\gamma_{N-1} > 0, \Gamma(\mathbf{X}_{N-1})$ – a square matrix of size *n*, R_{N-1} и β_{N-1} at each step are selected from the condition of the scantiness level of approximation error.

Let the optimized function $f(\mathbf{X})$ satisfies all the conditions, set forth in claim 1, on parameters γ_N of the following restrictions are imposed: $\sum_{N=1}^{\infty}\gamma_N = \infty$ $\sum_{N=1}^{\infty}\gamma_N = \infty$ and $\overline{\lim_{N\to\infty}}\gamma_N < K_1$, where K_1 –□ the some predetermined level, and υ^* assigned in such a way that the selected out of $\upsilon(R_{N-1}, \beta_{N-1}) \le \upsilon^*$ values of R_{N-1} and β_{N-1} would satisfy the condition:

$$\left[\nabla f(\mathbf{X}_{N-1})^T, \Gamma(\mathbf{X}_{N-1})\delta'_X(\mathbf{X}_{N-1}, R_{N-1}, \beta_{N-1})\right] \ge \Delta(\varepsilon) > 0$$

at $\|\nabla f(\mathbf{X}_{N-1}\| \ge \varepsilon$ for all $\varepsilon > 0$. Then, by corollary 2 out of Theorem 1 in [2], in the sequence $\{\mathbf{X}_N\}$ formed by the algorithm, there will be found a subsequence $\{\overline{\mathbf{X}}_N\}$ and a point $\mathbf{X}^*$ that

$$\nabla f(\mathbf{X}^*) = 0, \quad \{\overline{\mathbf{X}}_N \xrightarrow{\text{dir}} \mathbf{X}^*\}, \quad f(\overline{\mathbf{X}}_N) \xrightarrow{\text{dir}} f(\mathbf{X}^*).$$

On the basis of the above we construct the following two algorithms.

V. The Algorithm Of Local Search With The Approximating Spherical Hyper-Paraboloid

Algorithm №1. We'll describe the optimized function in a neighborhood of point $\mathbf{X}_0$ with the help of spherical hyper-paraboloid passing through the point $\mathbf{X}_0$:

$$\delta(\mathbf{X}, R, \delta) = c_1\sum_{i=1}^{n}\left(x_i - x_i^{(0)}\right)^2 + 2\sum_{K=2}^{n+1}c_K\left(x_{K-1} - x_{K-1}^{(0)}\right)$$

Here C_K – coefficients determined on the basis of information obtained in the previous stages of search and as a result of the straggling of trials in R – neighborhood of point $\mathbf{X}_0$.

Let the search algorithm "remembers" the $m > n+1$ points. We introduce the notation:

$$b_{1j} = \|\mathbf{X}_j - \mathbf{X}_0\|^2, \quad b_{i+1,j} = 2\left(x_i^{(j)} - x_i^{(0)}\right), \quad y_j = \Delta f(\mathbf{X}_j), \quad \rho_j(\mathbf{X}_j - \mathbf{X}_0, R, \beta), \qquad (9)$$

EXPLORING RESEARCH AND INNOVATIONS

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

$$a_{iK} = \sum_{j=1}^{m} \rho_j b_{ij} b_{Kj}, \qquad d_i = \sum_{j=1}^{m} \rho_j b_{ij} y_j \ .$$

$$V_n^{(2)}(1,\alpha)/S_{n-1}(1) = \sum_{i=0}^{k} (-1)^t \left[\binom{k}{t} / 2t+1 \right] \left(1 - \cos^{2t+1}(1/2\alpha)\right);$$

b) $n = 2k, \quad k > 0$,

$$\frac{V_n^{(2)}(1,\alpha)}{S_{n-1}(1)} = \frac{(2k-1)!!}{(2k)!!}(1/2\alpha) - \frac{\cos(1/2\alpha)\sin^{2k-1}(1/2\alpha)}{2k-1} -$$

$$-\sum_{t=1}^{k-1} \prod_{r=1}^{t} (n-2r+1) \left(\prod_{r=0}^{t} (n-2r+2)^{-1} \cos(1/2\alpha) \sin^{n-2t-1}(1/2\alpha) \right)$$

Now the condition (11) takes the following form:

$$\frac{V_n^{(1)}(1,\alpha) + V_n^{(2)}(1,\alpha)}{S_n(1)} = \frac{V_n^{(1)}(1,\alpha)/S_{n-1}(1) + V_n^{(2)}(1,\alpha)/S_{n-1}(1)}{V_n^{(2)}(1,2\pi)/S_{n-1}(1)} \geq p \qquad (12)$$

From (12) it is easy to find the minimum value of the cone opening angle of successful directions $\alpha \ (0 \leq \alpha \leq \pi)$. Knowing the corner α out of the expression (11) we find the step $a = 2\mu\cos(1/2\alpha)$.

Thus, the algorithm involves a number of following steps:

1. Is selected the starting point of the search $\mathbf{X}_0$, the initial value of the radius R_0, parameter β and the level of approximation error υ^*. In the R_0 – neighborhood of point $\mathbf{X}_0$ is scattered *m* trials, in which calculates the value of optimized function $f(\mathbf{X})$. Parameters points and values of the function at these points are remembered.
2. According to the information that we have is defind: vector $\mathbf{C}, \delta_x(\mathbf{X}_{N-1}, R_{N-1}, \beta)$, $(N \geq 1)$ and μ.
3. Is calculated the level of approximation error $\upsilon(R_{N-1}, \beta)$.

In case if $\upsilon(R_{N-1}, \beta) > \upsilon^*$ the radius R_N varies according to the following rule: $R_N = k_1 R_{N-1}$, where $0 < k_1 < 1$. After that, the search system returns to the point $\mathbf{X}_{N-1}$, are raffled the $m_1 < m$ trials in the R_N –vicinity of the point $\mathbf{X}_{N-1}$ and again are performed the operations are set out in Sec. 1 and Sec. 2.

4. Is searched the angle of opening the cone of successful directions α_{N-1} from the expression (11), and is performed a step in direction of antigradient:

$$\mathbf{X}_N = \mathbf{X}_{N-1} - \delta'_X(\mathbf{X}_{N-1}, R_{N-1}, \beta)\cos(1/2\alpha_{N-1})$$

5. If the value of objective function at the new point $f(\mathbf{X}_N) < f(\mathbf{X}_{N-1})$, the system moves to the point R_N [7]:

$$R_N = k_1 R_{N-1} + k_2 \|\mathbf{X}_N - \mathbf{X}_{N-1}\|, \quad (k_2 \geq 0, \ \ k_1 + k_2 \geq 1) \qquad (13)$$

If $f(\mathbf{X}_N) \geq f(\mathbf{X}_{N-1})$, it is believed $\mathbf{X}_N = \mathbf{X}_{N-1}$, is formed, according to (10), the value R_N, is restored partly the memory due to $m_1 < m$ test trials in the R_N – neighborhood of point X_N, then are repeated the operations with of Sec. 2.

Searching stops if $\|\delta'_X(\mathbf{X}_{N-1}, R_{N-1}, \beta)\| < \varepsilon, \ \ (\varepsilon > 0)$. The proposed condition is working very reliably in the case of not ravine's functions. In gully situation the stop criterion can bring to the fact that search system will stop at the bottom of the ravine. For ravine's functions suggest the following algorithm.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

VI. The Algorithm Of Local Search With The Approximating Elliptical Hyper-Paraboloid

Algorithm №2. We will produce an approximation of the increment function $f(\mathbf{X})$ in some orthonormal basis of the space $G=(g_1,g_2,....,g_n)^T$, which is obtained by rotating the axis of the main base $B=(b_1,b_2,....,b_n)^T$. In basis G the approximating function is defined as:

$$\delta(\mathbf{X},R,\beta)=\sum_{i=1}^{n}\left[c_i\left(x_i-x_i^{(0)}\right)^2+2c_{i+n}\left(x_i-x_i^{(0)}\right)\right] \quad (14)$$

Using the method of least squares at $m>2n$ for the calculation of the components of the vector C we get the parametrical gradient at the point $\mathbf{X}_0$:

$$1/2\delta_X^{'}(\mathbf{X}_0,R,\beta)=(c_{n+1},c_{n+2},....,c_{2n})^T.$$

For the central functions (14), the vector indicating the direction to the center, is as follows:

$$d(\mathbf{X}_0,R,\beta)=(-c_{n+1}/c_1,-c_{n+2}/c_2,-c_{2n}/c_n)^T$$

And the distance to the center $\mu=\left\|d(\mathbf{X}_0,R,\beta)\right\|$.

When building a matrix $\Gamma(\mathbf{X}_{N-1})$, describing the position of the axes of the rotated basis, the method of orthogonalization of the system of linearly independent vectors is used. The proposed version of the algorithm has the structure:

$$\mathbf{X}_N=\mathbf{X}_{N-1}+\gamma_{N-1}\Gamma(\mathbf{X}_{N-1})d(\mathbf{X}_{N-1},R_{N-1},\beta_{N-1}),\quad(\gamma_{N-1}\le 1) \quad (15)$$

And can be described by the following scheme.

1. Are performed the operations set out in Sec 1 of algorithm 1, and is accepted , that $\Gamma(\mathbf{X}_0)=\mathbf{B}$, where $\mathbf{B}$ – the unit matrix.
2. Is calculated the vector $\mathbf{C}$, direction $d(\mathbf{X}_{N-1},R_{N-1},\beta)$ and μ.
3. According Sec. 3 of algorithm 1, is checked the condition to achieve the required accuracy of approximation.
4. Is fulfilled a step, according to (15). At this

$$\gamma_{N-1}=\begin{cases}1 & \text{if}\quad \mu\le R_{N-1}\\ \dfrac{R_{N-1}}{\mu} & \text{if}\quad \mu>R_{N-1}\end{cases}.$$

5. Are performed operations set forth in Sec. 4 of algorithm 1.
6. We rotate the first axis of basis G:

$$g_1^{(N)}=\left(k_3g_1^{(N-1)}+k_4\frac{\mathbf{X}_N-\mathbf{X}_{N-1}}{\left\|\mathbf{X}_N-\mathbf{X}_{N-1}\right\|}\right)\left(\left\|k_3g_1^{(N-1)}+k_4\frac{\mathbf{X}_N-\mathbf{X}_{N-1}}{\left\|\mathbf{X}_N-\mathbf{X}_{N-1}\right\|}\right\|\right)^{-1}.$$

Here $k_3,k_4>0$ – coefficients, which are selected depending on the rate of change of direction of gradient along the bottom of the ravine and which determine the inertness of the basis G in the search process. The axes $g_K^{(N)}(K=\overline{2,n})$ are built by the orthogonalization method [8] out of $g_K^{(N-1)}$.

7) The procedure is repeated from Sec. 2. The search stops when

$$\{\delta(\mathbf{X}_{N-1},R_{N-1},\beta)-\delta[\Gamma(\mathbf{X}_{N-1})d(\mathbf{X}_{N-1},R_{N-1},\beta),R_{N-1},\beta]\}\mu^{-1}\le\varepsilon.$$

For the experimental study of losses in the process of search using the proposed algorithms were performed the computer calculations on the test functions.

At this the proposed algorithms were supplemented by the procedure of steepest descent along the successfully selected directions.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

With the help of the algorithm №1 was found extremum of the function:

$$f(X) = 0{,}1 \cdot (0{,}8x_1 + 0{,}6x_2)^2 + 2 \cdot (0{,}8x_2 - 0{,}6x_1)^2 .$$

As a starting points were selected points $x_1^{(0)} = 10,\ \ x_2^{(0)} = 10,\ \ R_0 = 5$. The search was finished, when $\delta_X \left(X_{N-1}, R_{N-1}, \beta\right) < 0{,}01$.

The Table 1 shows the results of calculations and losses on search for different values of parameters of algorithm (υ^* – the level of error; m_1 – the number of test trials).

Table 1
The results of calculations and losses on search for different values of the parameters of the algorithm №1

υ^*	m_1	The average number of attempts $\delta(\mathbf{X}, R, \beta)$	The average losses on search
		By 10 implementations	
1	2	46	118
1	4	33	134
1	6	42	267
1	8	51	366
1	10	53	384
1	12	64	481
0,5	2	63	154
0,25	2	76	171

The algorithm №2 was tested on the ravine function:

$$f(x) = 100\left(x_1^2 - x_2^2\right)^2 + \left(x_1 - 1\right)^2$$

As the initial points were selected points $x_1^{(0)} = -1$; $x_2^{(0)} = 1$; $x_1^{(0)} = -1{,}2$; $R = 0{,}2$. The results of calculations and comparative comparison with the results of calculations using of known of ravine algorithms [9] are shown in Table 2. Both algorithms were investigated at $\beta = 100,\ \ A_1 = 10$.

Table 2
The results of verification of algorithm №2 on test example

Algorithm №2						The results after 200 steps	
						Configuration method	The method of rotating coordinates
m at $m_1 = 2$	10	15	25	35	25	-	-
$x_1^{(0)}$	-1	-1	-1	-1	-1,2	-1,2	-1,2
$x_1^{(0)}$	1	1	1	1	1	1	1
$x_1^{(0)}$	1,0002	1,017	1,0012	1,010	1,002	-	0,995
$x_1^{(0)}$	1,005	1,023	1,0008	1,006	1,004	-	0,991
$f(\mathbf{X}^*)$	$1{,}8 \cdot 10^{-4}$	$3{,}0 \cdot 10^{-4}$	$0{,}1 \cdot 10^{-4}$	$1{,}2 \cdot 10^{-4}$	$0{,}6 \cdot 10^{-4}$	$1{,}03 \cdot 10^{-2}$	$0{,}22 \cdot 10^{-4}$
The number of approximations	30	26	22	23	38	-	-
The number of attempts	176	152	138	142	203	-	-

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 3, March 2017)

Let us investigate issues of convergence of algorithms using the smoothing operators.

To this end, consider a systems of functions $\varphi(\mathbf{X})$ comprising the increment of variables till a second order of type:

$$\varphi_1(\mathbf{X}) = \|\mathbf{X}-\mathbf{X}_0\|^2;\ \varphi_{j+1}(\mathbf{X}) = \mathbf{X}_j - \mathbf{X}_j^0,\ \ (j=1,2,...,n), \tag{16}$$

$$\varphi_1(\mathbf{X}) = (\mathbf{X}_j - \mathbf{X}_j^0)^2;\ \varphi_{n+1}(\mathbf{X}) = \mathbf{X}_j - \mathbf{X}_j^0,\ \ (j=1,2,...,n), \tag{17}$$

$$\varphi_1(\mathbf{X}) = \sum_{i=1}^{n} \alpha_i (\mathbf{X}_j - \mathbf{X}_j^0)^2 \tag{18}$$

Let us introduce the designation $\nabla\omega(\mathbf{X}_0, R) = \nabla f(\mathbf{X}_0) - \delta(\mathbf{X}_0, R)$ and show that for smooth functions the absolute gradient approximation error $\nabla f(\mathbf{X})$ depends continuously on the parameter R.

Theorem. Let for all there is $\lim\limits_{R\to 0}\|\nabla\omega(\mathbf{X}_0, R)\| = 0$

Evidence. We write the expansion $f(\mathbf{X})$ in the row of Taylor in the neighborhood

$$\nabla f(\mathbf{X}) = \langle \nabla f(\mathbf{X}_0), \mathbf{X}-\mathbf{X}_0 \rangle + L_\varepsilon(\mathbf{X})\|\mathbf{X}-\mathbf{X}_0\|^2,\ \ |\varepsilon(\mathbf{X}) \le 1|$$

For simplicity, we assume that the system has the form (16) Then the coefficients $C_{J+1}(\mathbf{X}_0, R)$ are determined from the normal equations of the least squares method.

$$C_{J+1}(\mathbf{X}_0, R) = \frac{\int_{E_n} \rho(\boldsymbol{\eta}-\mathbf{X}_0, R)(\eta_j - x_j^0)\Delta f(\boldsymbol{\eta})d\boldsymbol{\eta}}{\int_{E_n} \rho(\boldsymbol{\eta}-\mathbf{X}_0, R)(\eta_j - x_j^0)d\boldsymbol{\eta}},\ \ (j=1,2,...,n)\ . \tag{19}$$

Substituting in (19) and taking into account the orthogonalization of functions (16), we have:

$$C_{J+1}(\mathbf{X}_0, R) = \frac{\partial F(\mathbf{X}_0)}{\partial x_j} + L\frac{\int_{E_n} \rho(\boldsymbol{\eta}-\mathbf{X}_0, R)\varepsilon(\boldsymbol{\eta})\|\boldsymbol{\eta}-\mathbf{X}_0\|(\eta_j - x_j^0)d\boldsymbol{\eta}}{\int_{E_n} \rho(\boldsymbol{\eta}-\mathbf{X}_0, R)(\eta_j - x_j^0)^2 d\boldsymbol{\eta}},\ \ (j=1,2,...,n)\ . \tag{20}$$

At small value R the second term on the right-hand side of (20) - the value of the order R. Therefore, when $R \to 0$ we get $C_{j+1}(\mathbf{X}_0, R) \to \frac{\partial F(\mathbf{X}_0)}{\partial x_j}$ and therefore $\|\nabla\omega(\mathbf{X}_0, R)\| \to 0$. Similarly, it can be proved the theorem for systems of functions (17) and (18).

Corollary. $\|\nabla\omega(\mathbf{X}_0, R)\| \to 0$ evenly by $\mathbf{X}_0 \in \omega(\bar{a})$ when $R \to 0$. This follows directly out of (20) and out of compactness $\omega(\bar{a})$.

REFERENCES

[1] Rastrigin L.A., Ripa K.K. Automaton theory of random search. Riga: Zinatne, 1973. 343p.

[2] Zipkin J. Z. , Polyak B.T. Pseudo-gradient adaptation and learning algorithms // Automation and Remote Control. − 1973. №3. P.48-56.

[3] Katkovnik V.Y. The task of approximation of functions of several variables // Automation in the Remote Control. − 1971. № 2. P.181-185.

[4] Volynskij E.I., Poteshkina D.I., Filatov G.V. Application some of smoothing operators in the tasks of search of extremum // News. Acad. Science SSSR, Technical cybernetics. − 1975. №2. P.90-95.

[5] Fiakko A., McCormick G. Nonlinear Programming. Methods consistent unconditional optimization. − M .: Mir, 1972. 240 p.

[6] Antonov G.E., Katkovnik V.Y. Method of synthesis of a class of random search algorithms // Automation and telemehanika. − 1971. №6. P.154-157.

[7] Rastrigin L.A. Statistical methods of search. − M .: Science, 1968. 376 p.

[8] Gantmakher F.R. Theory of matrices. − M.: Science, 1967. 576 p.

[9] Rastrigin L.A., Tarasenko G.S. On one algorithm of random search // Problems of random search. − Riga. − 1974. − Issue.3. P.108-112.