

PAPER • OPEN ACCESS

## Justification and development of an adaptive algorithm for stochastic optimization of a multi-criteria process

To cite this article: Vladimir Kulikov 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **1001** 012073

View the [article online](#) for updates and enhancements.

You may also like

- [A Critical Review on Nature Inspired Optimization Algorithms](#)  
Vishnu Soni, Abhay Sharma and Vijander Singh
- [Sequential methods and algorithms](#)  
Igor Deshko and Victor Tsvetkov
- [Denni Algorithm An Enhanced Of SMS \(Scan, Move and Sort\) Algorithm](#)  
Denni Aprilsyah Lubis, Opim Salim Sitompul, Marwan et al.



**ECS**  
The  
Electrochemical  
Society  
Advancing solid state &  
electrochemical science & technology

**DISCOVER**  
how sustainability  
intersects with  
electrochemistry & solid  
state science research

# Justification and development of an adaptive algorithm for stochastic optimization of a multi-criteria process

Vladimir Kulikov<sup>1</sup>

<sup>1</sup>Moscow State University of Civil Engineering, 26, Yaroslavskoye shosse, Moscow, 129337, Russia

E-mail: KulikovVG@mgsu.ru

**Abstract.** The paper considers aspects of the development of algorithms for optimizing complex systems, and also develops an algorithm for solving problems of convex stochastic programming with a non-smooth goal function. The principles of constructing adaptive procedures for adjusting the parameters of variable gradient optimization algorithms are proposed. Note that in almost any iterative algorithm, as a rule, there are parameters that require their adjustment. To control and adjust parameters, - criteria are formed that determine the quality of adjustments. At the same time, the problem of determining the best value of the adjustment parameters belongs to the same class as the original optimization problem. Usually, iterative algorithms that work in the same class of optimization problems are used to adjust the available parameters. Thus, it turns out that two algorithms work simultaneously, in the source space and in the parameter space. Since algorithms are adapted by parameters during operation, this type of algorithm is called adaptive. The current stage of development of computer technology and mathematical support requires the development of algorithms that must function successfully without the user's participation, both in the process of solving the problem and in the process of finding a solution to the problem. The paper provides recommendations for software implementation of adaptive stochastic algorithms and construction of computational procedures for stochastic computational experiments based on them.

## 1. Subject of research

Adequate modeling of complex systems involves taking into account their nonlinear, stochastic structure. This provision is necessary in the process of modeling systems, when the "randomness" seems to be dependencies in the simulated system, such as: - productivity-weather, electric power stations, solar activity, the adaptability of manufacturing operations - the volume of replacement jobs, needs picking product from the direct presence or absence of the desired product production in a given time, etc.

We refer the proposed algorithm to direct algorithms for optimization of stochastic systems, since they do not require translation of the stochastic problem to a deterministic equivalent. The problem is solved without the specified intermediate stage.

Difficulties. Numerical methods for optimizing stochastic systems involve calculating two functionals, the goal functional and the constraint functional, which are usually represented by integrals - the multiplicity of which can reach very large values determined by the decision-maker (LPR), taking into account the computational tools available to the LPR.



Features. The problem of stochastic programming with incomplete information about the goal function, existing constraints, and their combinations is considered.

Idea. The implemented algorithm uses random directions of stochastic gradients instead of exact values of gradients, as probabilistic estimates of their vectors. The problem of minimizing a convex function  $f(x)$  is presented as follows:

$$f(x) \rightarrow \min, x \in X \quad (1)$$

where  $X$  – is a convex set in Euclidean space  $R^n$ .

If the function  $f(x)$  gradients as:

$$f(x) = M\varphi(x, \omega) = \int \varphi(x, \omega)P(d\omega) \quad (2)$$

is convex in  $X$ , then the following condition must be met for the differential of this convex function:

$$\partial f(x) = \int \partial \varphi(x, \omega)P(d\omega) \quad \partial f(x) = \int \partial \varphi(x, \omega)P(d\omega) \quad (3)$$

If this is the case, then  $\partial \varphi(x, \omega)$  – can be represented by a set of vectors that are probabilistic estimates of the generalized gradient of the function  $f(x)$ .

On the convergence of the algorithm. By the algorithm we mean the rule for constructing a sequence of points  $\{x^s\}$  that belong to the set  $X \in R^n$ .

Conditions for convergence of the algorithm. Consider some set of initial solutions  $X^* \in X$ . We will consider the algorithm convergent if the following condition is met:

$$\lim_{s \rightarrow \infty} d(x^s, X^*) = 0, \text{ where } d(x^s, X^*) = \inf_{x \in X} |x^s - x| \quad (4)$$

From (4) follows if we consider the problem of linear programming:

$$\min f(x), x \in X \in R^n \quad (5.1)$$

the concept of the algorithm will match the following rule:

$$x^{s+1} = x^s + \theta_s \cdot (x^0, \dots, x^s), s = 0, 1, \dots, \quad (5.2)$$

solving the problem 5.1.

As the set  $X^*$  can be taken multiple solutions to problem (5.1), or a set of the necessary conditions of extremum in this task.

In this case, the statement about the convergence of the algorithm will be a statement about the fulfillment of the necessary extremum conditions at all limit points of the sequence  $\{x^s\}$ .

Let's clarify the conditions that may imply convergence of the algorithm.

Assume that the sequence  $\{x^s\}$  and the set of solutions  $x \in X \in R^n$  are such that the following conditions hold:

1.  $\{x^s\} \in X$ , where  $X$  is a compact subset of  $R^n$ ;
2.  $W: X \rightarrow R$  is a continuous function;
3. If the sequence  $\{x^s\}$  converges at  $x'$ , such that  $d(x', X^*) > 0$ , then for any  $\xi > 0$ , there is a subset of the index sequence  $\{i_k\}$  for which  $W(x^\tau) \leq W(x') + \xi$  for  $i_k \leq \tau \leq i_{k+1}$  and  $\lim_{k \rightarrow \infty} W(x^{i_k}) = W^- < W(x')$ ;
4.  $(W^-, W(x')) \setminus W(X^*) = \emptyset$ , i.e. the open interval  $W^-, W(x')$  is not contained on the set  $W(X^*) = \Delta \{W(x^*): x^* \in X^*\}$ ;
5. If a subset of the sequence  $\{x^{s_k}\}$  converges at a point  $x^*$  such that  $d(x^*, X^*) = 0$ , then

$$(W(x^{s_k+1}) - W(x^{s_k})) + \Delta \max\{0, W(x^{s_k+1}) - W(x^{s_k})\} \rightarrow 0 \quad (6)$$

when  $k \rightarrow \infty$ .

Generalization of the idea. Suppose that the LPR needs to select a resource (stock) of the product of production  $x$ . Let the demand for the resource be given by a random variable  $\theta$ . If the demand is

greater than the supply, i.e.  $x < \theta$ , then we assign a penalty equal to:  $b \cdot (\theta - x)$ . And if  $x > \theta$ , then the penalty will be defined as  $a \cdot (x - \theta)$ . In both cases,  $a$  and  $b$  are positive penalty values.

We write the target function as:

$$f(x) = \text{Mmax}\{a \cdot (x - \theta), b \cdot (\theta - x)\} \quad (7)$$

random function

$$\varepsilon(x) = \begin{cases} a, & \text{if } x \geq \theta, \\ -b, & \text{if } x < \theta \end{cases} \quad (8)$$

let's represent it as a stochastic gradient.

An unknown minimum point of a convex function  $f(x)$  on a set  $X$  is evaluated by a recurrent sequence

$$x^{s+1} = \pi_X \cdot (x^s - \rho_s \cdot \varepsilon^s), s=0, 1, \dots, \infty. \quad (9)$$

where  $s$  - is the iteration number of the algorithm,  $X$  - is a convex closed set in  $R^n$ ,  $\pi_X$  is a projection operation on the set  $X$ ,  $\varepsilon^s$  - is a stochastic gradient, the conditional expectation  $M_s$  of this vector is relative to the  $\sigma$ -algebra  $F_s$ , given by random vectors  $(x^0, \varepsilon^0, x^1, \varepsilon^1, \dots, x^s)$ , which satisfies the condition:

$$M_s \varepsilon^s = \nabla f(x^s) + b^s \quad (10)$$

$\rho_s$  - a sequence of random variables, step multipliers;  $b^s$  - a sequence of random vectors.

For the  $\rho_s$  steps, we suggest choosing a pre-defined sequence  $\{a_k\}$  that satisfies the following conditions:

$$\sum_0^\infty a_k = \infty, a_k > 0, k=0, 1, \dots \quad (11)$$

We outline a way to minimize a convex, non-smooth function  $f(x)$  on a convex compact subset  $X$  of the space  $R^n$ . The function  $f(x)$  has known stochastic gradients. Let all the random variables used be defined on the probability space  $(\Omega, A, P)$ . The algorithm will generate sequences of random directions  $d^s$  and points  $x^n \in R^n$  ( $s=0, 1, \dots$ ) based on the following relations:

$$d^n = \frac{(\varepsilon^s + i_s \cdot \gamma_s \cdot d^{s-1})}{(1 + \gamma_s)} \quad (12)$$

$$x^{s+1} = \begin{cases} \pi_X \cdot (x^s - \rho_s \cdot (1 + \gamma_s) \cdot d^s), & \text{if } \rho_s \cdot (1 + \gamma_s) \cdot |d^s| \leq t \\ \pi_X \cdot \left(x^s - t \cdot \frac{d^s}{|d^s|}\right), & \text{if } \rho_s \cdot (1 + \gamma_s) \cdot |d^s| > t \end{cases} \quad (13)$$

where  $\varepsilon^s$  - stochastic gradient, i.e.  $M[\varepsilon^s | F_s] \in \partial f(x^s)$ , the  $\sigma$ -algebra generated by  $F_s$  random variables  $(x^0, \dots, x^s, \varepsilon^0, \dots, \varepsilon^{s-1})$ ;  $\rho_s$  - positive step multiplier;  $\gamma_s$  - positive averaging factor;  $i_s \in \{0, 1\}$  - reduction factor;  $t_s \in \{0; +\infty\}$  is the bounding constant.

At the starting point  $x \in X$ , we assume  $d^1=0$ . In accordance with (12), it follows that the direction  $d^s$  is a convex combination of the zero vector and the previous stochastic gradients  $\xi^i$  ( $i=0, \dots, s$ ).

The recovery factor  $i_s$  defined as follows:

$$i_s \in \{0, 1\} \text{ if } |\varepsilon^{s-1}| \leq \sigma \quad (14)$$

$i_s = 0$ , если  $|\xi^{s-1}| > \sigma$

where  $i_s$  a fixed threshold value.

To construct recurrent relations for the modification of parameters  $\rho_s, \gamma_s$  we assume that the algorithm operates in the allowed range  $X$  and  $t = +\infty$ . For data  $x^{s-1}, d^{s-2}$ , and  $\lambda \geq 0$ , we consider a function that characterizes the quality of the selected parameters  $\rho$  and  $\gamma$ .

$$\varphi_s(\rho, \gamma) = f(x^s \cdot (\rho, \gamma, \varepsilon^{s-1})) - f(x^{s-1}) + \frac{1}{2} \cdot \lambda \cdot |x^s \cdot (\rho, \gamma, \varepsilon^{s-1}) - x^{s-1}|^2 \quad (15)$$

where -  $x^s \cdot (\rho, \gamma, \varepsilon^{s-1}) = x^s - \rho \cdot (\varepsilon^{s-1} + i_{s-1} \cdot \gamma \cdot d^{s-2})$  are determined by the relations (12,13). The values of  $\rho_{s-1}$  and  $\gamma_{s-1}$  must be chosen from the minimality condition of the function  $\Phi_s(\rho, \gamma) = E_{s-1}\varphi_s(\rho, \gamma)$ .

Differentiate the function  $\varphi_s(\rho, \gamma)$ . at the point  $\rho_{s-1}$  and  $\gamma_{s-1}$ , and we reduce the expression to the form:

$$\partial\varphi_s(\rho_{s-1}, \gamma_{s-1}) = \{(\hat{u}, \hat{c}): \hat{u} = \frac{1}{\rho_{s-1}} [ < g^s, \Delta x^s > + \lambda \cdot |\Delta x^s|^2 ] \quad (16)$$

$$\hat{c} = \frac{\rho_{s-1} \cdot i_{s-1}}{\rho_{s-1} \cdot (1 + \gamma_{s-1})} \cdot [ < g^s, \Delta x^{s-1} > + \lambda \cdot < \Delta x^s, \Delta x^{s-1} > ], g^s \in \partial f(x^s) \quad (17)$$

where  $\Delta x^s = x^s - x^{s-1}$ .

Given the previous expressions

$$u_s = < \varepsilon^s, \Delta x^s > + \lambda \cdot |\Delta x^s|^2, \quad (18)$$

$$c_s = i_{s-1} \cdot (< \varepsilon^s, \Delta x^{s-1} > + < \Delta x^s, x^{s-1} >) \quad (19)$$

get:

$$M_s = \left( \frac{\frac{1}{\rho_{s-1}} u_s}{\frac{\rho_{s-1}}{\rho_{s-2} \cdot (1 + \gamma_{s-2})} c_s} \right) \in \partial\Phi_s(\rho_{s-1}, \gamma_{s-1}) \quad (20)$$

Based on the above, the vector  $(u_s, c_s)$  can be interpreted as a stochastic gradient of the function  $\Phi$  at the point  $(\rho_{s-1}, \gamma_{s-1})$  up to positive factors.

Thus, the use of the vector  $(u_s, c_s)$  for constructing recurrent relations for calculating the step factor  $\rho_0 > 0$  is justified.

$$\rho_s = \min\{\rho^-, \rho_{s-1} e^{\{\min(\eta, -\alpha \cdot u_s - j_s \cdot \delta \cdot \rho_{s-1})\}} \quad (21)$$

where -  $\rho^- > 0, \eta > 0, \alpha > 0, \lambda \geq 0, \delta > 0$  – are fixed parameters, and the coefficient  $j_s$  is defined as follows:

$$j_s \in \{0, 1\}, \text{ if } |\Delta x^s| \geq \Delta_{min} \quad (22)$$

$$j_s = 1, \text{ if } |\Delta x^s| < \Delta_{min} \quad (23)$$

$\Delta_{min}$  is a small positive value.

The ratio for calculating the averaging coefficients  $\gamma_{s1}$  is represented as follows:

$$\begin{cases} \gamma_0 = \gamma_1 > 0 \\ \gamma_s = \min\{\gamma^-, \gamma_{s-1} e^{\{-\beta \cdot \vartheta_s - j_s \cdot \aleph \cdot \gamma_{s-1}\}} \end{cases} \quad (24)$$

where -  $\gamma^- > 0, \aleph > 0$ .

In the relations (23,24), the terms  $(j_s \cdot \delta \cdot \rho_{s-1})$  and  $(j_s \cdot \aleph \cdot \gamma_{s-1})$  increase the rate of decrease of the coefficients  $(\rho_s, \gamma_s)$  if the values  $u_s$  and  $\vartheta_s$  are close to zero.

A note on generalization. The previous arguments justify the following assumptions regarding the parameters of the developed algorithm and the function  $f(x)$ :

1. Condition for the control coefficient  $\lambda$  and the convexity constant of the function –  $v$ :

$$\lambda + v > 0 \quad (25)$$

where –

$$v = \sup\{\mu: f(y) \geq f(x) + < g, y - x > + \mu \cdot |y - x|^2\} \quad (26)$$

for all  $x, y \in X$  and all  $g \in \partial f(x)$ .

It follows from (25) that in the case of a strong convexity of the objective function, the control coefficient  $\lambda$  in (15) can be assumed to be zero.

2. There are constants  $q_0$  and  $Q$  such that

$$M_s e^{<z, r^s>} <0 \quad s=0,1, \dots \quad (27)$$

For any  $z \in \mathbb{R}^n$  such that  $\|z\| \leq q_0$ , where

$$r^s = \varepsilon^s - M_s \cdot \varepsilon^s \quad (28)$$

Condition (27) is related to the exponential form of relations (21, 23). Condition (27) holds for both bounded and unbounded distributions.

Problems with probabilistic estimates of gradients of objective functions arise when minimizing functions of the following type:

$$f(x) = M \cdot \varphi(x, \omega) = \int_{\omega \in \Omega}^{\gamma_l} \varphi(x, \omega) \cdot P(d\omega) \quad (29)$$

Since, under any General assumptions, the generalized differential of the function  $f(x)$  is calculated by the following formula:

$$\partial f(x) = \int_{\omega \in \Omega}^{\gamma_l} \partial_x \varphi(x, \omega) \cdot P(d\omega) \quad (30)$$

then  $\partial_x \varphi(x, \omega)$  is the set of vectors that are probabilistic estimates of the gradients of the function  $f(x)$ .

The classical stochastic generalization problem (Weber's problem) is formulated as follows:

- given  $n$  points  $y_i (i=1, \dots, n)$  in two-dimensional Euclidean space  $\mathbb{R}^2$ , you need to find a point  $x \in \mathbb{R}^2$  such that the sum of the distances to all points  $y_i (i=1, \dots, n)$  is minimal. In the generalized statement, we assume that each point  $y_i (i=1, \dots, n)$  is a random variable defined by some probability measure  $\theta_i(y)$  on  $\mathbb{R}^2$ .

The problem is to find a point  $x \in \mathbb{R}^2$  that will minimize the weighted sum of mathematical expectations of distances from point  $x$  to points  $y_i (i=1, \dots, n)$ :

$$\Psi(x) = \sum_{i=1}^n \beta_i \cdot \iint_{\mathbb{R}^2} |x - y| \cdot \theta(dy) \rightarrow \min \quad (31)$$

If random variables  $y_i (i=1, \dots, n)$  have densities of distributions  $\theta_i (i=1, \dots, n)$ , then under certain conditions imposed on the function  $G(x) = \sum_{i=1}^n \beta_i \cdot \theta_i(x)$  the function  $\Psi(x)$  is strictly convex and twice differentiable.

Software implementation. Software implementation of algorithms requires the introduction of heuristic elements. Denote by  $I$  the unit matrix.

Algorithm. Set  $s=0$  at the beginning of the count, set  $x^0, \rho_0, P=n^{-1} \cdot I$ .

Step 1. Calculation of the stochastic gradient  $\xi^s$ ;

Step 2. If  $s=0$  go to step 10.

Step 3. Averaging the shift rate:

$$\Delta x^s = x^s - x^{s-1}, G_s = G_{s-1} + (|\Delta x^s| - G_{s-1}) \cdot D$$

At the beginning of the account,  $G_0=0$ .

Step 4. Checking the end of the invoice:

If  $G_s < G_*$  or  $s > s_*$ , then finish the count, otherwise go to the next step;

Step 5. The calculation of the scalar product,  $T_s$ :

$$T_s = -\langle \xi^s, \Delta x^s \rangle;$$

Step 6. Averaging the  $T_s$  module:

$$z_s = z_{s-1} + (|T_s| - z_{s-1}) \cdot D$$

At the beginning of the account,  $z_0=0$ .

Step 7.  $\rho_s$  step adjustment:

$$\rho_s = \rho_{s-1} \cdot R^{\frac{T_s}{z_s}} = \begin{cases} 1, & \text{if } T_s > 0 \\ U, & \text{if } T_s \leq 0 \end{cases}$$

Step 8. Checking the value of the step change  $\rho_s$ :

$$\rho_s = \begin{cases} \rho_{s-1} \cdot 3, & \text{if } \rho_s \cdot \rho_{s-1}^{-1} > 3, \\ \frac{\rho_s}{4}, & \text{if } \rho_s \cdot \rho_{s-1}^{-1} < 4^{-1}, \\ \rho_s & \text{in other cases;} \end{cases}$$

Step 9. Calculating  $P_s$  weights:

$$P^s = \begin{bmatrix} p_1(s) & 0 & \dots & 0 \\ 0 & p_2(s) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & p_n(s) \end{bmatrix},$$

$$P_i(s) = \alpha \cdot p_i(s-1) + \lambda_i(s) \cdot (1 - \alpha),$$

$$\lambda_i(s) = \begin{cases} 0, & \text{if } (x_i^{s-1} - x_i^s) \cdot \varepsilon_i^s \leq 0, \\ \frac{1}{m}, & \text{if } (x_i^{s-1} - x_i^s) \cdot \varepsilon_i^s > 0, \\ \frac{1}{n} & \text{in other cases;} \end{cases}$$

$m$  is the number of components for which  $(x_i^{s-1} - x_i^s) \cdot \varepsilon_i^s > 0$ ;

Step 10. Finding the following approximation:

$$x^{s+1} = x^s - \rho_s \cdot P^s \cdot \varepsilon^s;$$

Step 11. Design for a valid area:

$$x^{s+1} = \pi_x \cdot (x^{s+1});$$

Step 12. Go to step 1 by increasing  $s$  by one.

The algorithm implements two stop criteria: the first one is based on the number of iterations, and the second one is based on the average shift value. When the shift value becomes less than the threshold value  $G_*$ , the algorithm terminates (steps 3, 4).

Step adjustment includes the following points. The value of  $T_s$  in the exponent of  $R$  is normalized to a certain value of the module  $T_s$ . Therefore, if, for example,  $R=2$ , the step  $R=2$  will change (increase or decrease) due to the multiplier  $R^{T_s/Z_s}$  on average twice.

Instead of the step reduction parameter  $\delta$ , an additional step reduction is used using the coefficient  $U$ ,  $0 < U \leq 1$ .

Additional reduction occurs only when

$$T_s = -\langle \varepsilon^s, \Delta x^s \rangle \leq 0$$

Since  $T_s/Z_s$  in the indicator  $R$  is a random variable, the step  $\rho_s$  can both increase and decrease a very large number of times (step 7). In order for the next step to differ slightly from the previous one, the threshold coefficients for reducing and increasing the step multiplier are set (step 8).

The weight coefficients  $\rho_s, \dots, p_n(s)$  (step 9) remain positive, with  $\sum_{i=1}^n p_i(s) = 1$  for  $s = 0, 1, \dots$

Formulas for modifying the weights  $p_i(s)$  have the following meaning. If a shift occurs in one direction for any component, then the corresponding weight of this component increases, otherwise it decreases.

The design operation (step 11) is equivalent to solving a nonlinear programming problem. If  $X = \{x: A \cdot x \leq b\}$  ( $A$  is a matrix,  $b$  is a vector), then the implementation of the design operation is reduced to a quadratic programming problem.

Recommendations for selecting algorithm parameters.

1. The value of the average step change  $R$  ( $1 < R < 3$ ) is chosen to be equal to two.

2. Initially, the value of the initial step  $\rho_0$  does not significantly affect the convergence rate of the algorithm. However, if there is information, the initial value can be set as follows:  $\frac{|x^0 - x^*|}{|\varepsilon^0|}$ , where  $x^*$  is the extremum point.
3. The  $k$  parameter defines the averaging coefficient  $d = \frac{1}{k}$  in the averaging formulas (steps 3, 6). We recommend choosing  $k$  in the range  $4 \leq k \leq 5$  when the algorithm uses stochastic estimates of the gradients of the objective function, and  $k=1$  if the gradients are specified exactly.
4. Parameter  $U$ -additional step reduction coefficient-is selected in the range  $0,5 \leq U \leq 1$ .
5. If the information about gradients is probabilistic, then  $U$  is selected in the range  $0,8 \leq U \leq 1$ .
6. If the gradients are specified exactly,  $U = 0,6$ note that for  $k > 1$ the coefficient  $U$  can be equal to 1, since the step decreases quickly and without additional reduction.
7. The value of the average shift  $G_*$  in the stop criterion is set to the order of the required accuracy of the solution for the components  $x$ .
8. The  $s_*$  parameter allows you to finish the work after a certain number of iterations of the algorithm set in advance.
9. We recommend setting the parameter  $\alpha$  ( $0 < \alpha < 1$ ), (step9) to 0,5.

## References

- [1] McGinley, P. Decision analysis software survey. OR/MS Today 41 (5), 2014.
- [2] Podinovskij V.V. Vvedenie v teoriyu vazhnosti kriteriev v mnogokriterial'nyh zadachah prinyatiya reshenij: uchebnoe posobie. Fizmatlit, 2007.
- [3] French, S. and D.L. Xu. Comparison study of multi-attribute decision-analytic software. Journal of Multi-Criteria Decision Analysis 13, 2005:
- [4] Kushner H. Probabilistic approximation methods in stochastic control problems and the theory of elliptic equations. - Moscow: Nauka, 1985.
- [5] Kolmogorov A. N., Fomin S. V. Elements of the theory of functions and functional analysis. - Moscow: Nauka, 1981.
- [6] Nirenberg L. Lectures on nonlinear functional analysis. - Moscow: Mir, 1977
- [7] Moiseev N. N. Elements of the theory of optimal systems, Moscow: Nauka, 1975.
- [8] Varga, George. Optimal control of differential and functional equations. - Moscow: Nauka 1977.
- [9] Burkov V. N. Fundamentals of the mathematical theory of active systems, Moscow: Nauka, 1977.