Scientific Research

Solution Building for Arbitrary System of Linear Inequalities in an Explicit Form

Demetrius V. Shapot[*], Alexander M. Lukatskii

Energy Research Institute, Russian Academy of Sciences, Moscow, Russia
Email: [*]macro@eriras.ru

ABSTRACT

The known Fourier-Chernikov algorithm of linear inequality system convolution is complemented with an original procedure of all dependent (redundant) inequalities deletion. The concept of "almost dependent" inequalities is defined and an algorithm for further reducing the system by deletion of these is considered. The concluding algorithm makes it possible to hold actual-time convolution of a general inequality system containing up to 50 variables with the rigorous method of dependent inequalities deletion and up to 100 variables with the approximate method of one. The main application of such an approach consists in solving linear inequality system in an explicit form. These results are illustrated with a series of computer experiments.

1. Introduction

Polyhedron orthogonal projection (POP) was devised by Fourier [1] as early as in the 20s of the XIX century. Some properties of an arbitrary convex set orthogonal projection are considered by Shapot [2]. However, the closed algorithm of generating orthogonal projections to any subspaces is known only for linear polyhedral sets. For a long time the POP didn't find obvious practical use as with every elimination of variables it initiated a large number of dependent (redundant) inequalities. It resulted in a rapid increase in their total number and as a rule didn't make it possible to solve the problem within an acceptable timeframe. In the middle of the XX-th century Chernikov [3] devised methods of dependent inequality determination, making substantial progress towards resolving this problem. With their help it became possible to increase the dimension of the problem, solved with POP within acceptable time, from 5 - 8 to 8 - 15, however, a further increase of dimensions resulted in the former problems of expansion. Methods of dependent inequalities in large linear systems determination began to develop in the 80s (ref., for example, Bushenkov and Lotov [4], Lotov [5], Eremin and Makhnev [6]). A new constructive approach to the implementation of the Fourier-Chernikov algorithm (FCA), which makes it possible to control inequality number expansion during the

process of variable elimination, was devised by Shapot and Lukatskii [7]. From this point on we shall refer it to as the *constructive algorithm of convolution*—CAC.

Here we give a preliminary formulation of the main algorithm. Let the S_0 system be given, enclosing k of linear equations and m of linear inequalities, defining non-empty set in real space $\mathbf{R}^n$. Let us suppose that the S_0 investigation aim is to find its nonbasic variable population and to write each of them in an explicit form. Such a notation can be represented in two ways: 1) in the form of this variable equality to a number or to a linear function of numerically defined variables; 2) in the form of value bounds restricted by either numbers or by linear functions, which depend on previously numerically defined variables. Previously defined variable numerical values" should be considered only within the framework of a procedure containing the following two stages:

- Elimination of all equations from S_0 by means of "Step of Jordan Elimination" (JE) series. Then $r \leq k$ of variables will be assumed to be set equal to the linear functions, which depend on the rest $p = n - r$ variables, where $r = \text{rank}$ of the equation subsystem ($r = k$, if all equations are mutually independent). These variables will gain numerical values after solution of the remaining system of inequalities S_1 to $\mathbf{R}^p$. Let us suppose that *every inequality defines non-negativity of the corresponding basic variable.*
- S_1 "inequality system solution" is generated by an

investigator in the form of an arbitrary sequence of p variables, in which the first variable has one or two numerical boundaries and each of the consequent variables is limited by linear functions, which depend on previously considered variables. Such a pyramidal (S_2) representation of inequality system S_1 is similar to limit defining in multiple integrals. With such a representation it is quite convenient to select a working point in S_1, being guided by some conceptual criteria.

Let us consider an algorithm of S_2 sequence generation. Let us compare S_1 inequality system with the matrix M^p, which defines p-dimensional convex polyhedron. Let us decrease its dimension to $p-1$, *i.e.* let us pass to the matrix M^{p-1}, eliminating an arbitrary variable x_j from the coordinate set. This can be done in two ways: either by assigning a numerical value to x_j, *i.e.* by making S_1 section or by constructing the union of all sections along x_j, *i.e.* the S_1 orthogonal projection on subspace not containing x_j. We will use the second way of eliminating variables. Let us generate $\{M^{p-t}\}$ matrix sequence with $M^{p-1}, \cdots, M^{p-t}, \cdots, M^1$ polyhedra decreasing dimensions. Precisely this sequence will make it possible to easily generate S_2. Really the M^1 matrix contains only two columns, the first with the coefficients a_{uj} corresponds to x_j variable and the other one with the coefficients b_j—to free members.

Denote by $q_{11} = \min_u \left(-b_u/a_{uj} \mid a_{uj} < 0 \right)$ and $q_{12} = \max_u \left(-b_u/a_{uj} \mid a_{uj} > 0 \right)$.

Then it is not too difficult to see that $q_{12} \le x_j \le q_{11}$, where q_{11} and q_{12} are numbers. In the same way it is possible to represent the bounds for x_r variable using the M^2 matrix with the x_r and x_j variables, $q_{22} \le x_r \le q_{21}$, where q_{21} and q_{22} are linear functions, which depend on x_j. If you select x_j "attractive" value within the specified range, the x_r acceptable range will be a numerical one. This method of boundary function $\{M^{p-t}\}$ matrix construction and the selection of variable "attractive" values can be prolonged to the M^p parent matrix. From this point on we shall refer the considered approach to as POP.

Fourier algorithm is described in Section 2. Different methods of redundant inequality cleanup are considered and a formal description of POP is given in Section 3. The stability and complexity of POP algorithms are estimated in Section 4. The results of numerical experiments are discussed in Section 5.

2. Orthogonal Projection Fourier Algorithm

Let $\mathbf{R}^n$ be a real n-dimensional space. The set

$$P_k Q_n = Q_{n-k} : \begin{pmatrix} \forall x' \in Q_{n-k} \exists u' \in \mathbf{R}^k : (x', u') \in Q_n \\ \forall y' \notin Q_{n-k} \; \overline{\exists} v' \in \mathbf{R}^k : (y', v') \in Q_n \end{pmatrix}$$

is referred to as the $Q_n \subset \mathbf{R}^n$ set *projection* on the $\mathbf{R}^{n-k}$ subspace.

Let us consider the Q_n non-empty bounded set, defined by the linear system of inequalities

$$Q_n = \left\{ \sum_{i=1}^n a_{ji} x_i + b_j \ge 0, \left(j = 1, 2, \cdots m \right) = J \right\}.$$

Suppose we want to build $Q_{n-1} \subset \mathbf{R}^{n-1}$, where $\mathbf{R}^{n-1}$ doesn't depend on an x_r variable. According to the algorithm of linear inequality systems free convolution devised by Fourier [1] let us divide J index set into three groups:

$$Q^{(1)} = \left(\nu = 1, \cdots, s \right); Q^{(2)} = \left(\mu = s+1, \cdots, p \right);$$
$$Q^{(3)} = \left(\lambda = p+1, \cdots, m \right)$$

so, that for them $a_{\nu r} > 0; a_{\mu r} < 0; a_{\lambda r} = 0$.

Solving each of two first groups of inequalities with respect to x_r we obtain

$$\left\{ l_\nu = \frac{1}{a_{\nu r}} \left(-\sum_{i \ne r} a_{\nu i} x_i - b_\nu \right) \right\}$$
$$\le x_r \le \left\{ l_\mu = \frac{1}{|a_{\mu r}|} \left(\sum_{i \ne r} a_{\mu i} x_i + b_\mu \right) \right\}$$
$$\nu = 1, \cdots, s; \mu = s+1, \cdots, p \tag{1}$$

Eliminating x_r from these relations, *i.e.* combining each inequality of the first group with each inequality of the second group we obtain $P_1 Q_n = Q_{n-1}$, generated by $s(p-s)$ inequalities, which have the form

$$\left\{ \sum_{i \ne r} \left(\frac{a_{\mu i}}{|a_{\mu r}|} + \frac{a_{\nu i}}{a_{\nu r}} \right) x_i + \left(\frac{b_\mu}{|a_{\mu r}|} + \frac{b_\nu}{a_{\nu r}} \right) \ge 0 \right\}$$
$$\nu = 1, \cdots, s; \mu = s+1, \cdots, p \tag{2}$$

and $m - p$ inequalities of the third group.

As a matter of fact, $\forall x' \in Q_{n-1}$:

$$\alpha(x') = \max_{\nu=1,\cdots s} \left\{ l_\nu(x') \right\} \le \min_{\mu=s+1,\cdots k} \left\{ l_\mu(x') \right\} = \beta(x')$$

and then $\forall x_r \in \left[\alpha(x'), \beta(x') \right]$:

$$(x', x_r) \in Q_n \quad .$$

On the other hand, if x' doesn't comply with the (ν_0, μ_0)-th inequality of (2), then any x_r, complying with the ν_0 inequality of (1), contradicts the μ_0-th inequality and vice versa.

If the first or the second group of inequalities is empty, *i.e.* $\forall j \in J : a_{jr} \ge 0 \left(a_{jr} < 0 \right)$, then for any x', complying with inequalities of the third group, $\left(x' \in Q^{(3)} \right)$ and for any $x_r \ge \alpha(x') \left(x_r \le \beta(x') \right) (x', x_r) \in Q_n$. Therefore, in this case $P_1 Q_n = Q^{(3)}$.

With the help of the considered algorithm it is possible to eliminate any subset of variables and to construct the $P_k Q_n \subset \mathbf{R}^{n-k}$ orthogonal projection, where $1 \le k \le n$.

Since all computations are performed in the form of a matrix notation, the Q_n set is defined by the matrix, which we shall denote as M_0. Then the $P_k Q_n$, projection generated as a result of eliminating k variables, is defined by the M_k matrix. If $k = n - 1$, then the M_k contains only x_n column and the column of constant term. The (1) formula, which corresponds to it, has the form: $\underline{x}_n \leq x_n \leq \overline{x}_n$, where $\underline{x}_n = \max(b_1, b_2, \cdots, b_s)$, $\overline{x}_n = \min(b_{s+1}, b_{s+2}, \cdots, b_{s+p})$. Such a notation makes it possible to select any value of $x_n = x_n^0 \in [\underline{x}_n, \overline{x}_n]$ variable of the Q_n projection on the x_n axis, *i.e.* a permissible value. The same situation will occur, if in the M_{n-2} matrix you multiple the x_n column by x_n^0. And in general, during "the descent from the sequence of M_s matrices" after selection of the $x_s = x_s^0 : s = n - 1, n - 2, \cdots, n - k$ variables arbitrary permissible values and multiplying the M_k matrix corresponding columns by them we shall obtain the acceptable range for the x_{n-k-1} .variable value selection. Therefore, it is possible to say, that the $M_k : k = n - 1, n - 2, \cdots, 1$ matrix sequence allows us to obtain in explicit form the solution of the inequality system defined by Q_n.

The considered Fourier algorithm, in spite of its seeming simplicity, is usable only for the simplest systems of inequalities. The necessity of all $(\mu\nu)$-th pairs combining while eliminating each variable causes a rapid expansion of inequality number with elimination of variables. In particular, if after $k - 1$ variable elimination a system of r_{k-1} inequalities is obtained, then after k-th variable elimination we shall obtain $r_{k-1} - 1 \leq r_k \leq (r_{k-1}/2)^2$. The r_k lower boundary is implemented in the case if $p = m$ and $s = 1$, or $p - s = 1$, and the upper boundary if $p = m$ and $s = r_{k-2}/2$. Here a large percentage consists of generated inequalities are dependent. So, if $r_0 = 10$, then the accessible upper bound for r_5 is 2×10^{13} of inequalities.

Thus, intensive expansion of inequality number with elimination of variables is the main obstacle for POP use in problems of practical importance. To overcome this obstacle it is first of all necessary to have effective algorithms for determination of dependent inequalities generated while using the Fourier algorithm. From this point on we shall refer to them as "matrix cleanup" algorithms.

3. Methods of Eliminating Dependent Inequalities

3.1. Fourier-Chernikov Algorithm (FCA)

Motzkin *et al*. [8] and Chernikov [3] offered a serious improvement to the Fourier algorithm related to the abandonment of generating some dependent inequalities. By endowing each inequality of parent system with a primary subscript (number) and by joining (disjuncting) the subscripts while combining inequalities in pairs Cher-

nikov complemented the Fourier algorithm with the following two rules:

(**ChR₁**) With eliminating the h-th in succession inequality only those inequalities which generate the subscript containing not more than $h + 1$ of primary subscripts are to be combined (Chernikov's first rule);

(**ChR₂**) The pairs containing some other inequality total subscript shouldn't be combined in pairs (Chernikov's second rule).

In FCA (**ChR₁**) is verified during the process of Fourier inequalities combining and the (**ChR₂**) is verified after all combinations not contradicting (**ChR₁**) have been generated. With the help of Chernikov's rules all dependent inequalities are determined in the total polyhedra assembly, *i.e.* for all systems of homogeneous inequalities. However, for the inequalities with fixed constant terms these rules do not by any means determine all dependent inequalities.

The validity of this statement is readily illustrated by simple numerical examples. To generate them and to conduct consequent computational experiments we used general linear programming (LP) generator of inequality system:

$$y_j = \sum_{i=1}^{N} a_{ji} x_i + b_j \geq 0, \quad j = 1 \cdots N,$$

where

$$a_{ji} = \begin{cases} \dfrac{1}{3} \left(\dfrac{3i}{j+i} \right)^{0.5} \left((-1)^{\frac{j+i}{3} - 1} \right), & (i+j) \bmod 3 = 0; \\ 0 \end{cases}$$

$$b_j = \frac{\left(\dfrac{1}{3} + (-1)^j \right)}{j^{0.5}}, \quad i, j = 1, \cdots, N, \qquad (3)$$

$$\sum_{i=1}^{N} c_i x_i \to \max, c_i = \frac{(-1)^{(i+1)}}{i}, i = 1, \cdots, N.$$

The bounds of nonbasic variable are: $0.5 \leq x_i \leq 2$, $i = 1, 2, \cdots, N$.

In the first example for the case where $N = 7$ and using the Fourier algorithm without applying Chernikov's rules 587115 inequalities were generated with the elimination of the 7-th variable. In the case where $N = 8$ the use of the Fourier algorithm led to the memory overflow. In the second example where $N = 10$, application of Chernikov's rules, but without determination of all redundant inequalities, generated the results given in **Table 1**. But if you use the algorithm of dependent inequality complete cleanup in this example, convolution results will appear to be quite different. They are given in **Table 2**.

Thus, the prospects for practical use of convolution

Table 1. No determination of all redundant inequalities.

Iter. No.	1	2	3	4	5	6	7	8	9
Fourier	26	50	115	298	1466	8023	30,777	125,149	403,925
Chernikov	26	34	46	106	212	393	749	1335	2369

Table 2. With determination of all redundant inequalities.

Iter. No.	1	2	3	4	5	6	7	8	9
Fourier	19	25	16	11	10	7	8	10	3
Chernikov	19	18	13	11	10	7	8	8	3
Complete Cleanup	16	13	11	9	7	6	6	3	3

algorithms substantially depend on the possibility of determination of dependent inequalities generated in the framework of FCA. It should be pointed out, that Chernikov [3] also considered the method of all redundant inequalities with fixed constant terms complete cleanup. It is based on a computation of cone generator fundamental system associated with the system of linear inequalities. According to Preparata and Shamos [9], Shevchenko and Gruzdev [10] the corresponding algorithm has an operation number polynomial estimator on linear inequality system cardinality with a fixed dimension. In particular, if m is the inequality number (cardinality), n is the variable number (dimension), then the operation number in this algorithm has the estimate

$$O\left(m^{\left[\frac{n}{2}\right]+1}\right).$$ As the number of inequalities can increase

rapidly during the convolution process, such algorithm use does not seem to be of practical importance.

3.2. A Formalized Description of POP

Step 0. Clear the counter of variable eliminations $I = 0$.

Read an initial matrix M_p of the linear inequality system, where p —the number of variables.

Endow each linear inequality $l \geq 0$ a primary subscript $\text{ind}(l)$.

Step 1. Choose a column t, which has A positive and B negative elements, and multiplication $A \times B$ being minimal. If such a column is absent, then procedure is complete, otherwise $I = I + 1$.

Note, that under parallelepiped restrictions on all variables a premature end is possible only for an incompatible system, as otherwise there would be a non bounded variable, in the contrary system.

Step 2. Exclude the variable x_t by (1), (2) formula from the inequality system. As each new inequality is generated, let us form its subscript (ind) by uniting the subscripts of the corresponding pair. We do not include the inequalities being redundant with the 1-st Chernikov

rule, *i.e.* number of primary indexes (**ChR₁**) $(\text{length}(\text{ind}) > I + 1)$ don't include in the system.

Step 3. Carry out a control of subscript pairs on (**ChR₂**) ($ind' \subseteq ind''$), then delete the inequality with subscript ind'' from the system as a redundant one.

Step 4. Carry out after the full cleanup in the inequality system.

Step 5. Save M_{p-I} matrix I-th iteration of inequality system and the number $(p - I)_t$ of excluding variable. If $I \leq p - 1$, then return to Step 1, otherwise the end.

3.3. The Algorithm of Building of an Inequality System Solutions in Explicit Form

Here we give a formal description of the algorithm in 1.

This procedure means that all $p - 1$ steps of variable elimination have been implemented. Then the matrix M_{p-1} contains the single variable with number $t(1)$ and the matrix M_{p-j} contains j variables with numbers $t(1), \cdots, t(j)$. M_0 is the initial matrix of linear problem with p variables.

For $j = 1, \cdots, p$ do

Step j. If $j = 1$, then go to 2, otherwise:

1. In the matrix M_{p-j} we fix the values of variables, which were changed in the preceding steps

$$x_{t(1)} = x_{t(1)}^0, \cdots, x_{t(j-1)} = x_{t(j-1)}^0,$$

as a result the matrix M_{p-j} contains the single variable $x_{t(j)}$.

2. Using the matrix M_{p-j}, the solution of an inequality system with this single variable is formed as the range of its acceptable values: $A_j \leq x_{t(j)} \leq B_j$. We choose a desired value $x_{t(j)}^0$ from this range. If $j \leq p - 1$, then $j = j + 1$ and return to 1, otherwise the end.

3.4. Constructive Approach to Determination of Dependent Inequalities in Polyhedral Sets

The problem of dependent inequalities determination is rather relevant for various applications. It was considered by many authors both in the context of projections method Bushenkov and Lotov [4], Lotov [5], Eremin and Makhnev [6] and independently Efremov, Kamenev and Lotov [11], Golikov and Evtushenko [12], Gutman and Ioslovich [13], Paulraj and Sumathi [14]. Before we developed and programmatically implemented several approaches to the determination of dependent inequalities in linear systems Shapot and Lukatskii [7,15]. We describe and substantiate the algorithm of Shapot and Lukatskii [7] further in this text.

In spite of the fact that the problem of dependent inequalities determination after FCA algorithm iteration is considered, the devised algorithm is of a general nature and is not associated with the orthogonal projection spe-

cific procedure. To ensure investigated polyhedron boundedness we require that the upper and lower boundaries are defined for all free variables. These boundaries shall define the survey coverage of models in the form of a rectangular parallelepiped, of interest for the investigator. However, in the case when any of parallelepiped restrictions appears to be dependent it should be eliminated from the corresponding matrix like any other dependent inequality. Here we remind the dependent inequality concept.

Definition 3.1. *The inequality $l(x) \geq 0$ is dependent* (*redundant*) *on a set Q, if it is satisfied in each point of this set and is incompatible with Q, if it is satisfied in no point $x \in Q$.*

The analysis of $l(x) \geq 0$ inequality dependence or incompatibility conditions with respect to the Q set can be founded on any of the following equivalent and obvious criteria.

(**Crit₁**) *In order for the $l(x) \geq 0$ inequality could be*:
a) *dependent on Q, or*
b) *incompatible with Q,*
It is necessary and sufficient to meet the conditions:
a) $\min_{x \in Q} l(x) \geq 0,$
b) $\min_{x \in Q} l(x) < 0.$ *r*

(**Crit₂**) *The $l_v(x) \geq 0$ inequality is dependent on*
$Q_v = \left\{ l_j(x) \geq 0 \,\middle|\, j \neq v \right\}$ *if and only if*
It is representable as $l_v = \sum_{j \neq v} u_j l_j + a_v, \, u_j \geq 0, \, a_v \geq 0$
Chernikov [3].

(**Crit₃**) *a*) *Elimination of any independent inequality from the system of inequalities causes region of feasibility expansion, i.e. $Q \neq Q_v$ and $Q \subset Q_v$;*
b) *Elimination of any dependent inequality from the system of inequalities doesn't cause region of feasibility expansion, i.e. $Q = Q_v$.*

In the general case, when the $l_j(x) \geq 0$ inequality dependence on the

$$\bar{Q}_j = \left\{ \begin{array}{l} \underline{x}_i \leq x_i \leq \bar{x}_i, \, i = 1, \cdots, n, \\ \\ l_v(x) = \sum_{i=1}^{n} a_{vi} x_i + b_v \geq 0, \, v = 1, \cdots, m; \, v \neq j \end{array} \right\},$$

set is investigated it is sufficient to use (**Crit₁**) and to solve the corresponding problem of linear programming (LP). However, we are interested in the complete cleanup of matrix in the framework of orthogonal projecting procedure (OPP), when inequalities number rapid expansion is possible with variables elimination. Suppose, that at the OPP next step after FCA implementation the generated system of m inequalities can contain noticeable amounts of dependent inequalities. In this situation the approach based on (**Crit₁**) direct use for m inequalities testing assumes LP m problems solving. It is obvious that such an approach is not attractive. Let us consider a

fundamentally different way of slump in amount of corresponding computation while verifying inequalities independence.

Suppose that a basic solution is found for the linear inequality system defining n-dimensional compact polyhedral set

$$Q = \left\{ \underline{x}_i \leq x_i \leq \bar{x}_i, i = 1, \cdots, n, \, l_s(x) \geq 0, \, s = 1, 2, \cdots m \right\} \quad (4)$$

obtained after the OPP next step and containing no pairwise equivalent inequalities. As a reminder, some solution of (4) is referred to as a basic one if it satisfies all constraints and transforms at least n inequalities into equalities. Relying on the conditions considered below we can verify dependence of all inequalities corresponding to the basic solution matrix (BSM). We shall first make some terms more precise.

By using variable shift we may reduce our problem to the case $\underline{x}_i = 0$. We also transform $x_i \leq \bar{x}_i$ to the form $-x_i + \bar{x}_i \geq 0$ and add it to the constrain set $\{l_s\}$. Below we shall refer to the x_j variables in the formula (4) as nonbasic (free) ones and the l_s variables—as basic (constrained) ones. Moreover, in respect of any simplex-matrix we shall speak of *column and row* variables. Any of these variables is associated with its non-negativity condition. Therefore, each inequality corresponds to one of the variables.

From this point onwards we shall refer to the difference between the dependence in respect of the Q set and the $l_j(x) \geq 0$ dependent inequality:

Definition 3.2. *The l_j is the first kind, if*
$\forall x \in Q : l_j(x) > 0.$

Definition 3.3. *The l_j is the second kind, if*
$\forall x \in Q : l_j(x) \geq 0$ *and* $\exists x \in Q : l_j(x) = 0.$

The analysis of inequality dependence, defining the Q polyhedral bounded set, can be based on the conditions given below.

1) *For row variables. If after the appropriate JE step in the course of the BSM construction or investigation the x_s-th with all non-negative coefficients constant term inclusive is discovered, then in accordance with the (**Crit₂**) the $x_s \geq 0$ inequality is dependent and the x_s-th row can be eliminated from the matrix.*

2) *For column variables. If all constant terms in the BSM are positive, then independent inequalities correspond to all column variables non-negativity condition.*

As a matter of fact, if an x_s column variable is given $\delta x_s < 0$, negative increment, the constant terms of each j-th row b_j will have the value $b'_j = b_j - a_{js} |\delta x_s|$. If $a_{js} = 0$, then $b'_j = b_j$, if $a_{js} < 0$, then $b'_j > b_j$, if $a_{js} > 0$ and $|\delta x_s| \leq \min_j b_j / a_{js}$, then $b'_j \geq 0$. This means, that there's the interval $-\delta x < x_s < 0$, where the only $x_s \geq 0$. inequality is violated, *i.e.* the reference set without this constraint will be an extended one. There-

fore, in accordance with (**Crit₃**) this inequality is independent.

From this point onwards we shall assume, that the homogeneous inequalities containing zero constant terms correspond to the first p rows of the BSM. In this case all column variables correspond to the $x_s \geq 0$ inequalities, which on the basis of the (**Crit₁**) are not the first kind dependent, but represent the tangent hyperplanes. Here two cases are possible:

a) The $x_s \geq 0$ half-space contains the Q *face*, which has dimension $n-1$, where $n = \dim Q$. Then $x_s \geq 0$ is an independent inequality.

b) The $x_s = 0$ hyperplane contains the Q *edge*, rather than the Q *face*, generated by k faces intersection, which has dimension $n-k$, $2 \leq k \leq n$. Then $x_s \geq 0$ is a dependent inequality of the second kind.

The following is valid.

3) *The $x_s \geq 0$ column inequality is independent, when and only when there exists a point of the $x_s = 0$ hyperplane, in which all row homogeneous inequalities are satisfied with positive change of all other column variables.*

As a matter of fact, if the $x_s \geq 0$ inequality is independent, there is $x^0 : x_s^0 = 0$, $l_j\left(x^0\right) > 0$, $j \neq s$ for all Q inequalities homogeneous ones inclusive. But if
$\exists x^0 : x_s^0 = 0, x_j^0 > 0, j \neq s$ for all column variables, then $x^0 \in \mathbf{R}^{n-1}$. Therefore, the $x_s = 0$ hyperplane contains the Q space, *i.e.* it is an independent inequality.

In order to check a column inequality independence with regard to (2), it is sufficient to check fulfillment of (3) only for homogeneous inequalities. While checking the $x_s \geq 0$ column inequality independence let us consider such a small shift d from the basic solution point simultaneously in all column variables but x_s, with which none of *homogeneous* row inequalities is violated. Let us assume $x_s = 0, \forall i \neq s : x_i = u_i d, q \leq u_i \leq 1$, where q is a small positive number, for example, $q = 0.0001$. Herewith, $0 < x_i \leq d$. Let us write down the row homogeneous inequality system $\sum_{i \neq s} a_{ji} x_i = d \sum_{i \neq s} a_{ji} u_i \geq 0$ or $\sum_{i \neq s} a_{ji} u_i \geq 0$. Making a change of $v_i = u_i - 0.0001$, we obtain

$$\sum_{i \neq s} a_{ji} v_i + 0.0001 \sum_{i \neq s} a_{ji} \geq 0, \ j = 1, \cdots, p, \ 0 \leq v_i \leq 0.9999. \ (5)$$

According to (3) the $x_s \geq 0$ inequality is independent when and only when the system (5) is compatible, *i.e.* it is possible to find a basic solution for it. Let us check this possibility by solving the corresponding linear problem.

With the help of (3) it is possible to determine all dependent column inequalities and to eliminate each of them from Q by first performing the appropriate JE step. But if we take into account that any row inequality in the BSM can become a column one with the help of

one or several JE steps, the following concluding proposition is true:

Proposition 3.1. *With the help of* (1), (2), (3) *all dependent inequalities can be determined in the BSM.*

Returning to the orthogonal projection procedure let us note that the computation amount in the course of matrix complete cleanup can be reduced with regard for the following important point. If with elimination of a next x_r variable the M_r matrix contains j-th rows, for which $a_{jr} = 0$, and the corresponding j-th inequality dependence was determined in the course of matrix cleanup during the preceding iterations, their independence repeated check is meaningless. Legitimacy of such an approach follows from the following obvious proposition.

Proposition 3.2. *If $l\left(x_{s+1}, \cdots, x_n\right) \geq 0$ is the inequality independent on the Q_n set, then it is the inequality independent on Q_{n-s} set and vice versa.*

Therefore, under the FCA framework with each new inequality subscript building it is reasonable to complement it with "its independence temporary non-identification" criterion.

3.5. Algorithm of Full Cleanup of Redundant Inequalities (Step 4 of 3.2)

Step 1. Reduce the matrix of linear inequalities system to the BSM form, in which all free members are nonnegative.

Step 2. If there are inequalities having all coefficients nonnegative in BSM, we mark them as redundant.

If all free members in BSM are positive, then we mark all column variables as independent and go to Step 4, otherwise to Step 3.

Step 3. Mark subset of rows with null free members. In the cycle by column variables x_s form linear problem (5). Solving it, we obtain the answer about the independence of variable x_s.

Step 4. If all variables are identified by independence mark, then we complete the process, otherwise go to Step 5.

Step 5. Exchange BSM by means of JE in order for some row variables that were not tagged with the sign of independence to move to column ones. Return to Step 2.

3.6. Methods of Matrix Additional Cleanup

Appropriate polyhedron boundaries moderately small alteration, resulting in transforming some independent inequalities into dependent ones, is associated with matrix additional cleanup.

Similar boundaries alteration can be grounded as initial system parameters as determination accuracy is not high as a rule. In such cases the position of boundaries is to be considered in a "near-boundary layer". In the most general case similar description needs the introduction of a poly-

hedral set comparison special measure Lotov [5], Efremov, Kamenev and Lotov [11].

Essentially boundaries alteration can be made with a sacrifice in both expansion (inequalities part elimination) and contraction (variable range restriction) of inequality system feasibility region. From this point onwards we shall consider only the first way.

Surface fine-cellular structure is typical for linear polyhedra if they are defined by a large number of inequalities. Therefore, its minor "coarsening" in application studies is admissible with a sacrifice in neglecting of some side defined as "almost dependent (redundant)" inequalities. To implement such an approach it is first of all necessary to introduce a measure of any inequality dependence on a corresponding set. It is obvious that such measure selection is heuristic.

Assume $l(x) = \sum_i a_i x_i + b \geq 0$ is one of inequalities defining Q set, and the Q' is the same set, but in the absence of this inequality. It appears that it is convenient to characterize this inequality measure of dependence on the Q set with a value proportional to $d = \min_{x \in Q'} l(x)$. As a matter of fact, we are dealing with BSM therefore, the Q system constant term vector doesn't contain negative constant terms. If $d \geq 0$, then $l(x) \geq 0$ is a dependent inequality. But if $d < 0$, its elimination from Q definition is equivalent to the b constant term increase by the value $|d|$. This value is proportional to the distance between the $l(x) = 0$ hyperplane and the vertex of Q' (in half-space $l < 0$), the least distant from it, and therefore, it makes it possible to infer the Q extension degree. In particular, for projection of Q on x_i axis its extension is equal $\cos(u)d$, where u is the angle between the $l(x) = 0$ hyperplane normal and x_i axis. Therefore, the d value itself can be used for the upper estimate of such an extension. It is obvious, that each inequality in the Q dependence degree estimates should be comparable to each other. It follows, that they should be represented in "relative units", in particular, in fractions of corresponding constant terms, *i.e.* by the value $e = d/b$, under the condition, that $b > 0$. Therefore, with the assumption of a rate value e_0, we shall consider the $l_j(x) \geq 0$ non-homogeneous inequality to be "almost dependent" on Q, if the $l_j(x) + e_0 b_j \geq 0$ inequality is dependent on the Q' set, *i.e.* if $\min_{x \in Q'} l_j(x) \geq -e_0 |b_j|$.

Let us consider a simple algorithm to estimate dependence degree of some non-homogeneous inequalities in the BSM containing only independent inequalities. Let us compute and put in order of increasing the $d_{sj} = b_j / |a_{js}|$ values for all $j : a_{js} < 0$ for each column variable x_s. Assume the first place in this list is occupied by the row (several rows) corresponding to the x_u, variable and the second—to the x_v variable. (In the case, when $b_u = 0$, we proceed to the next column variable). If

we make the JE with the a_{vs} resolvent element, we will find that the $b'_u = b_u - b_v a_{us} / a_{vs}$ constant term is negative. With b_u increase by the b'_u value the $x_u \geq 0$ inequality would be found out dependent of the second kind. It follows that $b'_u = d_u$.

Proposition 3.3. *If $d_u \geq -e_0 b_u$, and no negative coefficients are found out among the u-th row coefficients, this inequality can be considered to be "almost dependent" and it can be eliminated from the BSM. Similar computation cycles should be repeated, the pattern of column variables should be changed every time.*

3.7. Inequalities Number Expansion Process Control with Variables Elimination by CAC

While using matrix additional cleanup it is possible to find a simple trade-off between two contradictory stimuli: the desire to prevent too rapid number expansion of the inequalities defining projections and the desire to prevent polyhedron surface excessive alteration. To implement such a trade-off let us assume two parameters.

The first one is represented as the (e_0) inequalities coarsening maximum admissible rate in the course of matrix additional cleanup and therewith associated the "working value" $e = 0.1 q e_0$ of this rate, where the q controlled variable can take the discrete values $0, 1, 2, \cdots, 9, 10$. For practical use it is reasonable to assume the e_0 value to be within the range of 0.05 - 0.10.

The second parameter is the maximum admissible multiplicity (h) of row number in the matrix increase with elimination of variables in respect of the parent matrix. Thus inequalities number admissible threshold valueв is $PRG = r_0 h$. The h rational value depends on the available "computation resources" and can vary within 3 - 10.

Let us assume that after the elimination of the next variable x_k the working value $q_k > 0$, and the number of inequalities in the matrix is $r_k < PRG$. Assume that after the elimination of the variable x_{k+1} with the help of FCA algorithm and the complete cleanup of matrix the inequalities number therein exceeds the preset threshold. In this case let us additionally clean the matrix with the e_k rate. If it doesn't provide the r_{k+1} value decrease needed, let us assume $q_{k+1} = q_k + 1$, then let us again additionally clean the matrix, but now in an "intense mode", *i.e.* with the e_{k+1} rate. Let us repeat this procedure until one of the two conditions is fulfilled:

1) $r_{k+1} < PRG$ and $q_{k+1} < 10$, or

2) $r_{k+1} \geq PRG$ and $q_{k+1} = 10$.

In the first case it will be possible to further intensify matrix cleanup restricting inequalities number in newly generated matrices during the following iterations if necessary. In the second case the process of variable elimination with complete and additional cleanup of generated

matrices will be continued, however, the possibility of matrix additional cleanup intensification will be lost.

If in the course of variable elimination it is necessary to resort to matrix additional cleanup, then not only the corresponding matrix is compared to each k-th iteration, but also the e_k coarsening rate value, which was used during its generation. It makes it possible to avoid inadmissible solutions occurring as a result of the polyhedron "improper" extension during "descent of matrix pyramid". Precisely, in the formally computable range of the x_k-th variable $\left[\underline{x}_k, \overline{x}_k\right]$ it is permissible to chose x_k values only in the range $\left[\underline{x}_k + \Delta_k, \overline{x}_k - \Delta_k\right]$, where $\Delta_k = 0.5 e_k\left(\overline{x}_k - \underline{x}_k\right)$.

4. Stability of the Polyhedron Orthogonal Projection and Complexity of Its Algorithm

4.1. Stability of the POP Algorithm

Let us give an inequality system (4). It gives us a compact polyhedron $Q \subset \mathbf{R}^n$. Here two cases are possible: Q is solid or singular. In the first case it is valid.

Proposition 4.1. *If the polyhedron Q is solid, then POP is stable.*

Proof. Using (4), we introduce $M = \max_i \max \left\{\left|\underline{x}_i\right|+1, \left|\overline{x}_i\right|+1\right\}$. Denote by l_j^δ such a perturbation of inequality l_j, in which all coefficients a_{ji} and free member b_j have an increment in absolute value not exceeding a given $\delta > 0$. Denote by Q^δ the polyhedron obtained if in (4) inequalities l_j replaced by l_j^δ. Suppose that Γ_j is the face in Q and Γ_j^δ—in Q^δ asked by the conditions $l_j = 0$ and $l_j^\delta = 0$, respectively. For an inequality l_j we introduce the rate $N\left(l_j\right) = \sqrt{\sum_i \left(a_{ji}\right)^2}$. Suppose that all inequalities have nonzero rates. Define $N = \max_j \left\{\dfrac{1}{N\left(l_j\right)}\right\}$. We fix $\varepsilon > 0$ and take $\delta = \dfrac{\varepsilon}{2N\left(Mn+1\right)}$. Then we have

$$\forall x \in Q : \left|l_j(x) - l_j^\delta(x)\right| = \left|\sum_{i=1}^n \delta\left(a_{j,i}\right)x_i + \delta\left(b_j\right)\right| \le Mn\delta + \delta$$

It follows that $\forall x \in Q : \left|l_j(x) - l_j^\delta(x)\right| \le \dfrac{\varepsilon}{2N}$. We now take $x \in \Gamma_j$. Then we have $l_j(x) = 0$. Hence $\left|l_j^\delta(x)\right| = \left|l_j(x) - l_j^\delta(x)\right| \le \dfrac{\varepsilon}{2N}$. Note that $\rho\left(x, \Gamma_j^\delta\right) = \dfrac{\left|l_j^\delta(x)\right|}{N\left(l_j^\delta\right)}$. Immediately it is verified that there exists $\delta_0 > 0$ such that with condition $\delta \le \delta_0$ for any

j it is valid: $\dfrac{1}{2} \le \dfrac{N\left(l_j\right)}{N\left(l_j^\delta\right)} \le 2$. It follows that

$\forall x \in \Gamma_j : \rho\left(x, \Gamma_j^\delta\right) < \varepsilon$. Here $\rho\left(x, \Gamma_j^\delta\right)$ is the distance from the point to the space.

Similarly, it is easy to see that $\forall x \in \Gamma_j^\delta : \rho\left(x, \Gamma_j\right) < \varepsilon$. Denote by $\varepsilon(Q)$ ε—neighborhood of Q. Then we have

$$Q^\delta \subset \varepsilon(Q), \ Q \subset \varepsilon\left(Q^\delta\right). \tag{6}$$

Denote by P the orthogonal projection on a subspace. Since the distance between the points cannot increase in the projection, from (6) it follows that

$$P\left(Q^\delta\right) \subset \varepsilon\left(P(Q)\right), \ P(Q) \subset \varepsilon\left(P\left(Q^\delta\right)\right) \tag{7}$$

From (7) it follows that a sufficiently small perturbation of coefficients and free members of inequalities leads to a small perturbation of polyhedron projection. This completes the proof.

In the case of singular polyhedron an arbitrary small perturbation can lead to a significant change in projection as follows

Example 4.1. Let Q be the polyhedron defined by the inequalities:

$$x_1 - x_2 \ge 0, \ -x_1 + x_2 \ge 0, \ 0 \le x_1 \le 1, \ 0 \le x_2 \le 1.$$

Denote by P_1 the orthogonal projection on the axis x_1. We introduce a perturbation Q^δ of Q.

$$x_1 - \left(1+\delta\right)x_2 \ge 0, -x_1 + \left(1-\delta\right)x_2 \ge 0, 0 \le x_1 \le 1, 0 \le x_2 \le 1.$$

Then we have $P_1(Q) = [0,1]$ and for any $\delta > 0$ it holds $P_1\left(Q^\delta\right) = \{0\}$.

4.2. The Complexity of the POP Algorithm

The complexity of POP algorithm is defined by two factors:

1) the growth of inequality number in the system by iterations;

2) the complexity of the algorithm of redundant inequality cleanup for a single iteration.

Let us consider 1). Let $Q \subset \mathbf{R}^n$ be a compact polyhedron defined by the inequality system (4) and P_k be the orthogonal projection onto a subspace $\mathbf{R}^{n-k} \subset \mathbf{R}^n$. Note that single iteration of the orthogonal projection procedure is accompanied by a full cleanup of redundant inequalities. Therefore, an inequality $(l \ge 0)$ in the system of projection corresponds to hyperplane $(l = 0)$ containing a face Γ of $P_k(Q)$. Similar to [16,17] this hyperplane can be represented as π projection of the intersection of $k+1$ hyperplanes $\left(l_{j(1)} = \cdots = l_{j(k+1)} = 0\right)$ of the original inequality system. Hence, we obtain an estimation of the number N_k of inequalities for this projection by the binomial coefficients:

$$N_k \leq C_{m+2n}^{k+1}. \tag{8}$$

Here m is the number of inequalities (l_j) in (4) and $2n$ correspond to parallelepiped restrictions.

At 2) we estimate the cause of complexity of redundant inequality cleanup. It is determined by the number of appeals to the solution of linear problems (5). Typically this problem has a low dimension and is solved by means of the simplex-algorithm. As it is well known the simplex-algorithm has an exponential complexity [18]. Note that in the linear problem (5) the threshold q of accuracy of its solution is introduced. This allows the use of the polynomial algorithms of linear problem solution. Here Khachiyan [19] and Karmarkar [20] algorithms are known. The Karmarkar algorithm [20] is more efficient and has the complexity

$$Compl_k = O\left(n^{3.5}L^2 \ln(L) \ln(\ln(L))\right).$$

Here n is the number of free variables of linear problem and L is the number of bits of input to the algorithm. It is suggested that a linear problem is solved for each inequality. Then for complexity $ComplConv_k$ of the redundant inequality cleanup of the iteration k, we have the majorant

$$ComplConv_k \leq N_k \times Compl_k. \tag{9}$$

5. Computational Experiment Results

In accordance with the considered algorithm we have developed *a demonstration version* of the program implementing the orthogonal projecting procedure (OPP) by using the language VISUAL BASIC 6th version with a 32-bit compiler. During computational experiments the following computer resources were used: PC Intel-Core2 with clock speed of 2.67 GHz, capacity of internal memory of 3.25 Gb and the operating system Microsoft Windows XP Professional 2003. The computational experiments aimed to analyse FCA improvement proposed measures effectiveness. The matrices of form (3) were investigated. The LP problem was used for computation accuracy strict check with OPP implementation. In particular, first the projection of the entire feasibility region is constructed on the objective function axis. Then, the optimal resolution vector is determined with the previously computed matrices sequence "descent". This estimation of solution accuracy is made by its comparison with the same problem solving by means of simplex-method. Herewith material inconsistency is admissible only in the case, if the problems pair being considered has an ambiguous solution.

When the procedure is used in practice, the following parameters are rather essential beyond computational accuracy: 1) expansion maximum degree of the number of inequalities with the elimination of variables; 2) com-

puting time needed for the elimination of a given set of variables. This time depends on available computation resources, the specific nature of the parent matrix, the number of eliminated variables and after all on the algorithm being used. Therefore, all experiments were held on the same computer, with the same structure matrices (3), with the same requirement of all nonbasic variables elimination. The matrices being considered had only various (N) dimensions of polyhedra. Herewith the parent matrix contains $2N$ rows, since N rows are supplemented by the requirements $x_i \leq 2$. Matrices inadmissible expansion with the use of FCA only is clearly illustrated in **Table 1** with $N = 10$. At the next stage algorithm performance capabilities were investigated with complete cleanup of matrices, but without additional cleanup, in particular, without their expansion control. At the final stage the algorithm with additional cleanup, *i.e.* with "coarsening" was considered.

Table 3 gives the consolidated results of numerical experiments for various dimensions of the parent matrices. The **Table 3** contains

N—parent matrix dimensions,

M—inequalities in the matrix number maximum expansion after its complete cleanup,

T—time (sec) of all variables elimination,

Relative deviations from the simplex solution of the LP appropriate problem:

$D1$—deviation of the optimal value of the objective function;

$D2$—maximum deviation of the optimal solution variables;

$D3$—average deviation of the optimal solution variables.

All **Table 3** rows (excluding the last two rows) reproduce results of convolution with the use of matrix complete cleanup, but without coarsening. With matrices cleanup in LP problems, which are represented in the last two rows, a coarsening with threshold 3 and maximum rate 0.1 for $N = 50$ and 0.12 for $N = 100$ was used. The **Table 3** analysis shows, that the rate-determining

Table 3. Combined results of experiments.

N	M	T	$D1$	$D2$	$D3$
10	20	1	<E–14	<E–14	<E–14
20	40	6	<E–14	<E–14	<E–14
30	66	26	3.25E–10	3.37E–9	2.82E–10
40	402	1176	3.61E–10	4.50E–9	4.35E–10
50	602	4564	9.72E–10	2.05E–8	1.19E–9
50 (*coarsening*)	343	454	4.00E–3	3.87E–2	1.87E–3
100 (*coarsening*)	506	3163	4.41E–2	2.74E–2	4.69E–4

factor with convolution of the high-dimensional inequality systems is not so much expansion thereof, as computation time. In the above mentioned conditions the admissible computation amount (76 min 4 sec) was obtained with $N = 50$ with the use of only matrix complete cleanup. The fragments of corresponding automatically generated protocol, giving the idea about inequality system convolution history, are represented in **Table 4**.

Table 4 illustrates the fact that the generated matrix cardinality dependence on the next iteration index is ambiguous. It is determined by the variables elimination order and by the distribution of positive and negative coefficients in the eliminated column.

Comparison of the data represented in the two rows of **Table 3** for problem with $N = 50$ dramatizes matrix additional cleanup algorithm (coarsening algorithm) performance capabilities. In the discussed example its use made it possible to reduce inequality system maximum expansion by 1.76 times, computation time—by 10.05 times, however, it is accompanied by objective function optimal value computation accuracy loss up to the level of 0.004. In those cases, when such loss is admissible, the use of the coarsening algorithm can be considered to be rather effective. Below the algorithm of choosing the of attractive admissible point 3.3 is illustrated in example

Table 4. Convolution history for $N = 50$.

Iteration	Fourier	Chernikov	Complete cleanup
Initial	100		93
1	157	157	106
3	682	148	137
6	2096	194	190
9	4358	253	195
12	2763	191	115
15	142	81	69
18	248	224	139
21	10742	643	439
24	35711	800	506
28	32577	676	602
31	3337	281	213
34	1109	137	76
37	2419	926	101
40	2704	303	99
43	1514	182	51
46	21	21	14
50	5	5	5

(3) with $N = 10$. The course of building of system (3) solution in the explicit form is given in **Table 5**. Note that the value of variable 1 is calculated, *i.e.* it is not given in the dialog.

6. Conclusions

The computer experiments performed enable us to establish the following:

- The algorithms of matrix complete cleanup make it possible to obtain quite a high accuracy of computation.
- A rapid increase in computation time span with problem dimensions increase is the main disadvantage of the demonstration version of the program used.
- The use of matrix additional cleanup is effective in cases, where the initial information is in error of several percents.
- We can hope that estimations (8), (9) of the complexity of POP are overvalued and can be improved significantly.

It is useful to mention the following trends of OPP use in applied research:

In defining linear model of an object whose variables are not relevant for the investigator, though they cannot be neglected, the corresponding set projection to subspace, not containing those variables, will make it possible to retain all the features of the model being investigated. Such an application of the convolution method is known

Table 5. Dialog interface for the building of an inequality system solution.

Name of variable	Iteration of convolution	Low variable bound	High variable bound	Field of input of variable value
Objective function	10	0.00000	3.84323	3.80000
Variable 8	9	3.42561	4.00000	3.50000
Variable 9	8	1.00000	1.02042	1.01000
Variable 10	7	3.94127	4.00000	3.98000
Variable 2	6	1.00000	1.00636	1.00300
Variable 5	5	3.52671	3.53210	3.53000
Variable 6	4	3.99698	4.00000	3.99800
Variable 3	3	2.64854	2.64920	2.64900
Variable 7	2	2.46478	2.46693	2.46500
Variable 4	1	1.00000	1.00013	1.00010
The substitution of the original system of constrains is over. There are no errors				
Option with the objective function. The objective function				3.80000
Variable 1				3.99999

in biophysics (Nikolaev, Burgard and Maranas [21]).

In parametric programming problems, in which it is useful to find the dependence of LP problem optimal solution on such linear model parameters as absolute terms of constrains and (or) boundaries of variables (Keerthi and Sridharan [22]).

In the problems of process-admissible deviations from any products rated values, for which operability scope thereof Q is defined by a linear model. The use of OPP will make it possible use Q much more fully with a transfer from universally accepted mutually independent tolerances to a system of dependent tolerances, because it allows us to replace a parallelepiped inscribed in Q with a polyhedron.

In various problems on agreeing solutions in search of a compromise between partners (Shapot and Lukatskii [7]).

In problems of a multi-objective choice while projecting region of feasibility on subspace of criteria (Karbovsky, Lukatskii and Shapot [23]).

7. Acknowledgements

REFERENCES

[1] J. B. Fourier, "Solution d'une Question Particulière du Calcul des Inégalités," *Nouvean Bulletin des Sciences par la Société Philomatique de Paris*, 1826, p. 99.

[2] D. V. Shapot, "On the Construction of Orthogonal Projections of Point Sets Defined by a System of Inequalities," *Journal Computational Mathematics and Mathematical Physics*, Vol. 11, 1971, pp. 1113-1126.

[3] S. N. Chernikov, "Linear Inequalities," Nauka, Moscow, 1968.

[4] V. A. Bushenkov and A. V. Lotov, "An Algorithm for Analysis of Inequality Dependences in a Linear System," *Journal Computational Mathematics and Mathematical Physics*, Vol. 20, 1980, pp. 562-572.

[5] A. V. Lotov, "On the Estimate of Stability and the Condition Number of the Solution Set of a System of Linear Inequalities," *Journal Computational Mathematics and Mathematical Physics*, Vol. 24, 1984, pp. 1763-1774.

[6] I. I. Eremin and A. A. Makhnev, "International Workshop on Algebra and Linear Optimization Dedicated to the 90th Anniversary of the Birth of S. N. Chernikov," *Ural State University Bulletin*, Vol. 30, 2004, pp. 183-184.

[7] D. V. Shapot and A. M. Lukatskii, "A Constructive Algorithm for Foldind Large-Scale Systems of Linear Inequalities," *Computational Mathematics and Mathematical Physics*, Vol. 48, No. 7, 2008, pp. 1100-1112. doi:10.1134/S0965542508070038

[8] T. S. Motzkin, H. Raiffa, G. L. Thompson and R. M. Trall, "The Double Description Method," *Contributions to the Theory of Games*, Vol. 2, 1953, pp. 51-73.

[9] F. P. Preparata and M. I. Shamos, "Computational Geometry. An Introduction," Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 1985.

[10] V. N. Shevchenko and D. V. Gruzdev, "Modifications of Fourier-Motskin Algorithm for Triangulation Construction," *Discrete Analysis and Investigation of Operations*, Series 2, Vol. 10, 2003, pp. 53-64.

[11] R. V. Efremov, G. K. Kamenev and A. V. Lotov, "Constructing an Economical Description of a Polytop Using the Duality Theory of Convex Sets," *Doklady Mathematics*, Vol. 70, 2004, pp. 934-936.

[12] A. I. Golikov and Y. G. Evtushenko, "A New Method for Solving Systems of Linear Equalities and Inequalities," *Doklady Mathematics*, Vol. 64, 2001, pp. 370-373.

[13] P. O. Gutman and I. Ioslovich, "On the Generalized Wolf Problem: Preliminary Analysis of Nonnegativity of Large Linear Programs Subject to Group Constrains," *Automation Telemech*, No. 8, 2007, pp. 116-125.

[14] S. Paulraj and P. Sumathi, "A Comparative Study of Redundant Constrains Identification Methods in Linear Programming Problems," *Mathematical Problems in Engineering*, Vol. 2010, 2010, pp. 1-16. doi:10.1155/2010/723402

[15] D. V. Shapot and A. M. Lukatskii, "A Constructive Algorithm for Building Orthogonal Projections of Polyhedral Sets," *Optimization*, *Control*, *Intelligence*, Sibirian Energy Institute, Irkutsk, 1995, pp. 77-91.

[16] D. Hilbert and S. Cohn-Fossen, "Anshauliche Geometrie," Verlag J. Springer, Berlin, 1932.

[17] A. D. Alexandrov, "Convex Polyhedra," Springer-Verlag, Berlin, 2005.

[18] V. Klee and G. Minty, "How Good Is the Simplex-Algorithm," *Proceedings of the Third Symposium on Inequalities*, Los Angeles, 1972, pp. 159-175.

[19] L. Khachiyan, "A Polynomial Algorithm in Linear Programming," *Soviet Mathematics—Doklady*, Vol. 20, 1979, pp. 191-194.

[20] N. Karmarkar, "A New Polinomial Time Algorithm for Linear Programming," *Combinatorics*, Vol. 4, No. 4, 1984, pp. 373-395. doi:10.1007/BF02579150

[21] E. V. Nikolaev, A. P. Burgard and C. D. Maranas, "Education and Structural Analysis of Concerved Pools for Genom-Scale Metabolic Reconstruction," *Biophysical Journal*, Vol. 88, No. 1, 2005, pp. 37-49. doi:10.1529/biophysj.104.043489

[22] S. S. Keerthi and K. Sridharan, "Solution of Parametrized Linear Inequalities by Fourier Elimination and Its Applications," *Journal of Optimization Theory and Applications*, Vol. 65, 1990, pp. 161-169.

[23] I. N. Karbovsky, A. M. Lukatskii and D. V. Shapot, "Use of Orthogonal Projection Method by the Example of Analysis of Industries Possible Price Response to Natural Monopolies Products Price Shocks," *Proceeding of the Second International Conference MLSD* 2008, Institute of Control Sciences, Moscow, 2008, pp. 117-126.