Parallel Method of Pseudoprojection for Linear Inequalities

Irina Sokolinskaya(✉)

South Ural State University, 76 Lenin prospekt, Chelyabinsk 454080, Russia
Irina.Sokolinskaya@susu.ru

Abstract. This article presents a new iterative method for finding an approximate solution of a linear inequality system. This method uses the notion of pseudoprojection which is a generalization of the operation of projecting a point onto a closed convex set in Euclidean space. Pseudoprojecting is an iterative process based on Fejer approximations. The proposed pseudoprojection method is amenable to parallel implementation exploiting the subvector method, which is also presented in this article. We prove both the subvector method correctness and the convergence of the pseudoprojection method.

Keywords: Linear inequality system · Iterative method
Fejer approximations · Pseudoprojection · Parallel algorithm
Convergence

1 Introduction

In various numerical problems, we are often confronted with the task of solving a system of linear inequalities:

$$l_i(x) = \sum_{j=1}^{n} a_{ij}x_j - b_i \leqslant 0 \quad (i = 1, \dots, m) \tag{1}$$

under the condition that system (1) is consistent. In the general case, the task of solving a system of linear inequalities is a difficult one. Thus, in practice, methods making it possible to find an approximate solution in a finite number of iterations are frequently applied. In [1,2], Motzkin and Agmon proposed a relaxation method for finding an approximate solution of a consistent system of linear inequalities. Let us consider the main idea of this relaxation method. When considering system (1), it is convenient to use a geometric language. Thus, we look upon $x = (x_1, \dots, x_n)$ as a point in n-dimensional Euclidean space $\mathbb{R}^n$, and each inequality $l_i(x) \leqslant 0$ as a half-space P_i. The set of solutions of

I. Sokolinskaya—The study has been partially supported by the RFBR according to research project No. 17-07-00352-a and by the Government of the Russian Federation according to Act 211 (contract No. 02.A03.21.0011).

L. Sokolinsky and M. Zymbler (Eds.): PCT 2018, CCIS 910, pp. 216–231, 2018.
https://doi.org/10.1007/978-3-319-99673-8_16

system (1) therefore is the convex polytope $M = \bigcap_{i=1}^{m} P_i$. Each equation $l_i(x) = 0$ defines an hyperplane H_i. In [2], the following iterative algorithm for finding an approximate solution of the system (1) is proposed. Below, λ such that $0 < \lambda < 2$ is a parameter of the algorithm. The parameter λ is called the *coefficient of relaxation.*

1. Choose an arbitrary point $x_0 \in \mathbb{R}^n$.
2. $x := x_0$.
3. If $x \in M$ then a solution is found; go to Step 8.
4. Select a half-space P_i such that $\text{dist}(x, P_i) = \max_j \text{dist}(x, P_j)$[1].
5. Calculate the point x' which is the orthogonal projection of x onto hyperplane H_i.
6. $x := x + \lambda(x' - x)$.
7. Go to Step 3.
8. Stop.

Thus, the algorithm computes a sequence of points $x_0, x_1, \ldots, x_k, x_{k+1}, \ldots$, where $x_{k+1} = x_k + \lambda(x'_k - x_k)$, and x'_k is the orthogonal projection of the point x_k onto hyperplane H_i bounding the half-space P_i, so that

$$\text{dist}(x_k, P_i) = \max_j \text{dist}(x_k, P_j).$$

There are two alternatives: (1) the process terminates after K steps with the point $x_k \in M$; (2) the process continues indefinitely, producing an infinite sequence $\{x_k\}$. In [1], Agmon showed that if $0 < \lambda < 2$ and the sequence $\{x_k\}$ is infinite, then x_k converges, as $k \to \infty$, to a point on the boundary of the polytope M. In this case, we can use the condition $\text{dist}(x_k, P_i) < \varepsilon$ as a stopping criterion. Here, $\varepsilon > 0$ is an arbitrarily small positive quantity. After stopping, the last point x_k is taken as an approximate solution of system (1).

The Motzkin–Agmon method has been extended in a number of works. In [3], a generalized relaxation method was proposed and investigated, based on the introduction of so-called *subcavities*. In certain cases, this generalized method provides faster convergence in comparison with the Motzkin–Agmon method. In [4,5], an extension of the relaxation method was considered for finding the common point of convex sets. In [6], the relaxation method is extended for solving systems of non-linear inequalities. In [7], a new parameter "cone angle" is introduced and the convergence and finiteness of the relaxation method for different values of this parameter are investigated. In [8,9], an extension of the relaxation method for systems with an infinite number of linear inequalities in a finite-dimensional space was proposed and investigated. In [10], the *underrelaxation* method with $0 < \lambda < 1$ is studied, and new bounds on convergence are obtained when the linear inequalities are processed in a cyclical order. In [11,12], a combined relaxation method for non-linear convex variational inequalities is described and studied.

[1] Here $\text{dist}(x, P) = \inf\{\|x - y\| : y \in P\}$.

In 1922, Leopold Fejer introduced the following definition of the closeness of points to a closed set M in the Euclidean space $\mathbb{R}^n$ (see [13]). If x and x' are points of $\mathbb{R}^n$ such that

$$\|x - y\| > \|x' - y\| \tag{2}$$

for every $y \in M$, then we say that x' is *point-wise* closer than x to the set M. If x is such that there is no point x' which is point-wise closer than x to M, then x is called the closest point to the set M. Fejer pointed out that the set of closest points to M is identical to the convex hull of the set M. Using this observation, Eremin in [14,15] introduced and investigated Fejer mappings, making it possible to construct iterative methods for solving problems of various types: systems of convex inequalities and problems of convex programming, ill-posed problems of mathematical physics in the presence of additional functional constraints, and others. The notion of pseudoprojecting a point onto a convex bounded set was introduced in [16]. The pseudoprojection operation is an extension of the projection operation using Fejer mappings. Based on the pseudoprojection operation, the authors of [16] developed a pseudoprojection method for solving linear inequality systems. This method is an extension of the relaxation method proposed by Motzkin and Agmon. Based on the pseudoprojection method, a set of parallel methods for solving large-scale non-stationary linear programming problems was developed and investigated in [16–19].

An iterative method for solving systems of linear inequalities based on determining the centroid is proposed in [20]. Each inequality defines a half-space of feasible points. The method starts with an arbitrary point in $\mathbb{R}^n$ as an initial approximation, and then calculates at each step the centroid of a subsystem of masses placed at the reflections of the previous iterate with respect to the bounding hyperplanes of only the violated half-spaces defined by the system of inequalities. This centroid is taken as the new iterate. In [21], a similar method is presented. In this method, each iterate lies in the half line determined by the previous one and a convex combination of its orthogonal projections on all the half spaces defined by the inequalities. The authors of [22] describe another iterative method for solving a system of linear inequalities in which each step consists of finding the orthogonal projection of the current point onto a hyperplane corresponding to a surrogate constraint constructed through a positive combination of a group of violated constraints. Note that the last three methods can be efficiently parallelized.

The present article is devoted to the development and investigation of a parallel pseudoprojection method to find an approximate solution of a system of linear inequalities. The method starts with an arbitrary point in $\mathbb{R}^n$ as an initial approximation, and then calculates a pseudoprojection of this point onto a convex polytope defined as the set of feasible solutions of linear inequality system (1). The subvector method is used to parallelize the Fejer process. The main idea of this method is that the vector determining the current approximation is divided into subvectors. For each subvector, a certain number of Fejer iterations is performed in parallel. Then the modified subvectors are combined into a single vector. The calculations are repeated until the required precision of approximation is obtained.

The rest of the paper is organized as follows. The formal definitions of Fejer mapping, Fejer process, as well as that of the pseudoprojection operation are given in Sect. 2. Section 3 is devoted to describing the algorithm for constructing a pseudoprojection onto a convex closed set. Section 4 describes the method of subvectors used for parallelization of the pseudoprojection algorithm. In Sect. 5, we prove the convergence theorem for the pseudoprojection calculation algorithm. The results obtained are summarized in Sect. 6, and further research directions are outlined herein.

2 Fejer Mappings and the Pseudoprojection Operation

Let us consider a consistent system of m linear inequalities,

$$Ax \leqslant b, \tag{3}$$

given in the n-dimensional Euclidean space $\mathbb{R}^n$ and written in matrix form. The matrix A has dimension $m \times n$. Let M be a polytope defined as the set of feasible solutions of linear inequality system (3). Such a polytope is always a closed convex set. A single-valued mapping $\psi \colon \mathbb{R}^n \to \mathbb{R}^n$ is said to be fejerian relatively to a set M (or briefly, M-fejerian) if

$$\psi\left(y\right) = y, \forall y \in M; \qquad \left\|\psi\left(x\right) - y\right\| < \left\|x - y\right\|, \ \forall y \in M, \ \forall x \notin M. \tag{4}$$

Let a_i be an i-th row of the matrix A $(i = 1, \ldots, m)$. Let us denote by $\langle a_i, x\rangle$ the dot product of vectors a_i and x. It is known [15,23] that the mapping

$$\varphi\left(x\right) = x - \frac{\lambda}{m}\sum_{i=1}^{m}\frac{\max\left\{\langle a_i, x\rangle - b_i, 0\right\}}{\left\|a_i\right\|^2} \cdot a_i \tag{5}$$

is a continuous single-valued M-fejerian mapping for the relaxation coefficient $0 < \lambda < 2$. We will use the notation

$$\varphi^s(x) = \underbrace{\varphi \ldots \varphi(x)}_{s}.$$

The *Fejer process* generated by the mapping φ for an arbitrary initial approximation $x_0 \in \mathbb{R}^n$ is the sequence $\{\varphi^s(x_0)\}_{s=0}^{+\infty}$. It is known [15] that the Fejer process converges to a point belonging to the polytope M:

$$\{\varphi^s(x_0)\}_{s=0}^{+\infty} \to \bar{x} \in M. \tag{6}$$

Let us denote this concisely as $\lim\limits_{s\to\infty} \varphi^s(x_0) = \bar{x}$. Let the φ-*projection* (*pseudo-projection*) of a point $x \in \mathbb{R}^n$ on the polytope M be understood as the mapping $\pi_M^{\varphi}(x) = \lim\limits_{s\to\infty} \varphi^s(x)$.

3 Parallel Algorithm for Constructing a Pseudoprojection

Let us introduce the following notation. Given an arbitrary linear subspace $\mathbb{P} \subset \mathbb{R}^n$, let us denote by $\pi_{\mathbb{P}}(x)$ the orthogonal projection of $x \in \mathbb{R}^n$ onto the linear subspace $\mathbb{P}$. Everywhere below, a linear subspace will be called simply a subspace. Denote by $\rho(\mathbb{P}, x) = \min_{p \in \mathbb{P}} \|p - x\|$ the distance between the point x and the subspace $\mathbb{P}$. Let the linear manifold $\mathbb{L}$ be constructed from subspace $\mathbb{P}$ by translating it by a vector z: $\mathbb{L} = \mathbb{P} + z$. Denote by $\pi_{\mathbb{L}}(x)$ the orthogonal projection of $x \in \mathbb{R}^n$ onto the linear manifold $\mathbb{L}$:

$$\pi_{\mathbb{L}}(x) = \pi_{\mathbb{P}}(x) + z. \tag{7}$$

Let $\varphi \in \{\mathbb{R}^n \to \mathbb{R}^n\}$ be a single-valued continuous M-fejerian mapping, where M is a convex closed set. Let us define a decomposition of the space $\mathbb{R}^n$ into a direct sum of orthogonal subspaces: $\mathbb{R}^n = \mathbb{P}_1 \oplus \ldots \oplus \mathbb{P}_r$, where $\mathbb{P}_i \perp \mathbb{P}_j$ for $i \neq j$. Let us construct a linear manifold $\mathbb{L}_i$ for each subspace

$$\mathbb{P}_i \ (i = 1, \ldots, r)$$

in the following way. Suppose that $\bar{x}^i \in \operatorname{Arg}\min_{x \in M} \rho(\mathbb{P}_i, x)$. Define $\bar{z}^i = \pi_{\mathbb{P}_i^{\perp}}(\bar{x}^i) \in \mathbb{P}_i^{\perp}$. Here, $\mathbb{P}_i^{\perp}$ denotes the orthogonal complement to the subspace $\mathbb{P}_i$. Let us construct the linear manifold $\mathbb{L}_i$ by translating $\mathbb{P}_i$ by a vector $\bar{z}^i$:

$$\mathbb{L}_i = \mathbb{P}_i + \bar{z}^i. \tag{8}$$

For each $i \in \{1, \ldots, r\}$, define the mapping $\varphi_i \in \{\mathbb{R}^n \to \mathbb{L}_i\}$ as

$$\varphi_i(x) = \pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}(x)\right)\right). \tag{9}$$

Assume that s is a positive integer and ε is a positive real number. The following algorithm calculates the pseudoprojection of the point $\mathbf{0} \in \mathbb{R}^n$ ($\mathbf{0}$ is the zero vector) onto the polytope M.

Algorithm $\mathfrak{S}$:

1. $k := 0$; $x_0 = \mathbf{0} \in \mathbb{R}^n$.
2. $x_{k+1} := \sum_{i=1}^{r} \left(\varphi_i{}^s\left(\pi_{\mathbb{L}_i}(x_k)\right) - \bar{z}^i\right)$.
3. If $\|x_{k+1} - x_k\| < \varepsilon \ \vee \ d_M(x_{k+1}) < \varepsilon$ then go to 6.
4. $k := k + 1$.
5. Go to 2.
6. Stop.

The performance of algorithm $\mathfrak{S}$ for $n = 2$ and $s = 2$ is shown in Fig. 1. To apply the algorithm $\mathfrak{S}$ to an arbitrary initial point $x_0 \in \mathbb{R}^n$, you must transfer the origin to the point x_0. In Step 3, the algorithm computes the residual function

$$d_M = \sum_{j=1}^{m} \max\{\langle a_j, x\rangle - b_j, 0\}. \tag{10}$$

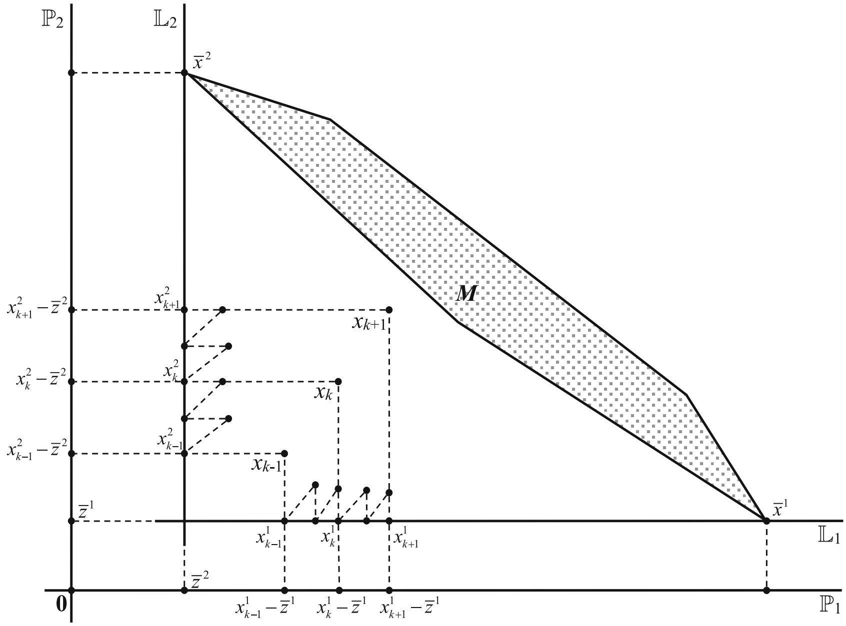

Fig. 1. The work of algorithm $\mathfrak{S}$: $x^1_k = \pi_{\mathbb{L}_1}(x_k)$, $x^1_{k+1} = {\varphi_1}^2(x^1_k)$; $x^2_k = \pi_{\mathbb{L}_2}(x_k)$, $x^2_{k+1} = {\varphi_2}^2(x^2_k)$.

This function determines the degree of closeness of the point x_{k+1} to the polytope M. We will show later on that the positive integer s is an important parameter influencing the potential scalability of algorithm $\mathfrak{S}$. By increasing s, we increase the resource of parallelism inherent in algorithm $\mathfrak{S}$. However, if one takes too large a value for the parameter s, then the sequence $\{x_k\}$ may converge to a point that does not belong to the polytope M. In this case, the iterative process will stop when the condition $\|x_{k+1} - x_k\| < \varepsilon$, included in the stopping criterion, is satisfied. If this happens, one needs to decrease the value of s and repeat the computational process. It is obvious that the most compute-intensive step of algorithm $\mathfrak{S}$ is Step 2, in which the Fejer process is implemented. To parallelize this step, the subvector method discussed below can be applied. The main idea of the method is as follows. For each subspace, a simultaneous Fejer process is performed. After every s steps, the results obtained on the subspaces are combined into one vector which is taken as the next approximation. If the stopping criterion checked in Step 3 is satisfied, then the last approximation is accepted as pseudoprojection. Otherwise, calculations continue.

4 Subvector Method

Let us consider the *subvector method*, applied to parallelize Step 2 of the $\mathfrak{S}$ algorithm. Let $r \in \mathbb{N}$ be such that $r \leqslant n$, where n is the space dimension. For simplicity, we always assume that r is a multiple of n: $n = r \cdot l$. Assume that

$$\{e_1, \ldots, e_n\} \tag{11}$$

is an orthonormal basis of the space $\mathbb{R}^n$. Let us define the linear subspaces as

$$\mathbb{P}_i = \mathrm{Lin}(\{e_{1+(i-1)l}, \ldots, e_{l+(i-1)l}\}) \tag{12}$$

for $i = 1, \ldots, r$. In Eq. (12), Lin denotes the linear hull of vectors. It is obvious that $\mathbb{P}_i \perp \mathbb{P}_j$ for $i \neq j$, and $\mathbb{P}_1 \oplus \ldots \oplus \mathbb{P}_r = \mathbb{R}^n$. Let $\bar{x}^i \in \operatorname*{Arg\,min}_{x \in M} \rho(\mathbb{P}_i, x)$. Denote $\bar{z}^i = \pi_{\mathbb{P}_i^\perp}(\bar{x}^i) \in \mathbb{P}_i^\perp$ $(i = 1, \ldots, r)$. For $i = 1, \ldots, r$, define the mapping $\tau_i \in \{\mathbb{R}^n \to \mathbb{R}^l\}$ as follows. Let $(x_1, \ldots, x_n)$ be the coordinates of a vector $x \in \mathbb{R}^n$ in the orthonormal basis (11). Then,

$$\tau_i(x) = (x_{1+(i-1)l}, \ldots, x_{l+(i-1)l}). \tag{13}$$

Denote by $\bar{\tau}_i \colon \mathbb{P}_i \to \mathbb{R}^l$ the restriction of the mapping τ_i to subspace $\mathbb{P}_i \subset \mathbb{R}^n$. In the basis (11), an arbitrary vector has the following coordinates:

$$x = (0, \ldots, 0, x_{1+(i-1)l}, \ldots, x_{l+(i-1)l}, 0, \ldots, 0).$$

By comparing this with (13), we see that $\bar{\tau}_i$ is a one-to-one correspondence. Hence, $\bar{\tau}_i$ has the inverse mapping $\bar{\tau}_i^{-1}$. In the context of Eqs. (5) and (8), let us define the mapping $\varphi_i \in \{\mathbb{R}^n \to \mathbb{L}_i\}$ as

$$\varphi_i(x) = \bar{\tau}_i^{-1}\left(\tau_i(x) - \frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle\tau_i(a_j), \tau_i(x)\rangle + \left\langle a_j, \bar{z}^i\right\rangle - b_j, 0\right\}}{\|a_j\|^2} \cdot \tau_i(a_j)\right). \tag{14}$$

The following theorem shows that we can use Eq. (14) to calculate $\varphi_i(x)$ in Step 2 of algorithm $\mathfrak{S}$.

Theorem 1. *The mapping φ_i $(i = 1, \ldots, r)$ defined by Eq. (14) satisfies Eq. (9).*

Proof. From Eq. (5), we obtain

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}(x)\right)\right) = \pi_{\mathbb{L}_i}\left(\pi_{\mathbb{L}_i}(x) - \frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j, \pi_{\mathbb{L}_i}(x)\rangle - b_j, 0\right\}}{\|a_j\|^2} \cdot a_j\right).$$

By expanding the parentheses, we obtain

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}(x)\right)\right) = \pi_{\mathbb{L}_i}(x) - \pi_{\mathbb{L}_i}\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j, \pi_{\mathbb{L}_i}(x)\rangle - b_j, 0\right\}}{\|a_j\|^2} \cdot a_j\right).$$

According to Eq. (8), we have

$$\begin{aligned}&\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}(x)\right)\right)\\ &= \pi_{\mathbb{P}_i}(x) + \bar{z}_i - \left(\pi_{\mathbb{P}_i}\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j, \pi_{\mathbb{P}_i}(x) + \bar{z}_i\rangle - b_j, 0\right\}}{\|a_j\|^2} \cdot a_j\right) + \bar{z}_i\right).\end{aligned}$$

By expanding the parentheses and eliminating $\bar{z}_i$, we transform the equation above to the form

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)=\pi_{\mathbb{P}_i}\left(x\right)-\pi_{\mathbb{P}_i}\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j,\pi_{\mathbb{P}_i}\left(x\right)+\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot a_j\right).$$

By distributivity of the dot product over the addition, this is equivalent to the equation

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)$$
$$=\pi_{\mathbb{P}_i}\left(x\right)-\pi_{\mathbb{P}_i}\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j,\pi_{\mathbb{P}_i}\left(x\right)\rangle+\langle a_j,\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot a_j\right).$$

Transform now the right side of the last equation as follows:

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)$$
$$=\bar{\tau}_i^{-1}\left(\bar{\tau}_i\left(\pi_{\mathbb{P}_i}\left(x\right)-\pi_{\mathbb{P}_i}\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j,\pi_{\mathbb{P}_i}\left(x\right)\rangle+\langle a_j,\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot a_j\right)\right)\right).$$

Since the mapping $\bar{\tau}_i$ is linear, this implies that

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)$$
$$=\bar{\tau}_i^{-1}\left(\bar{\tau}_i\left(\pi_{\mathbb{P}_i}\left(x\right)\right)-\bar{\tau}_i\left(\pi_{\mathbb{P}_i}\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j,\pi_{\mathbb{P}_i}\left(x\right)\rangle+\langle a_j,\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot a_j\right)\right)\right).$$

By comparing the subscripts in (12) and (13), we find that $\bar{\tau}_i\left(\pi_{\mathbb{P}_i}\left(x\right)\right)=\tau_i\left(x\right)$. Applying this to the right side of the preceding equation, we obtain

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)$$
$$=\bar{\tau}_i^{-1}\left(\tau_i\left(x\right)-\tau_i\left(\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j,\pi_{\mathbb{P}_i}\left(x\right)\rangle+\langle a_j,\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot a_j\right)\right).$$

The mapping $\bar{\tau}_i$ is linear, therefore this means that

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)=$$
$$\bar{\tau}_i^{-1}\left(\tau_i\left(x\right)-\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle a_j,\pi_{\mathbb{P}_i}\left(x\right)\rangle+\langle a_j,\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot\tau_i\left(a_j\right)\right).$$

Let us compare again the subscripts in (12) and (13), we obtain

$$\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right)=\bar{\tau}_i^{-1}\left(\tau_i\left(x\right)-\frac{\lambda}{m}\sum_{j=1}^{m}\frac{\max\left\{\langle\tau_i(a_j),\tau_i(x)\rangle+\langle a_j,\bar{z}\rangle-b_j,0\right\}}{\left\|a_j\right\|^2}\cdot\tau_i\left(a_j\right)\right).$$

Finally, compare the last equation and Eq. (14), and we obtain that

$$\varphi_i\left(x\right)=\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x\right)\right)\right).$$

Q.E.D.

5 Convergence Theorem

We will prove now the convergence theorem for algorithm $\mathfrak{G}$. For this we will need the two lemmas given below. The first lemma shows that each mapping $\varphi_i \in \{\mathbb{L}_i \to \mathbb{L}_i\}$ constructed by algorithm $\mathfrak{G}$ is Fejerian for the set $\mathbb{L}_i \cap M$.

Lemma 1. *Consider a convex closed set $M \subset \mathbb{R}^n$ and a single-valued M-fejerian mapping $\varphi \in \{\mathbb{R}^n \to \mathbb{R}^n\}$. Let $\mathbb{P}$ be a proper linear subspace of the space $\mathbb{R}^n$, and suppose that $\mathbb{T} = \mathbb{P}^\perp$ is the orthogonal complement of the subspace $\mathbb{P}$. Assume that*

$$\bar{x} \in \operatorname*{Arg\,min}_{x \in M} \rho(\mathbb{P}, x).$$

Write $\bar{x}$ as a sum of orthogonal vectors taken from the subspaces $\mathbb{P}$ and $\mathbb{T}$:

$$\bar{x} = \pi_{\mathbb{P}}(\bar{x}) + \pi_{\mathbb{T}}(\bar{x}).$$

Denote $\bar{z} = \pi_{\mathbb{T}}(\bar{x}) \in \mathbb{T}$. Construct the linear manifold $\mathbb{L}$ as a translation of $\mathbb{P}$ by the vector $\bar{z}$:

$$\mathbb{L} = \mathbb{P} + \bar{z}.$$

Define the mapping $\varphi_{\mathbb{L}} \in \{\mathbb{L} \to \mathbb{L}\}$ as

$$\varphi_{\mathbb{L}}(x) = \pi_{\mathbb{L}}\left(\varphi\left(\pi_{\mathbb{L}}(x)\right)\right). \tag{15}$$

Take

$$M_{\mathbb{L}} = \mathbb{L} \cap M. \tag{16}$$

Then, the mapping $\varphi_{\mathbb{L}}$ is $M_{\mathbb{L}}$-fejerian.

Proof. Let us start by showing that

$$\varphi_{\mathbb{L}}(y) = y, \quad \forall y \in M_{\mathbb{L}}. \tag{17}$$

Let $y \in M_{\mathbb{L}}$. Then, by (16), $y \in M$. Since the mapping φ is M-fejerian, then $\varphi(y) = y$. Taking into account that $y \in \mathbb{L}$, we see that

$$\varphi_{\mathbb{L}}(y) = \pi_{\mathbb{L}}(\varphi(\pi_{\mathbb{L}}(y))) = \pi_{\mathbb{L}}(\varphi(y)) = \pi_{\mathbb{L}}(y) = y,$$

and so Eq. (17) holds.

Let us show now that

$$\|\varphi_{\mathbb{L}}(x) - y\| < \|x - y\|, \quad \forall y \in M_{\mathbb{L}}, \forall x \notin M_{\mathbb{L}}. \tag{18}$$

Assume that

$$y \in M_{\mathbb{L}},\ x \in \mathbb{L},\ x \notin M_{\mathbb{L}}.$$

By (16), it follows that $x \notin M$ in this case. Since the mapping φ is M-fejerian, then

$$\|\varphi(x) - y\| < \|x - y\|. \tag{19}$$

Construct the decomposition of $\varphi(x)$ and y as a sum of two orthogonal vectors belonging to $\mathbb{P}$ and $\mathbb{T}$:

$$\varphi(x) = \pi_{\mathbb{P}}(\varphi(x)) + \pi_{\mathbb{T}}(\varphi(x)), \tag{20}$$

$$y = \pi_{\mathbb{P}}(y) + \bar{z}. \tag{21}$$

We now substitute these decompositions into (19) and obtain

$$\|\pi_{\mathbb{P}}(\varphi(x)) + \pi_{\mathbb{T}}(\varphi(x)) - (\pi_{\mathbb{P}}(y) + \bar{z})\| < \|x - y\|, \tag{22}$$

which, after rearrangement, yields

$$\|(\pi_{\mathbb{P}}(\varphi(x)) - \pi_{\mathbb{P}}(y)) + (\pi_{\mathbb{T}}(\varphi(x)) - \bar{z})\| < \|x - y\|. \tag{23}$$

Note that $(\pi_{\mathbb{P}}(\varphi(x)) - \pi_{\mathbb{P}}(y)) \in \mathbb{P}$ and $(\pi_{\mathbb{T}}(\varphi(x)) - \bar{z}) \in \mathbb{T}$ are mutually orthogonal vectors. As we know, the square of the norm of a sum of orthogonal vectors is equal to the sum of the squares of their norms, so it follows from (23) that

$$\|\pi_{\mathbb{P}}(\varphi(x)) - \pi_{\mathbb{P}}(y)\|^2 + \|\pi_{\mathbb{T}}(\varphi(x)) - \bar{z}\|^2 < \|x - y\|^2. \tag{24}$$

The left side of the inequality is a sum of two non-negative terms. This means that, if we remove the second one, we obtain a valid inequality:

$$\|\pi_{\mathbb{P}}(\varphi(x)) - \pi_{\mathbb{P}}(y)\|^2 < \|x - y\|^2,$$

from which, in turn, we get

$$\|\pi_{\mathbb{P}}(\varphi(x)) - \pi_{\mathbb{P}}(y)\| < \|x - y\|. \tag{25}$$

By construction of $\mathbb{L}$, we have $\pi_{\mathbb{P}}(\varphi(x)) = \pi_{\mathbb{L}}(\varphi(x)) - \bar{z}$. Substitute this expression into (25), and we obtain

$$\|\pi_{\mathbb{L}}(\varphi(x)) - \bar{z} - \pi_{\mathbb{P}}(y)\| < \|x - y\|,$$

which is equivalent to

$$\|\pi_{\mathbb{L}}(\varphi(x)) - (\pi_{\mathbb{P}}(y) + \bar{z})\| < \|x - y\|. \tag{26}$$

But $x \in L$, so we may conclude that $x = \pi_{\mathbb{L}}(x)$. If we substitute the expression $\pi_{\mathbb{L}}(x)$ for x into the left side of (26), then we obtain

$$\|\pi_{\mathbb{L}}(\varphi(\pi_{\mathbb{L}}(x))) - (\pi_{\mathbb{P}}(y) + \bar{z})\| < \|x - y\|.$$

Taking into account (15) and (21), this implies

$$\|\varphi_{\mathbb{L}}(x) - y\| < \|x - y\|,$$

i.e., inequality (18) holds. Q.E.D.

To prove the convergence theorem, we need an additional lemma.

Lemma 2. *Let $\{x_k\}$ be the sequence of points produced by algorithm $\mathfrak{S}$ in Step 2:*

$$x_{k+1} := \sum_{i=1}^{r} \left({\varphi_i}^s (x_k) - \bar{z}^i \right);\ k = 0, 1, \ldots \tag{27}$$

Under the conditions of algorithm $\mathfrak{S}$, let us define

$$M_{\mathbb{L}_i} = \mathbb{L}_i \cap M \quad (i = 1, \ldots, r).$$

If

$$x_0^i = \pi_{\mathbb{L}_i}(x_0),\ x_{k+1}^i = \varphi_i(x_k^i), \tag{28}$$

then

$${\varphi_i}^s (x_k) = x_{s\cdot(k+1)}^i, \quad \forall k \in \mathbb{Z}_{\geqslant 0}. \tag{29}$$

Proof. Our proof will be by induction on k. Let $k = 0$. By Eq. (9), we have

$${\varphi_i}^s (x_0) = \varphi_i^{s-1} \left(\pi_{\mathbb{L}_i} \left(\varphi \left(\pi_{\mathbb{L}_i} (x_0) \right) \right) \right).$$

According to the first equation in (28), we obtain

$${\varphi_i}^s (x_0) = \varphi_i^{s-1} \left(\pi_{\mathbb{L}_i} \left(\varphi \left(x_0^i \right) \right) \right). \tag{30}$$

But $x_0^i \in \mathbb{L}_i$, so we may write $x_0^i = \pi_{\mathbb{L}_i}(x_0^i)$. Now we substitute the expression $\pi_{\mathbb{L}_i}(x_0^i)$ for x_0^i into (30), and obtain

$${\varphi_i}^s (x_0) = \varphi_i^{s-1} \left(\pi_{\mathbb{L}_i} \left(\varphi \left(\pi_{\mathbb{L}_i} \left(x_0^i \right) \right) \right) \right).$$

Taking (9) into account, we get

$${\varphi_i}^s (x_0) = {\varphi_i}^s \left(x_0^i \right).$$

By applying the second equation from (28), we obtain

$${\varphi_i}^s (x_0) = \varphi_i^{s-1} \left(x_1^i \right).$$

If we repeat this substitution another $(s - 1)$ times, we obtain

$${\varphi_i}^s (x_0) = x_s^i,$$

that is, the induction basis holds. Now let $k > 0$, and consider the trivial equation

$${\varphi_i}^s (x_k) = {\varphi_i}^s (x_k). \tag{31}$$

According to (27), $x_k := \sum_{j=1}^{r} \left(\varphi_j^s (x_{k-1}) - \bar{z}^j \right)$. Substitute this expression into the right side of (31) to obtain

$${\varphi_i}^s (x_k) = {\varphi_i}^s \left(\sum_{j=1}^{r} \left(\varphi_j^s (x_{k-1}) - \bar{z}^j \right) \right).$$

From this, by the induction hypothesis, it follows that

$$\varphi_i{}^s\left(x_k\right) = \varphi_i{}^s\left(\sum_{j=1}^{r}\left(x^j_{s\cdot k} - \bar{z}^j\right)\right),$$

which is equivalent to

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(\varphi_i\left(\sum_{j=1}^{r}\left(x^j_{s\cdot k} - \bar{z}^j\right)\right)\right).$$

Using now (9), we obtain

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(\sum_{j=1}^{r}\left(x^j_{s\cdot k} - \bar{z}^j\right)\right)\right)\right)\right).$$

According to (7) and (8), this implies

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{P}_i}\left(\sum_{j=1}^{r}\left(x^j_{s\cdot k} - \bar{z}^j\right)\right) + \bar{z}^i\right)\right)\right).$$

Remember that $x^j_{s\cdot k} - \bar{z}^j \in \mathbb{P}_j \quad (j = 1, \ldots, r)$ and $\mathbb{P}_i \perp \mathbb{P}_j$ for $i \neq j$. Then the last implies

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(\pi_{\mathbb{L}_i}\left(\varphi\left(x^i_{s\cdot k} - \bar{z}^i + \bar{z}^i\right)\right)\right),$$

i.e.

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(\pi_{\mathbb{L}_i}\left(\varphi\left(x^i_{s\cdot k}\right)\right)\right). \tag{32}$$

Since $x^i_{s\cdot k} \in \mathbb{L}_i$, we have $x^i_{s\cdot k} = \pi_{\mathbb{L}_i}\left(x^i_{s\cdot k}\right)$. Substitute the expression $\pi_{\mathbb{L}_i}\left(x^i_{s\cdot k}\right)$ for $x^i_{s\cdot k}$ into (32) and obtain

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(\pi_{\mathbb{L}_i}\left(\varphi\left(\pi_{\mathbb{L}_i}\left(x^i_{s\cdot k}\right)\right)\right)\right).$$

Together with Eq. (9), this implies

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\varphi_i\left(x^i_{s\cdot k}\right),$$

which is equivalent to

$$\varphi_i{}^s\left(x_k\right) = \varphi_i{}^s\left(x^i_{s\cdot k}\right).$$

By applying the second equation from (28), we obtain

$$\varphi_i{}^s\left(x_k\right) = \varphi_i^{s-1}\left(x^i_{s\cdot k+1}\right).$$

Repeating this substitution another $(s-1)$ times, we finally arrive at

$$\varphi_i{}^s\left(x_k\right) = x^i_{s\cdot k+s},$$

i.e.

$$\varphi_i{}^s\left(x_k\right) = x^i_{s(k+1)}.$$

Q.E.D.

Now we are ready to prove the *convergence theorem* for algorithm $\mathfrak{S}$.

Theorem 2. *Let $\{x_k\}$ be the sequence of points produced by algorithm $\mathfrak{S}$ in Step 2:*

$$x_{k+1} := \sum_{i=1}^{r} \left({\varphi_i}^s \left(x_k \right) - \bar{z}^i \right);\ k = 0, 1, \ldots \tag{33}$$

Then

$$\{x_k\}_{k=0}^{+\infty} \to \bar{x} \in M.$$

Proof. Under the conditions of algorithm $\mathfrak{S}$, let us define

$$M_{\mathbb{L}_i} = \mathbb{L}_i \cap M \quad (i = 1, \ldots, r).$$

According to Lemma 1, the mapping φ_i is $M_{\mathbb{L}_i}$-fejerian. Take

$$x_0^i = \pi_{\mathbb{L}_i}(x_0), \quad x_{k+1}^i = \varphi_i(x_k^i). \tag{34}$$

The continuity of the mappings $\pi_{\mathbb{L}_i}$ and φ (see [23]) implies the continuity of the mappings φ_i. Hence, by Lemma 39.1 in [23], we may affirm that

$$\{x_k^i\}_{k=0}^{+\infty} \to \bar{x}^i \in M_{\mathbb{L}_i},\ \forall i \in \{1, \ldots, r\}. \tag{35}$$

Now let us define

$$\bar{x} = \sum_{i=1}^{r} \left(\bar{x}^i - \bar{z}^i \right). \tag{36}$$

Fix an arbitrary real number $\varepsilon > 0$. By (35), there exists a number K_i such that

$$\left\| x_k^i - \bar{x}^i \right\| < \frac{\varepsilon}{\sqrt{r}},\ \forall k > K_i. \tag{37}$$

Let $K = \max\limits_{1 \leqslant i \leqslant r} K_i$. We will show that the inequality $\|x_k - \bar{x}\| < \varepsilon$ holds for any $k > K$. Fix an arbitrary $k > K$. By (33), we may write

$$\|x_k - \bar{x}\| = \left\| \left(\sum_{i=1}^{r} \left({\varphi_i}^s \left(x_{k-1} \right) - \bar{z}^i \right) \right) - \bar{x} \right\|,$$

and by (36), we obtain

$$\|x_k - \bar{x}\| = \left\| \sum_{i=1}^{r} \left({\varphi_i}^s \left(x_{k-1} \right) - \bar{z}^i \right) - \sum_{i=1}^{r} \left(\bar{x}^i - \bar{z}^i \right) \right\|.$$

After rearrangement, this yields

$$\|x_k - \bar{x}\| = \left\| \sum_{i=1}^{r} \left({\varphi_i}^s \left(x_{k-1} \right) - \bar{z}^i - \bar{x}^i + \bar{z}^i \right) \right\|,$$

which is equivalent to

$$\|x_k - \bar{x}\| = \left\| \sum_{i=1}^{r} \left({\varphi_i}^s \left(x_{k-1} \right) - \bar{x}^i \right) \right\|. \tag{38}$$

According to Lemma 2, this implies

$$\|x_k - \bar{x}\| = \left\| \sum_{i=1}^{r} \left(x^i_{s \cdot k} - \bar{x}^i \right) \right\|. \tag{39}$$

Remember now that $x^i_{s \cdot k} = \pi_{\mathbb{P}_i} \left(x^i_{s \cdot k} \right) + \bar{z}^i$ and $\bar{x}^i = \pi_{\mathbb{P}_i} \left(\bar{x}^i \right) + \bar{z}^i$, and substitute these expressions into (39):

$$\|x_k - \bar{x}\| = \left\| \sum_{i=1}^{r} \left(\pi_{\mathbb{P}_i} \left(x^i_{s \cdot k} \right) + \bar{z}^i - \pi_{\mathbb{P}_i} \left(\bar{x}^i \right) - \bar{z}^i \right) \right\|,$$

i.e.

$$\|x_k - \bar{x}\| = \left\| \sum_{i=1}^{r} \left(\pi_{\mathbb{P}_i} \left(x^i_{s \cdot k} \right) - \pi_{\mathbb{P}_i} \left(\bar{x}^i \right) \right) \right\|. \tag{40}$$

Note that both vectors under the summation sign in (40) are mutually orthogonal. The square of the norm of a sum of orthogonal vectors is equal to the sum of the squares of their norms. It thus follows from (40) that

$$\|x_k - \bar{x}\|^2 = \sum_{i=1}^{r} \left\| \pi_{\mathbb{P}_i} \left(x^i_{s \cdot k} \right) - \pi_{\mathbb{P}_i} \left(\bar{x}^i \right) \right\|^2.$$

This is equivalent to

$$\|x_k - \bar{x}\|^2 = \sum_{i=1}^{r} \left\| \pi_{\mathbb{P}_i} \left(x^i_{s \cdot k} \right) + \bar{z}^i - \pi_{\mathbb{P}_i} \left(\bar{x}^i \right) - \bar{z}^i \right\|^2.$$

Since $x^i_{s \cdot k} = \pi_{\mathbb{P}_i} \left(x^i_{s \cdot k} \right) + \bar{z}^i$ and $\bar{x}^i = \pi_{\mathbb{P}_i} \left(\bar{x}^i \right) + \bar{z}^i$, we obtain

$$\|x_k - \bar{x}\|^2 = \sum_{i=1}^{r} \left\| x^i_{s \cdot k} - \bar{x}^i \right\|^2.$$

In view of (37), this implies

$$\|x_k - \bar{x}\|^2 < \sum_{i=1}^{r} \left(\frac{\varepsilon}{\sqrt{r}} \right)^2 = r \left(\frac{\varepsilon}{\sqrt{r}} \right)^2 = \varepsilon^2,$$

i.e.

$$\|x_k - \bar{x}\| < \varepsilon.$$

Q.E.D.

6 Conclusion

A new iterative method for solving linear inequality systems is proposed in the article. This method is based on the operation of pseudoprojecting a point onto a polytope which is defined as the set of feasible solutions of a linear inequality system in Euclidean space. The pseudoprojection operation is an extension of the projection operation. It exploits Fejer iterative processes developed by Eremin in [14,15,23]. For an effective parallelization of the pseudoprojection algorithm, we suggest here the subvector method. Also, we proved the convergence theorem for the pseudoprojection algorithm. The algorithm that computes the pseudoprojection was implemented in C++ using the OpenMP parallel programming library. Computational experiments have confirmed the effectiveness of the proposed method of parallelization for computer systems using multi-core accelerators Intel Xeon Phi [16]. As future research, we intend to do the following: implement the pseudoprojection algorithm in C++ language using the MPI library and the BSF algorithmic skeleton [24]; perform an analytical and experimental evaluation of the scalability of this parallel program on cluster computing systems; compare the proposed algorithm with other parallel iterative algorithms by performing computational experiments on cluster computing systems.

References

1. Agmon, S.: The relaxation method for linear inequalities. Can. J. Math. **6**, 382–392 (1954). https://doi.org/10.4153/CJM-1954-037-2
2. Motzkin, T.S., Schoenberg, I.J.: The relaxation method for linear inequalities. Can. J. Math. **6**, 393–404 (1954). https://doi.org/10.4153/CJM-1954-038-x
3. Merzlyakov, Y.I.: On a relaxation method of solving systems of linear inequalities. USSR Comput. Math. Math. Phys. **2**, 504–510 (1963). https://doi.org/10.1016/0041-5553(63)90463-4
4. Bregman, L.M.: The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Math. Phys. **7**, 200–217 (1967). https://doi.org/10.1016/0041-5553(67)90040-7
5. Gubin, L.G., Polyak, B.T., Raik, E.V.: The method of projections for finding the common point of convex sets. USSR Comput. Math. Math. Phys. **7**, 1–24 (1967). https://doi.org/10.1016/0041-5553(67)90113-9
6. Germanov, M.A., Spiridonov, V.S.: On a method of solving systems of non-linear inequalities. USSR Comput. Math. Math. Phys. **6**, 194–196 (1966). https://doi.org/10.1016/0041-5553(66)90066-8
7. Goffin, J.L.: The relaxation method for solving systems of linear inequalities. Math. Oper. Res. **5**, 388–414 (1980). https://doi.org/10.1287/moor.5.3.388
8. González-Gutiérrez, E., Todorov, M.I.: A relaxation method for solving systems with infinitely many linear inequalities. Optim. Lett. **6**, 291–298 (2012). https://doi.org/10.1007/s11590-010-0244-4
9. González-Gutiérrez, E., Hernández Rebollar, L., Todorov, M.I.: Relaxation methods for solving linear inequality systems: converging results. TOP **20**, 426–436 (2012). https://doi.org/10.1007/s11750-011-0234-4

10. Mandel, J.: Convergence of the cyclical relaxation method for linear inequalities. Math. Program. **30**, 218–228 (1984). https://doi.org/10.1007/BF02591886
11. Konnov, I.: Combined Relaxation Methods for Variational Inequalities. LNE, vol. 495. Springer, Heidelberg (2001). https://doi.org/10.1007/978-3-642-56886-2
12. Konnov, I.V.: A modified combined relaxation method for non-linear convex variational inequalities. Optimization **64**, 753–760 (2015). https://doi.org/10.1080/02331934.2013.820298
13. Fejér, L.: Über die Lage der Nullstellen von Polynomen, die aus Minimumforderungen gewisser Art entspringen. In: Hilbert, D. (ed.) Festschrift, pp. 41–48. Springer, Heidelberg (1982). https://doi.org/10.1007/978-3-642-61810-9_6
14. Eremin, I.I.: Methods of Fejer's approximations in convex programming. Math. Notes Acad. Sci. USSR **3**, 139–149 (1968). https://doi.org/10.1007/BF01094336
15. Vasin, V.V., Eremin, I.I.: Operators and Iterative Processes of Fejér Type. Theory and Applications. Walter de Gruyter, Berlin, New York (2009)
16. Sokolinskaya, I., Sokolinsky, L.: Revised pursuit algorithm for solving non-stationary linear programming problems on modern computing clusters with many-core accelerators. In: Voevodin, V., Sobolev, S. (eds.) RuSCDays 2016. CCIS, vol. 687, pp. 212–223. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-55669-7_17
17. Sokolinskaya, I.M.: Scalable algorithm for non-stationary linear programming problems solving. In: 2017 2nd International Ural Conference on Measurements (UralCon), pp. 49–53 (2017). https://doi.org/10.1109/URALCON.2017.8120685
18. Sokolinskaya, I., Sokolinsky, L.B.: Scalability evaluation of NSLP algorithm for solving non-stationary linear programming problems on cluster computing systems. In: Voevodin, V., Sobolev, S. (eds.) RuSCDays 2017. CCIS, vol. 793, pp. 40–53. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71255-0_4
19. Sokolinskaya, I., Sokolinsky, L.B.: On the solution of linear programming problems in the age of big data. In: Sokolinsky, L., Zymbler, M. (eds.) PCT 2017. CCIS, vol. 753, pp. 86–100. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67035-5_7
20. Censor, Y., Elfving, T.: New methods for linear inequalities. Linear Algebra Appl. **42**, 199–211 (1982). https://doi.org/10.1016/0024-3795(82)90149-5
21. De Pierro, A.R., Iusem, A.N.: A simultaneous projections method for linear inequalities. Linear Algebra Appl. **64**, 243–253 (1985). https://doi.org/10.1016/0024-3795(85)90280-0
22. Yang, K., Murty, K.G.: New iterative methods for linear inequalities. J. Optim. Theory Appl. **72**, 163–185 (1992). https://doi.org/10.1007/BF00939954
23. Eremin, I.I.: Teoriya lineynoy optimizatsii [The theory of linear optimization]. Publishing House "Yekaterinburg", Ekaterinburg (1999). (in Russian)
24. Sokolinsky, L.B.: Analytical estimation of the scalability of iterative numerical algorithms on distributed memory multiprocessors. Lobachevskii J. Math. **39**, 571–575 (2018). https://doi.org/10.1134/S1995080218040121