

## Research Article

# A Method of Indefinite Krylov Subspace for Eigenvalue Problem

M. Aliyari and M. Ghasemi Kamalvand 

Department of Mathematics, Lorestan University, Khorramabad, Iran

Correspondence should be addressed to M. Ghasemi Kamalvand; ghasemi.m@lu.ac.ir

Received 7 November 2017; Revised 1 March 2018; Accepted 3 April 2018; Published 16 May 2018

Academic Editor: Raffaele Solimene

Copyright © 2018 M. Aliyari and M. Ghasemi Kamalvand. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We describe an indefinite state of Arnoldi's method for solving the eigenvalues problems. In the following, we scrutinize the indefinite state of Lanczos' method for solving the eigenvalue problems and we show that this method for the  $J$ -Hermitian matrices works much better than Arnoldi's method.

## 1. Introduction

Around the early 1950s the idea of Krylov subspace iteration was established by Cornelius Lanczos and Walter Arnoldi. Lanczos' method was based on two mutually orthogonal vector sequences and his motivation came from eigenvalue problems. In that context, the most prominent feature of the method is that it reduces the original matrix to tridiagonal form. Lanczos later applied his method to solve linear systems, in particular symmetric ones. Krylov subspace iterations or Krylov subspace methods are iterative methods which are used as linear system solvers and also iterative solvers of eigenvalue problems.

On the other hand, the indefinite inner product defined by  $J = \text{diag}(j_1, \dots, j_n)$ ,  $j_k \in \{-1, +1\}$ , arises frequently in applications. It is used, for example, in the theory of relativity and in the research of the polarized light. More on the applications of such products can be found in [1–6]. These applications in other fields of science inspired us to resume Lanczos, FOM, and Arnoldi's methods in this mentioned indefinite inner space. Indefinite Arnoldi's method is a process that constructs a  $J$ -orthonormal basis for the nondegenerated Krylov subspace; the basis that we proved have a particular common property, about the structure of the product of their vectors. In the following, iterative Arnoldi's method for eigenvalue problem has been renovated in the mentioned indefinite inner product space and also a process is made which is useful for solving eigenvalue problem with  $J$ -Hermitian coefficient matrices. Indefinite iterative Lanczos'

method for eigenvalue problem is the same as Lanczos' method that has been restored in the indefinite inner product space.

This paper is organized as follows: In Section 2, we recall the standard inner product and indefinite inner product in  $\mathbb{C}^n$ . In the following part of this section we show some examples on indefinite inner product that will guide us to the objectives of this article. We express the indefinite Arnoldi algorithm for the computation of a  $J$ -orthonormal basis of the Krylov subspace and describe property of the basis in the form of several propositions to review and in the end of this section we exhibit the indefinite version of the Hermitian Lanczos algorithm.

In Section 3, we present a definition of the Ritz eigenpair of the a matrix and new algorithms from the indefinite Arnoldi and Lanczos' algorithm that is named modified methods by the Ritz approximation in the indefinite Arnoldi and Lanczos' process. This is our main goal and initiative in this article.

Since the initial vectors are very important at the start of each iteration to the above algorithms, in this article, we consider the Ritz eigenvector instead of the common eigenvector in the start of the each repetition algorithm. In Section 4 we present numerical examples and compare the modified method by the Ritz approximation in the Arnoldi and indefinite Lanczos algorithm. Algorithms are written with personal computer using MATLAB software and we illustrate examples by using these codes.

## 2. Indefinite Arnoldi and Lanczos Algorithms

Let  $\mathbb{C}^n$  be the  $n$ -dimensional space consisting of all column vectors  $x$  with complex coordinates  $x_i$ ,  $i = 1, 2, \dots, n$ . The standard inner product in  $\mathbb{C}^n$  is denoted by  $(\cdot, \cdot)$ . Thus

$$(x, y) = \sum_{i=1}^n x_i \bar{y}_i, \quad (1)$$

where  $x = (x_1, \dots, x_n)^T$  and  $y = (y_1, \dots, y_n)^T$  and the bar denotes the complex conjugation. Remember that a function  $[\cdot, \cdot]$  from  $\mathbb{C}^n \times \mathbb{C}^n$  to  $\mathbb{C}$  is called an indefinite inner product in  $\mathbb{C}^n$  if the following axioms are satisfied:

(i) Linearity in the first argument:

$$[\alpha x_1 + \beta x_2, y] = \alpha [x_1, y] + \beta [x_2, y], \quad (2)$$

for all  $x_1, x_2, y \in \mathbb{C}^n$  and all complex numbers  $\alpha, \beta$ .

(ii) Antisymmetry:

$$[x, y] = \overline{[y, x]}. \quad (3)$$

*Example 1.* Let  $x = (x_1, \dots, x_n)^T$ ,  $y = (y_1, \dots, y_n)^T \in \mathbb{C}^n$  and define

$$[x, y] = \sum_{i=1}^r x_{\sigma(i)} \bar{y}_{\sigma(i)} - \sum_{i=r+1}^n x_{\sigma(i)} \bar{y}_{\sigma(i)}, \quad (4)$$

where  $\sigma$  is a permutation for which  $\sigma(i) = j_i$  and  $j_i \in \{1, \dots, n\}$ . It is easy to see that  $[\cdot, \cdot]$  is an indefinite inner product. Indeed, (i), (ii) properties are clearly confirmed. But for nondegeneracy property, if  $[x, y] = 0$  for all  $y \in \mathbb{C}^n$ , then in special case,  $[x, y] = 0$  for  $y = (y_1, \dots, y_n)^T$ , where

$$y_{\sigma(i)} = \begin{cases} x_{\sigma(i)} & i = 1, \dots, r; \\ -x_{\sigma(i)} & i = r+1, \dots, k, \end{cases} \quad (5)$$

by choosing an appropriate permutation  $\sigma$ .

Thus,

$$\begin{aligned} & \sum_{i=1}^r \bar{x}_{\sigma(i)} x_{\sigma(i)} - \left( - \sum_{i=r+1}^n \bar{x}_{\sigma(i)} x_{\sigma(i)} \right) \\ &= \sum_{i=1}^r |x_{\sigma(i)}|^2 + \sum_{i=r+1}^n |x_{\sigma(i)}|^2 = \sum_{i=1}^n |x_{\sigma(i)}|^2 = 0, \end{aligned} \quad (6)$$

and this yields that  $x_i = 0$ , for  $i = 1, \dots, n$ . So,  $x = 0$ .

By [3], we know that, for every  $n \times n$  invertible Hermitian matrix  $H$ , the formula

$$[x, y] = (Hx, y), \quad x, y \in \mathbb{C}^n \quad (7)$$

determines an indefinite inner product on  $\mathbb{C}^n$  and conversely for every indefinite inner product  $[\cdot, \cdot]$  on  $\mathbb{C}^n$  there exists an  $n \times n$  invertible and Hermitian matrix  $H$  such that (7) holds and the established correspondence  $[\cdot, \cdot] \leftrightarrow H$  is a bijection between the set of all indefinite inner products on  $\mathbb{C}^n$  and the set of all  $n \times n$  invertible Hermitian matrices.

*Example 2.* In Example (7), the corresponding nonsingular Hermitian matrix to that indefinite inner product is written in the form  $J = \text{diag}(\pm 1)$  wherein  $r$  is the number of  $+1$  and  $n - r$  is the number of  $-1$ . This is because if

$$[x, y] = (Jx, y), \quad (8)$$

then

$$\begin{aligned} [x, y] &= (Jx, y) = y^* Jx = (\bar{y}_1, \dots, \bar{y}_n)^T J (x_1, \dots, x_n) \\ &= \sum_{i=1}^r x_{\sigma(i)} \bar{y}_{\sigma(i)} - \sum_{i=r+1}^n x_{\sigma(i)} \bar{y}_{\sigma(i)}, \end{aligned} \quad (9)$$

and conversely if

$$[x, y] = \sum_{i=1}^r x_{\sigma(i)} \bar{y}_{\sigma(i)} - \sum_{i=r+1}^n x_{\sigma(i)} \bar{y}_{\sigma(i)}, \quad (10)$$

then it is clear that  $[x, y] = (Jx, y)$ , for all  $x, y \in \mathbb{C}^n$ .

*Note.* For  $x = (x_1, \dots, x_n)^T$ , consider the relation  $[x, x] = x^* Jx$ .

That the result becomes zero is a very special case and depends on how we choose the matrix  $J$ .

Recently, new algorithms have been presented in this regard and have been compared in terms of the number of replays and the time required to run algorithms.

These methods could solve the linear systems of equations with  $J$ -Hermitian coefficient matrices and also considered Arnoldi, FOM, and Lanczos' methods for solving linear system of equations and resumed these methods in a space that is equipped with a special indefinite inner product and proposed new algorithms to run these methods. The indefinite Arnoldi algorithm for the computation of a  $J$ -orthonormal basis of the Krylov subspace

$$\mathcal{K}_m(A, v_0) = \text{span} \{v_0, Av_0, A^2 v_0, \dots, A^{m-1} v_0\}, \quad (11)$$

where  $A$  is  $n \times n$  matrix and  $v_i$ 's are  $n \times 1$  vectors, is shown below:

- (1) Choose a vector  $x$  such that  $[x, x] \neq 0$
- (2) Define  $v_1 = x / \sqrt{|[x, x]|}$
- (3) For  $j = 1, \dots, m$  Do:
- (4) For  $i = 1, \dots, j$  Do:
- (5) Compute  $h_{ij} := [Av_j, v_i]$  and  $t(v_i) = [v_i, v_i]$
- (6) Compute  $w_j := Av_j - \sum_{i=1}^j t(v_i) h_{ij} v_i$
- (7)  $h_{j+1,j} = \sqrt{|[w_j, w_j]|}$
- (8) if  $h_{j+1,j} = 0$  then stop
- (9)  $v_{j+1} = w_j / h_{j+1,j}$
- (10) EndDo
- (11) EndDo

Here  $H_m$  is an upper Hessenberg matrix.

The following proposition expressed indefinite Arnoldi's algorithm. (You can see [7, 8].)

**Proposition 3.** Assume that the indefinite Arnoldi algorithm does not stop before the  $m$ th step. Then the vectors  $v_1, \dots, v_m$  form a  $J$ -orthonormal basis of the Krylov subspace  $\mathcal{K}_m(A, v)$ .

*Proof.* Consider the following expression:

$$h_{j+1,j}v_{j+1} = Av_j - \sum_{i=1}^j t(v_i)h_{ij}v_i, \quad j = 1, \dots, m. \quad (12)$$

Certainly that vectors  $v_1, \dots, v_m$  are  $J$ -orthonormal.  $\square$

**Proposition 4.** Define

- (i)  $H_m$ , the  $m \times m$  Hessenberg matrix whose nonzero entries  $h_{ij}$  are defined by indefinite Arnoldi's algorithm,
- (ii)  $V_m$ , the  $n \times m$  matrix with column vectors  $v_1, \dots, v_m$ ,
- (iii)  $\hat{J} = \text{diag}(t(v_1), \dots, t(v_m))$ .

Then the following relations are valid:

$$AV_m = V_m \hat{J} H_m + h_{m+1,m} v_{m+1} e_m^T, \quad (13)$$

$$V_m^* J AV_m = H_m. \quad (14)$$

In particular, if  $m = n$  then

$$V_n^{-1} AV_n = \hat{J} H_n. \quad (15)$$

*Proof.* By considering lines (4), (5), and (7) of indefinite Arnoldi's algorithm, relation (13) is verified straightforwardly. Indeed, in general, (13) is the matrix representation of (12):

$$\begin{aligned} AV_m &= V_m \hat{J} H_m + w_m e_m^T = V_m \hat{J} H_m + h_{m+1,m} v_{m+1} e_m^T \\ &= (V_m \hat{J} \mid v_{m+1}) \begin{pmatrix} H_m \\ 0, \dots, 0, h_{m+1,m} \end{pmatrix}. \end{aligned} \quad (16)$$

Now, to see (14), left-multiply the relation (16) in  $V_m^* J$ . We earn

$$V_m^* J AV_m = V_m^* J V_m \hat{J} H_m + h_{m+1,m} V_m^* J v_{m+1} e_m^T. \quad (17)$$

On the other hand, given that the vectors  $v_1, \dots, v_m$  build a  $J$ -orthonormal basis, then, we have the following.

- (1) According to the definition of  $v_{m+1}$ ,  $v_{m+1} = 0$  or it is orthogonal to  $v_1, \dots, v_m$ , that is,  $[v_{m+1}, v_i] = v_i^* J v_{m+1} = 0$ , for  $i = 1, \dots, m$ . Thus,

$$V_m^* J v_{m+1} = 0. \quad (18)$$

- (2) We have

$$\begin{aligned} V_m^* J V_m &= \left( [v_i, v_j] \right)_{i,j} = \text{diag}([v_1, v_1], \dots, [v_m, v_m]) \\ &= \hat{J}. \end{aligned} \quad (19)$$

In other words,  $\hat{J} V_m^* J V_m = I$ .

Therefore, relation (17) can be summarized as follows:

$$V_m^* J AV_m = H_m, \quad (20)$$

and, by left-multiplying in  $\hat{J}$ ,

$$\hat{J} V_m^* J AV_m = \hat{J} H_m. \quad (21)$$

Particularly, if  $m = n$ , relation (19) yields that  $\hat{J} V_n^* J = V_n^{-1}$  and thereby

$$V_n^{-1} AV_n = \hat{J} H_n. \quad (22)$$

$\square$

*Remark 5.* It should be noted that a permutation matrix  $P$  is available such That

$$PJP = \hat{J}. \quad (23)$$

We know that the indefinite Hermitian Lanczos algorithm is a simplified version of the indefinite Arnoldi algorithm that is applied to  $J$ -Hermitian matrices. The  $m$ th step of the algorithm transforms matrix  $A$  into a Hessenberg and  $J$ -Hermitian matrix, in other words, a tridiagonal matrix  $T_m$ ; when  $m$  is equal to the dimension of  $A$ ,  $T_m$  is similar to  $A$ . To see more, refer to [7–10]. As can be seen in the following, this method is expressed as a special case of the indefinite Arnoldi method in the complex space for the special case when matrix  $A$  is  $J$ -Hermitian.

The indefinite version of the Hermitian Lanczos algorithm can be formulated as follows:

- (1) Choose an initial vector  $v_1$  such that  $\sqrt{|[v_1, v_1]|} = \pm 1$
- (2) Set  $\beta_1 \equiv 0, v_0 = 0$
- (3) For  $j = 1, \dots, m$  Do:
- (4)  $\omega_j := Av_j - t(v_{j-1})\beta_j v_{j-1}$
- (5)  $\alpha_j := [\omega_j, v_j]$
- (6)  $\omega_j := \omega_j - t(v_j)\alpha_j v_j$
- (7) if  $\sqrt{|[\omega_j, \omega_j]|} = 0$  then stop
- (8)  $v_{j+1} = \omega_j / \sqrt{|[\omega_j, \omega_j]|}$
- (9)  $\beta_{j+1} := t(v_{j+1})\sqrt{|[\omega_j, \omega_j]|}$
- (10) EndDo

### 3. Indefinite Iterative Arnoldi's Method for Eigenvalue Problem

*Definition 6.* Suppose that  $A, J', H_m$ , and  $V_m$  are the same as those in Proposition (8), let  $(\lambda_i^{(m)}, y_i^{(m)})$  be an eigenpair of  $H_m$ ,  $\lambda_i^{(m)}$  is called Ritz value of  $A$  and provides an approximation for eigenvalue of  $A$ .

$u_i^{(m)} = V_m y_i^{(m)}$  is called Ritz vector of  $A$  and provides an approximation for eigenvector of  $A$ .

**Proposition 7.** Let  $y_i^{(m)}$  be an eigenvector of  $J^l H_m$  associated with the eigenvalue  $\lambda_i^{(m)}$  and  $u_i^{(m)}$  the Ritz approximate eigenvector, so that  $u_i^{(m)} = V_m y_i^{(m)}$ . Then,

$$(A - \lambda_i^{(m)} I) u_i^{(m)} = h_{m+1,m} e_m^T y_i^{(m)} v_{m+1}. \quad (24)$$

*Proof.* Let  $u_i^{(m)} = V_m y_i^{(m)}$  and  $J_m^l H_m y_i^{(m)} = \lambda_i^{(m)} y_i^{(m)}$ . We have

$$\begin{aligned} AV_m y_i^{(m)} &= (V_m J_m^l H_m + h_{m+1,m} v_{m+1} e_m^T) y_i^{(m)} \\ AV_m y_i^{(m)} &= V_m J_m^l H_m y_i^{(m)} + h_{m+1,m} v_{m+1} e_m^T y_i^{(m)} \\ (AV_m - \lambda_i^{(m)} V_m) y_i^{(m)} &= h_{m+1,m} (e_m^T y_i^{(m)}) v_{m+1} \\ (A - \lambda_i^{(m)} I) V_m y_i^{(m)} &= h_{m+1,m} (e_m^T y_i^{(m)}) v_{m+1} \\ (A - \lambda_i^{(m)} I) u_i^{(m)} &= h_{m+1,m} (e_m^T y_i^{(m)}) v_{m+1}. \end{aligned} \quad (25)$$

□

One way to circumvent the difficulty is to restart the algorithm. After a run with  $m$  Arnoldi and Lanczos' vectors, we compute the approximate eigenvector and use it as an initial vector for the next run with Arnoldi and Lanczos' methods. This process, which is the simplest of this kind, is iterated to convergence. The iterative indefinite Arnoldi algorithm can be formulated as follows:

- (1) Start: choose an initial vector  $v_1$  and a dimension  $m$ .
- (2) Iterate: perform  $m$  steps of indefinite Arnoldi's algorithm.
- (3) Restart: compute the approximate eigenvector  $u_1^{(m)}$  associated with the rightmost eigenvalue  $\lambda_1^{(m)}$ .

If satisfied stop; else set  $v_1 \equiv u_1^{(m)}$  and go to (2).

Of course, it should be noted that this algorithm is not beneficial for us, because in any case it is better to use the definite algorithm.

In the following we have the same iterative indefinite Lanczos algorithm which can be formulated as follows:

- (1) Start: choose an initial vector  $v_1$  and a dimension  $m$ .

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} & A_{17} & A_{18} & A_{19} & A_{1,10} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} & A_{27} & A_{28} & A_{29} & A_{2,10} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & A_{36} & A_{37} & A_{38} & A_{39} & A_{3,10} \\ A_{41} & A_{42} & A_{43} & A_{44} & A_{45} & A_{46} & A_{47} & A_{48} & A_{49} & A_{4,10} \\ A_{51} & A_{52} & A_{53} & A_{54} & A_{55} & A_{56} & A_{57} & A_{58} & A_{59} & A_{5,10} \\ A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & A_{66} & A_{67} & A_{68} & A_{69} & A_{6,10} \\ A_{71} & A_{72} & A_{73} & A_{74} & A_{75} & A_{76} & A_{77} & A_{78} & A_{79} & A_{7,10} \\ A_{81} & A_{82} & A_{83} & A_{84} & A_{85} & A_{86} & A_{87} & A_{88} & A_{89} & A_{8,10} \\ A_{91} & A_{92} & A_{93} & A_{94} & A_{95} & A_{96} & A_{97} & A_{98} & A_{99} & A_{9,10} \\ A_{10,1} & A_{10,2} & A_{10,3} & A_{10,4} & A_{10,5} & A_{10,6} & A_{10,7} & A_{10,8} & A_{10,9} & A_{10,10} \end{pmatrix} \quad (27)$$

TABLE 1

Method	Iteration	Time	$e$
Arnoldi	1	0.31	$6 \times 10^{-14}$
Indefinite Lanczos	1	0.28	$1 \times 10^{-13}$

Here  $e = \|(A - \lambda_i^{(m)} I) u_i^{(m)}\|_2$ .

(2) Iterate: perform  $m$  steps of indefinite Lanczos' algorithm.

(3) Restart: compute the approximate eigenvector  $u_1^{(m)}$  associated with the rightmost eigenvalue  $\lambda_1^{(m)}$ .

If satisfied stop; else set  $v_1 \equiv u_1^{(m)}$  and go to (2).

This algorithm is used for  $J$ -symmetric matrices, since the number of flops in the performance of algorithm is certainly less than that of the Arnoldi algorithm. This is illustrated by the examples in the next section.

## 4. Numerical Example

In this section, we present numerical examples and compare the methods modified by the Ritz approximation in the Arnoldi and indefinite Lanczos process.

Algorithms are written with personal computer using MATLAB software and the following examples are done using these codes.

*Example 1.* Consider the matrices  $J = \begin{pmatrix} I & 0 \\ 0 & -I \end{pmatrix}$  and  $A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$ , where  $A_{11}, A_{22}$  are diagonal matrices with random elements in  $(0, 1)$  and also  $A_{12}, A_{21}$  are tridiagonal matrices with random elements in  $(0, 1)$  such that  $A_{12} = -A_{21}^T$  and  $v$  is  $n \times 1$  vector with random entries in  $(0, 1)$ .

Let  $v_1 = v / \sqrt{|v|}$ ; in this case  $n = 100, m = 50$ . The results are in Table 1.

*Example 2.* Consider the matrices

$$J = (I_{n/10}, -I_{n/10}, I_{n/10}, -I_{n/10}, I_{n/10}, -I_{n/10}, I_{n/10}, -I_{n/10}, I_{n/10}, -I_{n/10}), \quad (26)$$

where  $I$  is identity matrix and  $A \in M_n$  in the form

TABLE 2

Method	Iteration	Time	$e$
Arnoldi	5	1.25	$3.2 \times 10^{-8}$
Indefinite Lanczos	3	0.64	$4.3 \times 10^{-6}$

TABLE 3

Method	Iteration	Time	$e$
Arnoldi	1	0.27	$4.6 \times 10^{-10}$
Indefinite Lanczos	1	0.14	$2.6 \times 10^{-8}$

Here  $e = \|(A - \lambda_i^{(m)}I)u_i^{(m)}\|_2$ .

such that

$$\begin{aligned}
 -A_{i,i+1}^T &= A_{i+1,i} & \text{for } i = 1 : 9, \\
 A_{i,i+2}^T &= A_{i+2,i} & \text{for } i = 1 : 8 \\
 -A_{i,i+3}^T &= A_{i+3,i} & \text{for } i = 1 : 7, \\
 A_{i,i+4}^T &= A_{i+1,i} & \text{for } i = 1 : 6 \\
 -A_{i,i+5}^T &= A_{i+5,i} & \text{for } i = 1 : 5, \\
 A_{i,i+6}^T &= A_{i+6,i} & \text{for } i = 1 : 4 \\
 -A_{i,i+7}^T &= A_{i+7,i} & \text{for } i = 1 : 3, \\
 A_{i,i+8}^T &= A_{i+8,i} & \text{for } i = 1 : 2 \\
 -A_{i,i+9}^T &= A_{i+9,i} & \text{for } i = 1,
 \end{aligned} \tag{28}$$

wherein  $A_{ij}$ ,  $i, j = 1 : 10$ , are diagonal matrices with random elements (entries) in  $(0, 1)$ .

In this case,

(i) if  $n = 50, m = 30$ , the results are in Table 2;

(ii) if  $n = 200, m = 50$ ; the results are in Table 3.

It should be noted that obviously the number of external iterations depends on the number of internal iterations selected. The larger the  $m$  value is taken, the more the external iterations will decrease.

### 5. Conclusion

We know that the matrices of  $J$ -Hermitian are important and functional matrices. We can find a way to find the eigenvalues of these types of matrices that are more efficient than Arnoldi's method. To do this, we first constructed an indefinite Arnoldi method in order to be able to also make that indefinite Lanczos method. We tried to study and compare the Arnoldi and indefinite Lanczos methods. It was observed that Arnoldi's method is not efficient in indefinite status, but the indefinite Lanczos method for  $J$ -symmetric (Hermitian) matrices is much better than Arnoldi's method.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### References

- [1] K. Appi Reddy and T. Kurmayya, "Moore-Penrose inverses of Gram matrices leaving a cone invariant in an indefinite inner product space," *Special Matrices*, vol. 3, pp. 155–162, 2015.
- [2] N. J. Higham, "J-orthogonal matrices: properties and generation," *SIAM Review*, vol. 45, no. 3, pp. 504–519, 2003.
- [3] A. Kihçman and Z. A. Zhou, "The representation and approximation for the weighted Minkowski inverse in Minkowski space," *Mathematical and Computer Modelling*, vol. 47, no. 3-4, pp. 363–371, 2008.
- [4] B. C. Levy, "A note on the hyperbolic singular value decomposition," *Linear Algebra and its Applications*, vol. 277, no. 1-3, pp. 135–142, 1998.
- [5] R. Onn, A. O. Steinhardt, and A. Bojanczyk, "The hyperbolic singular value decomposition and applications," in *proceedings of the Applied Mathematics and Computing, Trans. 8th Army Conf.*, 108, 93 pages, NY, USA, 1991, ARO Rep. 91-1.
- [6] X. Sui and P. Gondolo, "Factorizations into Normal Matrices in Indefinite Inner Product Spaces," *Rings and Algebras (math.RA); Numerical Analysis (math.NA)*, 31, Oct, 2016.
- [7] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2nd edition, 2003.
- [8] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, 2nd edition, 2011.
- [9] A. de la Garza, "An iterative method for solving systems of linear equations," Union Carbide and Carbon Corb, 1951, K-25 plant, Oak Ridge, Tennessee, Report K-731.
- [10] I. Gohberg, P. Lancaster, and L. Rodman, *Indefinite linear algebra and applications*, Birkhauser, 2005.

