ScientificScholar®
Knowledge is power
Publisher of Scientific Journals

https://jksus.org

Journal of King Saud University – Science

2025 37 (1) 142024

Original Article

The improved preconditioned iterative integration-exponential method for linear ill-conditioned problems

Jingtao Sun[a], Jiulong Cheng[a,*], Lei Qin[b]

[a]State Key Laboratory for Fine Exploration and Intelligent Development of Coal Resources, China University of Mining and Technology-Beijing, Haidian District, Beijing-100000, China
[b]School of Civil Engineering, Sun Yat-sen University, Tangjiawan, Zhuhai City, Guangdong Province, Zhuhai – 519000, Guangdong, China

ARTICLE INFO

Keywords:
First order dynamical systems
Ill-conditioned problems
Iterative integration-exponential
Precise integration method

ABSTRACT

The preconditioned iterative integration-exponential method is an innovative iterative regularization technique designed to solve ill-conditioned linear problems. However, the preconditioned iterative integration-exponential method has been primarily applied to symmetric positive definite problems, and a notable limitation is its inability to adaptively determine the optimal number of iterations. To overcome this limitation, the present study demonstrates that the preconditioned iterative integration-exponential method can also be effectively applied to nonsymmetric positive definite linear systems. Furthermore, an improved preconditioned iterative integration-exponential method is proposed by combining the iterative refinement algorithm with the original approach. Addressing the challenge of adaptively determining the optimal number of iterations and Krylov subspace can solve the problem of low computational efficiency of the improved preconditioned iterative integration-exponential method in dealing with large-scale and sparse problems. Numerical results show that the newly proposed method is more robust than the original one.

1. Introduction

In recent years, ill-conditioned problems have attracted more and more attention and have been widely used in engineering and mathematics fields, such as geodesy (Yu et al., 2023), geophysical exploration (Li et al., 2024), and signal and processing (Wu, 2012; Yang and Deng, 2017). The solution methods of ill-conditioned equations have research significance.

The ill-conditioned system can be expressed as:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{1}$$

where $\boldsymbol{A} \in R^{n \times n}$ is an ill-conditioned matrix, $\boldsymbol{x}$ is the solution, and $\boldsymbol{b}$ is the observation. For an ill-conditioned system, a small disturbance in $\boldsymbol{b}$ or $\boldsymbol{A}$ can result in a significantly larger change in the solution $\boldsymbol{x}$. This phenomenon presents considerable challenges when attempting to solve the system (1) numerically. Consequently, the application of conventional numerical methods for solving such systems (1) may prove ineffective. To address this issue, several iterative regularization methods have been developed and widely used, including Tikhonov regularization (Benning and Burger, 2018; Chang et al., 2024; W. Wang et al., 2024) (TR), Landweber iteration (Mittal and Giri, 2021), and direct regularization methods like truncated singular value decomposition (Xu et al., 2024; Xu, 1998) (TSVD), truncated generalized singular value decomposition (Chen and Chan, 2017), and truncated randomized singular value decomposition (Bai et al., 2021; Huang et al., 2022). A notable characteristic of these regularization methods is that their performance depends on various regularization parameters, such as the truncation order in TSVD, the Tikhonov regularization parameter, and the number of iterations in iterative regularization methods. In recent years, iterative regularization methods for ill-conditioned equations that are based on the numerical solutions of dynamic systems have garnered attention (Dmytryshyn et al., 2022; Edvardsson et al., 2015).

The exploration of the connections between iterative numerical methods and continuous dynamical systems often deepens the understanding of iterative numerical techniques. Additionally, this investigation aids in the advancement of enhanced iterative numerical methods by utilizing numerical approaches for ordinary differential equations and continuous dynamical systems (Chu, 1988; Miyatake et al., 2018). For solving ill-conditioned linear systems, Ramm (2004) developed the dynamical systems method. Wu (2002) analyzed the relationship between Wilkinson iteration method and Euler method and proposed a new iterative improved solution method to solve the problem of ill-conditioned linear equations. Enlightened by Wu's work, Salkuyeh et al. (2011) proposed a new two-step iterative method. Beik et al. (2018) presented a generalized two-step iterative method. For symmetric positive definite linear ill-conditioned systems, Hoang and Ramm (2010) proposed a gradient algorithm for dynamic systems based on SVD decomposition, Wu (2023) introduced an exponential approximation method.

The precise integration method, proposed by Zhong (2004), is designed to calculate the matrix exponential. It is known for its high efficiency and precision, often yielding results nearly equivalent to exact solutions on a computer. As a result, it is widely used in solving various dynamic systems (J. Wang et al., 2024; Yang et al., 2021; Zhu et al., 2023, 2021). Fu and Zhang (2011) introduced a precise integration method for solving positive definite ill-conditioned linear systems and

***Corresponding author:**
E-mail address: jlcheng@cumtb.edu.cn (J. Cheng).

Received: 14 September, 2024 **Accepted:** 7 January, 2025 **Epub Ahead of Print:** 28 February, 2025 **Published:** 22 March, 2025

DOI: 10.25259/JKSUS_14_2024

proposed an iterative integration-exponential (IIE) method. Huang et al. (2021) gave rigorous proof of the convergence of the IIE method and discovered that: (1) the IIE method is an iterative regularization method similar to the TR method; (2) the Landweber iterative algorithm can be seen as a special form of the IIE method; (3) The number of iterations can be equivalent to the regularization parameter of the IIE method. In order to improve the precision of calculation and reduce the number of iterations, a preconditioned IIE (PIIE) method (Fu and Li, 2018) was proposed. The PIIE method has demonstrated effective results in addressing ill-conditioned systems, particularly in geoscience (Kailiang et al., 2023). However, determining the optimal number of iterations is an area requiring further research.

To solve this problem, this paper proposes an improved preconditioned precise integration iterative (IPIIE) method. The new method uses the residual error to correct the error of the solution of the original PIIE method, which avoids the selection of the optimal iteration termination parameter.

2. A brief review of the PIIE method

2.1 The IIE method

Multiply the transpose of the matrix $\boldsymbol{A}$ on both sides of Eq. (1).

$$\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}^T\boldsymbol{b} \tag{2}$$

Consider the following linear dynamical systems

$$\begin{aligned} &\frac{d\boldsymbol{x}(t)}{dt} = -\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}^T\boldsymbol{b} \\ &\boldsymbol{x}(0) = \boldsymbol{x}_0 \end{aligned} \tag{3}$$

Hoang and Ramm (2010) transformed solving linear eqs. (2) into solving linear dynamical system (3). The solution of a linear dynamical system (3) can be expressed as

$$\boldsymbol{x}(t) = \exp(-\boldsymbol{A}^T\boldsymbol{A}t)\boldsymbol{x}_0 + \int_0^t \exp(-\boldsymbol{A}^T\boldsymbol{A}(t-\tau))d\tau\boldsymbol{A}^T\boldsymbol{b} \tag{4}$$

Here, the $\boldsymbol{A}^T\boldsymbol{A}$ is a symmetric positive definite matrix. When t approaches infinity and $\boldsymbol{x}_0$ is zero vector, $\boldsymbol{x}(t)$ can be expressed as

$$\begin{aligned} \lim_{t\to\infty}\boldsymbol{x}(t) &= \lim_{t\to\infty}\int_0^t \exp(-\boldsymbol{A}^T\boldsymbol{A}t)dt\boldsymbol{A}^T\boldsymbol{b} \\ &= \lim_{t\to\infty}\boldsymbol{F}(t)\boldsymbol{A}^T\boldsymbol{b} \\ &= (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{b} \end{aligned} \tag{5}$$

where $\boldsymbol{F}(t) = \int_0^t -\boldsymbol{A}^T\boldsymbol{A}tdt$, Let $\tau > 0$ be small enough and set $t = 2^{k+1}\tau$

$$\begin{aligned} &\int_0^{2^{k+1}\tau}\exp(-\boldsymbol{A}^T\boldsymbol{A}t)dt \\ &= \int_0^{2^k\tau}\exp(-\boldsymbol{A}^T\boldsymbol{A}t)dt + \int_{2^k\tau}^{2^{k+1}\tau}\exp(-\boldsymbol{A}^T\boldsymbol{A}t)dt \\ &= \int_0^{2^k\tau}\exp(-\boldsymbol{A}^T\boldsymbol{A}t)dt + \exp(-\boldsymbol{A}^T\boldsymbol{A}2^k\tau)\int_0^{2^k\tau}\exp(-\boldsymbol{A}^T\boldsymbol{A}t)dt \end{aligned} \tag{6}$$

then the $\boldsymbol{F}(t)$ can be written in the following iterative form:

$$\boldsymbol{F}(2^{k+1}\tau) = \boldsymbol{F}(2^k\tau) + \exp(-\boldsymbol{A}^T\boldsymbol{A}2^k\tau)\boldsymbol{F}(2^k\tau) \tag{7}$$

To take this further, set

$$\boldsymbol{x}^k = \boldsymbol{F}(2^k\tau)\boldsymbol{A}^T\boldsymbol{b} \tag{8}$$

and by combining Eqs. (7) and (8), an iterative formula can be derived as follows.

$$\begin{aligned} \boldsymbol{x}^{k+1} &= (\boldsymbol{F}(2^k\tau) + \exp(-\boldsymbol{A}^T\boldsymbol{A}2^k\tau)\boldsymbol{F}(2^k\tau))\boldsymbol{A}^T\boldsymbol{b} \\ &= \boldsymbol{x}^k + \exp(-\boldsymbol{A}^T\boldsymbol{A}2^k\tau)\boldsymbol{x}^k \end{aligned} \tag{9}$$

In the IIE method, the matrix exponential $\exp(-(\boldsymbol{A}^T\boldsymbol{A})2^k\tau)$ in Eq. (9) is computed through precise integration. By Taylor expansion, it is obtained that

$$\begin{aligned} \exp(-(\boldsymbol{A}^T\boldsymbol{A})\tau) \approx \boldsymbol{B}_p(\tau) = \boldsymbol{I}\tau - (\boldsymbol{A}^T\boldsymbol{A})\tau + \\ \frac{(\boldsymbol{A}^T\boldsymbol{A})^2\tau^2}{2!} + \cdots + (-1)^p\frac{(\boldsymbol{A}^T\boldsymbol{A})^p\tau^p}{p!} \end{aligned} \tag{10}$$

$$\begin{aligned} &\boldsymbol{F}(\tau) = (\boldsymbol{A}^T\boldsymbol{A})^{-1}(\boldsymbol{I} - \exp(-(\boldsymbol{A}^T\boldsymbol{A}))) \\ &\approx \boldsymbol{F}_p(\tau) = (\boldsymbol{A}^T\boldsymbol{A})^{-1}(I - \boldsymbol{B}_p(\tau)) \\ &= \boldsymbol{I}\tau - \frac{(\boldsymbol{A}^T\boldsymbol{A})\tau^2}{2!} + \frac{(\boldsymbol{A}^T\boldsymbol{A})^2\tau^3}{3!} + \cdots + (-1)^p\frac{(\boldsymbol{A}^T\boldsymbol{A})^{p-1}\tau^p}{4!} \end{aligned} \tag{11}$$

here, p denotes the truncation order of Taylor expansion, and $\boldsymbol{I}$ denotes the identity matrix. Consequently, $\exp(-(\boldsymbol{A}^T\boldsymbol{A})2^k\tau)$ in Eq. (7) can be calculated using the following formula:

$$\begin{aligned} &\boldsymbol{T}_p^{(0)} \approx \boldsymbol{B}_p(\tau) - \boldsymbol{I} \\ &\textit{for } i = 1\cdots \text{k} \\ &\quad \boldsymbol{T}_p^{(i)} = 2\boldsymbol{T}_p^{(i-1)} + \boldsymbol{T}_p^{(i-1)}\boldsymbol{T}_p^{(i-1)} \\ &\textit{end} \\ &\boldsymbol{T}^{(k)} = \boldsymbol{I} + \boldsymbol{T}_p^{(k)} \end{aligned} \tag{12}$$

where $\boldsymbol{T}^{(k)}$ is the approximate value of $\exp(-(\boldsymbol{A}^T\boldsymbol{A})2^k\tau)$. By combining Eq. (5), (7) and (12), the IIE method for solving linear equations can be expressed in the following form:

$$\begin{cases} \boldsymbol{x}^{k+1} = 2\boldsymbol{x}^k + \boldsymbol{T}_p^{(k)}\boldsymbol{x}^k \\ \boldsymbol{T}_p^{(k+1)} = 2\boldsymbol{T}_p^{(k)} + \boldsymbol{T}_p^{(k)}\boldsymbol{T}_p^{(k)} \end{cases} \tag{13}$$

where the initial value is $\boldsymbol{x}^0 = \boldsymbol{F}_p(\tau)\boldsymbol{A}^T\boldsymbol{b}$. For IIE method, the iteration number k is considered as the regularization parameter.

2.2 The PIIE method and termination iteration condition

In order to accelerate the convergence speed of IIE, the PIIE method uses 1-norm normalization to reduce the condition number of the coefficient matrix.

Multiplying the preconditioned matrix $\boldsymbol{Q}$ on both sides of the Eq. (2)

$$\boldsymbol{Q}\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} = \boldsymbol{Q}\boldsymbol{A}^T\boldsymbol{b} \tag{14}$$

The definition of the matrix $\boldsymbol{Q}$ is as follows:

$$\boldsymbol{Q} = \begin{bmatrix} Q_1 & 0 & \cdots & 0 \\ 0 & Q_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q_n \end{bmatrix} \quad Q_n = \frac{1}{\sum_{j=1}^{n}|a_{ij}|} \tag{15}$$

where a_{ij} is the element of the matrix $\boldsymbol{A}^T\boldsymbol{A}$.

Fu et al. (2018) proposed an iterative termination condition for the PIIE algorithm. The error of the iteration is defined as

$$err^{(k)} = \|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\| \tag{16}$$

and define the following function:

$$\begin{aligned} &P(k) = \begin{cases} 1 & err^k / err^{k-1} \geq 1 \\ 0 & err^k / err^{k-1} < 1 \end{cases} \\ &q_n = \prod_{k}^{k+n} P(k) \end{aligned} \tag{17}$$

The parameter $n \in [2,10]$ is an integer. When $q_n = 1$, the iteration ends.

Algorithm 1: The PIIE algorithm

```
Compute the preconditioned matrix $\boldsymbol{Q}$, $\boldsymbol{A} = \boldsymbol{QA}, \boldsymbol{b} = \boldsymbol{Qb}$.
Compute ${\boldsymbol{T}_p}^{(0)}(\tau)$ and $\boldsymbol{F}(\tau)$.
Compute the initial value $\boldsymbol{x}_0 = \boldsymbol{F}(\tau) \cdot \boldsymbol{b}$
while satisfying termination conditions;
    $\boldsymbol{x}^{k+1} = 2\boldsymbol{x}^k + {\boldsymbol{T}_p}^{(k)}(\tau)\boldsymbol{x}^k$
    ${\boldsymbol{T}_p}^{(k+1)} = 2{\boldsymbol{T}_p}^{(k)} + {\boldsymbol{T}_p}^{(k)2}$
end
```

2.3 The PIIE method for nonsymmetric positive definite linear system

The PIIE method is suitable for symmetric positive definite matrix linear systems (Huang et al., 2021). In this section, we extend the proof to demonstrate that the PIIE method is also applicable to solving nonsymmetric positive definite linear systems. Consider the following linear system

$$\boldsymbol{Ax} = \boldsymbol{b} \tag{18}$$

where $\boldsymbol{A}$ is a nonsymmetric positive real definite matrix.

Theorem 1. Let $\boldsymbol{A} \in R^{n\times n}$ be a nonsymmetric positive definite matrix, t be a positive real number

$$\boldsymbol{x}(t) = \exp(-\boldsymbol{A}t)\boldsymbol{x}_0 + \int_0^t \exp(-\boldsymbol{A}(t-\tau))d\tau\boldsymbol{b} \tag{19}$$

we have

$$\lim_{t\to\infty} \boldsymbol{x}(t) = \int_0^\infty \exp(-\boldsymbol{A}(t-\tau))d\tau \cdot \boldsymbol{b} = \boldsymbol{A}^{-1}\boldsymbol{b} \tag{20}$$

The following two lemmas are used to prove Theorem 1.

Lemma 1. Let $\boldsymbol{A} \in R^{n\times n}$ be a nonsymmetric positive definite matrix, t be a positive real number

$$\lim_{t\to 0} \exp(\boldsymbol{A}t) = \boldsymbol{I} \tag{21}$$

Proof: According to the definition of matrix exponential, $\exp(\boldsymbol{A}t)$ can be written as:

$$\exp(\boldsymbol{A}t) = \boldsymbol{I} + \boldsymbol{A}t + \frac{(\boldsymbol{A}t)^2}{2} \tag{22}$$

it is easy to find $\lim_{t\to 0} \exp(\boldsymbol{A}t) = \boldsymbol{I}$.

Lemma 2. Let $\boldsymbol{A} \in R^{n\times n}$ be a nonsymmetric positive definite matrix, t be a positive real number

$$\lim_{t\to\infty} \exp(\boldsymbol{A}t) = \boldsymbol{O} \tag{23}$$

where $\boldsymbol{O}$ is zero matrix.

Proof: Consider the following linear dynamic systems of equations

$$\begin{cases} \dfrac{d\boldsymbol{x}(t)}{dt} = -\boldsymbol{Ax}(t) \quad t \ge 0 \\ \boldsymbol{x}(0) = \boldsymbol{x}_0 \end{cases} \tag{24}$$

where $\boldsymbol{x}_0 \in R^n$ is an arbitrarily preassigned vector, and $\boldsymbol{A} \in \boldsymbol{R}^{n\times n}$ is a positive definite matrix. The exact solution of a dynamic system (24) can be expressed as:

$$\boldsymbol{x}(t) = \exp(-\boldsymbol{A}t)\boldsymbol{x}_0 \tag{25}$$

The proof investigates two aspects regarding the existence of both repeated and non-repeated eigenvalues within the matrix.

(1) Assume $\boldsymbol{A}$ has n different eigenvalues $\lambda_1 \cdots \lambda_n$ and $\boldsymbol{v}_1 \cdots \boldsymbol{v}_n$ is the eigenvector corresponding to the eigenvalue.
The solution (Boyce and DiPrima, 1986) of linear dynamic system (24) can also be written as follows

$$\boldsymbol{x}(t) = c_1 \exp(\lambda_1 t)\boldsymbol{v}_1 + \cdots + c_n \exp(\lambda_n t)\boldsymbol{v}_n \tag{26}$$

where c_n is a constant, determined by the initial value $\boldsymbol{x}_0$. Because $\boldsymbol{A}$ is a positive definite matrix, the eigenvalues of $-\boldsymbol{A}$ are less than *0*. When t approaches infinity, we can have $\lim_{t\to\infty} \boldsymbol{x}(t) \to 0$.

(2) Assume the $\boldsymbol{A}$ has m different eigenvalues $\lambda_1 \cdots \lambda_m$, $\boldsymbol{v}_1 \cdots \boldsymbol{v}_m$ are the eigenvector corresponding to the eigenvalues, where $m < n$.
It means that there are repeated eigenvalues. In this case, the number of corresponding linearly independent eigenvectors is smaller than the algebraic multiplicity of the eigenvalue. Assume the maximum algebraic multiplicity to be $k \in N^*, k < n$. The solution (Boyce and DiPrima, 1986) of a linear dynamic system (24) can be written as follows

$$\boldsymbol{x}(t) = \exp(\lambda_1 t)\boldsymbol{C}_1\boldsymbol{v}_1 + \cdots + \exp(\lambda_m t)\boldsymbol{C}_m\boldsymbol{v}_m \tag{27}$$

Where $\boldsymbol{C}_m$ is a polynomial matrix, its degree is no more than $k-1$. Because $\boldsymbol{A}$ is a positive definite matrix, the eigenvalues λ of $-\boldsymbol{A}$ are less than 0. When t approaches infinity, it is easy to get $\lim_{t\to\infty} \exp(\lambda t)t^{k-1} \to 0$, so $\lim_{t\to\infty} \boldsymbol{x}(t) \to \boldsymbol{0}$.

Based on the discussion above, we can get

$$\lim_{t\to\infty} \boldsymbol{x}(t) = \exp(-\boldsymbol{A}t)\boldsymbol{x}_0 \to \boldsymbol{0} \tag{28}$$

Since $\boldsymbol{x}_0 \in R^n$ is an arbitrarily preassigned vector, we can get $\lim_{t\to\infty} \exp(-\boldsymbol{A}t) \to \boldsymbol{O}$.

According to Lemma 1, we can get

$$\begin{aligned} \lim_{t\to\infty} \boldsymbol{x}(t) &= \lim_{t\to\infty} \exp(-\boldsymbol{A}t)\boldsymbol{x}_0 + \lim_{t\to\infty} \int_0^t \exp(-\boldsymbol{A}(t-\tau))d\tau\boldsymbol{b} \\ &= \lim_{t\to\infty} \int_0^t \exp(-\boldsymbol{A}(t-\tau))d\tau\boldsymbol{b} \end{aligned} \tag{29}$$

Let $s = t - \tau$, then

$$\begin{aligned} \lim_{t\to\infty} \int_0^t \exp(-\boldsymbol{A}(t-\tau))d\tau &= \lim_{t\to\infty} -\int_t^0 \exp(-\boldsymbol{A}(s))ds \\ &= \lim_{t\to\infty} \int_0^t \exp(-\boldsymbol{A}(s))ds \end{aligned} \tag{30}$$

According to Lemma 2, we can get

$$\begin{aligned} \lim_{t\to\infty} \boldsymbol{x}(t) &= \lim_{t\to\infty} \int_0^t \exp(-\boldsymbol{A}(s))ds \\ &= \boldsymbol{A}^{-1}\left(\boldsymbol{I} - \lim_{t\to\infty} \exp(-\boldsymbol{A}t)\right)\boldsymbol{b} \\ &= \boldsymbol{A}^{-1}\boldsymbol{b} \end{aligned} \tag{31}$$

The PIIE method is appropriate for asymmetric positive definite linear systems, as demonstrated by the aforementioned proof.

3. The IPIIE method

The termination iteration condition proposed by Fu et al. (2018) facilitates the selection of the optimal number of termination iterations for the PIIE method. The optimal iteration termination parameter can be regarded as an alternative form of the PIIE regularization parameter. This section initially demonstrates, through a straightforward example, that for a given coefficient matrix, the observed values have a significant impact on the optimal termination iteration parameter. To enhance the computational accuracy of the PIIE method and to enable adaptive selection of parameters for iteration termination, this section introduces an improved algorithm that integrates the PIIE with the Iterative Refinement Method. This innovative combination of algorithms significantly enhances both the accuracy and robustness of the PIIE method.

3.1 The optimal termination iteration parameters

Considering the typical ill-conditioned linear nonsymmetric positive definite system with $m \times m$ Hilbert matrices $\boldsymbol{H}_m = (h_{ij})_{m\times m}$, where

$h_{ij} = \dfrac{1}{i+j+1}, i, j = 1,2,\cdots,m$

$$\boldsymbol{H}_m \boldsymbol{x} = \boldsymbol{b} \tag{32}$$

In this experiment, m is 300, the condition number of $\boldsymbol{H}_m$ is 3.93e+19. Assume that the two exact solutions of the Eq. (32) are

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{x}_1 = \sin(0.02\cdot \boldsymbol{y}) \\ \boldsymbol{x} &= \boldsymbol{x}_2 = \boldsymbol{y}^3 \end{aligned} \tag{33}$$

with $\boldsymbol{y} = [1,2,\cdots,200]^T$. And the two observations are $\boldsymbol{b}_1 = \boldsymbol{H}_m\boldsymbol{x}_1$ and $\boldsymbol{b}_2 = \boldsymbol{H}_m\boldsymbol{x}_2$. Fig. 1 illustrates that when the iteration termination parameter n is set to 2, the PIIE method yields satisfactory results with, $\boldsymbol{x} = \boldsymbol{x}_2$. However, when the termination parameter $n = 2$ is unchanged, a noticeable deviation arises between the PIIE method's results and the exact solution with $\boldsymbol{x} = \boldsymbol{x}_1$. Fig. 2 further demonstrates that as the parameter n increases, the solution from the PIIE method gradually converges to the exact solution, achieving satisfactory results when $n = 5$.

This numerical experiment demonstrates that, for a given coefficient matrix, the observed value has a substantial impact on the optimal termination iteration parameter, denoted as n.

3.2 The theory IPIIE method

Iterative refinement is a significant numerical technique used to improve the accuracy of initial solutions in various computational problems, especially in the context of solving linear systems and conducting numerical computations (Cui et al., 2023; Pan et al., 2021). The core principle of iterative refinement involves starting with an approximate solution and iteratively adjusting it based on the residuals or errors observed from the initial solution.

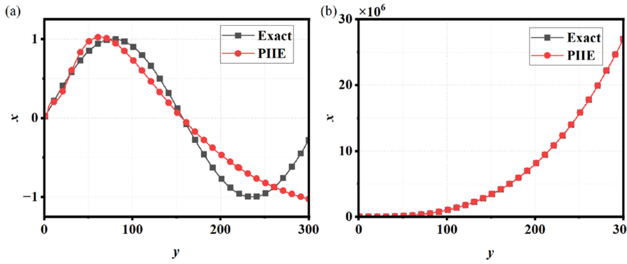

Fig. 1. The numerical results of the PIIE method. The termination iteration parameter n is 2. (a) $\boldsymbol{x} = \boldsymbol{x}_1$ (b) $\boldsymbol{x} = \boldsymbol{x}_2$

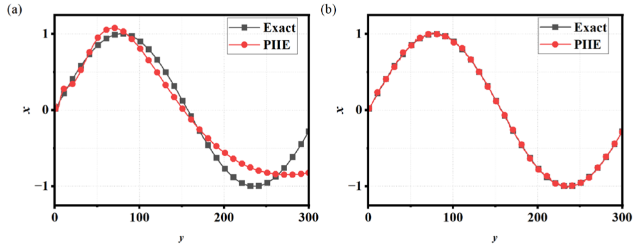

Fig. 2. The numerical results of the PIIE method with $\boldsymbol{x} = \boldsymbol{x}_1$. The parameter of the termination iteration parameter n takes different values (a) n = 4 (b) n = 5.

Let $\boldsymbol{A}$ and $\boldsymbol{x}$ be as in Eq. (18). Then, the iterative refinement algorithm defines the solution to Eq. (18) iteratively by,

$$\boldsymbol{A}(\boldsymbol{x}^k - \boldsymbol{x}^{k-1}) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{k-1} \tag{34}$$

with $\boldsymbol{x}^0 = \boldsymbol{0}$ as initial model. The solution of Eq. (34) is $\boldsymbol{c}^{k-1}$, then

$$\boldsymbol{x}^k = \boldsymbol{x}^{k-1} + \boldsymbol{c}^{k-1} \tag{35}$$

This iterative process continues until a desired level of accuracy or convergence is achieved, thereby improving the overall precision of the solution obtained.

Inspired by the Iterative Refinement method, the integration of this method with the PIIE algorithm can enhance the accuracy of numerical solutions. The steps for the calculation are outlined as follows:

(1) The Algorithm 1 is used to calculate the initial value, and the iteration termination parameter n is 2.

(2) Calculation of the residual vector $\boldsymbol{r}^k$:

$$\boldsymbol{r}^k = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^k \tag{36}$$

(3) If the residual satisfies the threshold, stop the calculation. Otherwise, use the PIIE approach to solve the Eq. (37):

$$\boldsymbol{A}\boldsymbol{c}^k = \boldsymbol{r}^k \tag{37}$$

(4) Correcting the calculation results:

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \boldsymbol{c}^k \tag{38}$$

Repeat steps 2 to 4 until the residual is sufficiently small or a predetermined number of iterations is reached.

Algorithm 2: The IPIIE algorithm

```
1: Compute the preconditioned matrix $\boldsymbol{Q}$, $\boldsymbol{A} = \boldsymbol{QA}, \boldsymbol{b} = \boldsymbol{Qb}$.
2: $\boldsymbol{x}^0 = PIIE(\boldsymbol{A}, \boldsymbol{b})$
3: Compute $\|\boldsymbol{r}^k\| = \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^k\|$
4: while until $\|\boldsymbol{r}\|$ satisfy termination conditions.
5:     $\boldsymbol{c}^k = PIIE(\boldsymbol{A}, \boldsymbol{r}^k)$
6:     $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \boldsymbol{c}^k$
7: end
```

3.3 Convergence analysis

The Algorithm 2 is a combination algorithm. As long as the PIIE convergence is guaranteed, IPIIE can be guaranteed to be convergent. Huang et al. (2021) studied the above problems in detail.

The PIIE method uses row normalization to preprocess the matrix. According to the Gershgorin Circle Theorem (Bell, 1965), the $\rho(\boldsymbol{QA}^T\boldsymbol{A})$ in Eq. (14) is less than or equal to 1.

If $p \geq 1$, $\tau > 0$, $\rho(\boldsymbol{B}_p(\tau)) < 1$ with $\boldsymbol{B}_p(\tau)$ defined in Eqs. (10) and $\rho(\cdot)$ denote the spectral radius of a matrix. For the PIIE method

$$\boldsymbol{I} + \boldsymbol{T}_p^{(k)} = (\boldsymbol{I} + \boldsymbol{T}_p^{(k-1)})^2 = (\boldsymbol{I} + \boldsymbol{T}_0)^{2^k} = (\boldsymbol{B}_p(\tau))^{2^k} \tag{39}$$

$$\begin{aligned} \boldsymbol{x}^k &= (\boldsymbol{I} + \boldsymbol{B}_p(\tau)^{2^{k-1}})\boldsymbol{x}^{k-1} \\ &= \left(\prod_{s=0}^{k-1}\left(\boldsymbol{I} + \boldsymbol{B}_p(\tau)^{k-1}\right)^{2^s}\right)\boldsymbol{x}^0 \\ &= \left[\boldsymbol{I} + \boldsymbol{B}_p(\tau) + (\boldsymbol{B}_p(\tau))^2 + \cdots + (\boldsymbol{B}_p(\tau))^{2^j-1}\right]\boldsymbol{F}_p(\tau)\boldsymbol{QAb} \\ &= (\boldsymbol{I} - \boldsymbol{B}_p(\tau))^{2^k}((\boldsymbol{QA}^T\boldsymbol{A})^{-1})\boldsymbol{QA}^T\boldsymbol{b} \end{aligned} \tag{40}$$

When k tends to infinity, PIIE will converge with $((\boldsymbol{QA})^{-1})\boldsymbol{Qb}$.

3.4 The validation of the IPIIE algorithm

This section still uses the Hilbert matrix model in subsection 3.1 to test the robustness of the IPIIE algorithm. Assume that the exact solution of the Eq.(32) are

$$\boldsymbol{x} = \boldsymbol{x}_1 = \sin(0.02 \cdot \boldsymbol{y}) \tag{41}$$

with $\boldsymbol{y} = [1,2,\cdots,300]^T$.The value of $\boldsymbol{x}_1$ is the same as that of the in subsection 3.1. The relative error (RE) is

$$RE = \frac{\left\|\boldsymbol{x}_{reg} - \boldsymbol{x}_{true}\right\|}{\left\|\boldsymbol{x}_{true}\right\|} \tag{42}$$

where $\boldsymbol{x}_{reg}$ is numerical solution and $\boldsymbol{x}_{true}$ is a true solution.

According to subsection 3.1 the optimal termination iteration parameter n is 5. The relative error of PIIE algorithm is 1.26*e*-2, and the relative error of IPIIE is 2.06*e*-3. According to Fig. 3 and the relative error of the two algorithms, it can be concluded that the IPIIE algorithm is better than that of the optimal solution of the PIIE algorithm.

Numerical tests indicate that the IPIIE method consistently outperforms the PIIE method. IPIIE delivers accurate results without requiring adjustments to the termination iteration parameter *n*, effectively reducing the sensitivity of termination iteration parameters to observations compared to PIIE.

4. Numerical experiments

All numerical experiments in this subsection were carried out using MATLAB R2024b on a laptop computer with an Intel Core i7-12700H CPU and 64 GB of RAM.

4.1 Ill-conditioned problems in one space-dimension

Considering the test problems (Hansen, 2007) *heat* and *gravity* , are linear discrete ill-posed problems. These examples originate from the discretization of Fredholm integral equations of the first kind, which have the general form:

$$\int_a^b k(s,t)x(t)dt = b(s) \tag{43}$$

where $b(s)$ is observable, $k(s,t)$ is a known kernel function. The Eq.(43) is transformed into Eq. (44) by discretizing $x(t)$ into N points,

$$\boldsymbol{K}\boldsymbol{x} = \boldsymbol{b} \tag{44}$$

where $\boldsymbol{K}$ is $N \times N$ matrix, usually an ill-conditioned matrix.

Examples 1

In this example, $k(s,t)$

$$k(s,t) = \frac{(s-t)^{\frac{-3}{2}}}{\left(2\kappa\sqrt{\pi}\exp\left(-\frac{1}{4\kappa^2(s-t)}\right)\right)} \tag{45}$$

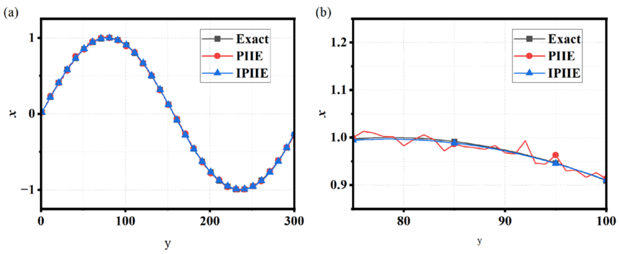

Fig. 3. The numerical results of the PIIE and IPIIE methods with $\boldsymbol{x} = \boldsymbol{x}_1$. The parameter of the termination iteration parameter *n* of PIIE is 5. (a) Original drawing and (b) Drawings of partial enlargement.

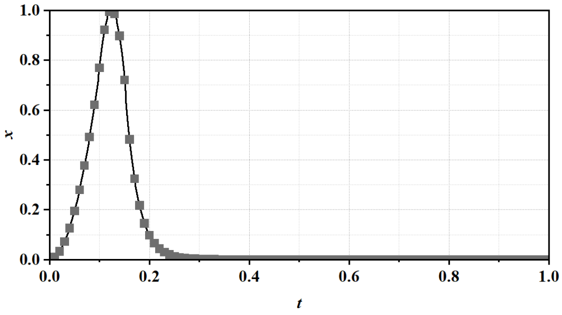

Fig. 4. The desired solution.

where $t \in [0,1]$. The three desired solutions $\boldsymbol{x} \in R^N$ are shown in Fig. 4. In this example, *N* is 1000, and the condition number of $\boldsymbol{K}$ is 2.33*e*+232.

When $\boldsymbol{x} = \boldsymbol{x}_1$, the optimal regularization parameter for the TR method is 1*e*-10, the optimal number of truncated singular values for the TSVD method is 813, and the optimal termination iteration parameter for the PIIE method is 4. The relative errors for the numerical results of PIIE, IPIIE, TR, and TSVD are 3.10*e*-03,2.95*e*-03, 3.62*e*-03, and 7.10*e*-02, respectively. Fig. 5 demonstrates close agreement between the results of PIIE, IPIIE, and TR, while the TSVD method exhibited substantially larger errors in detail.

Examples 2

In geological exploration, the location, shape, and some parameters of geological anomalies occurring in the interior of the earth are usually determined according to the data measured on the surface of the earth. Assuming that the vertical component of the gravity field $g(s)$ is measured on the surface, the mass distribution is located at depth $x(t)$. This inverse problem can be expressed as the Fredholm integral equation of the first kind, and its kernel function is

$$k(s,t) = h\left(h^2 + (s-t)^2\right)^{-3/2} \tag{46}$$

where $t \in [0,1]$, $s \in [0,1]$ and h is 0.5. The two desired solutions $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are shown in Fig. 6

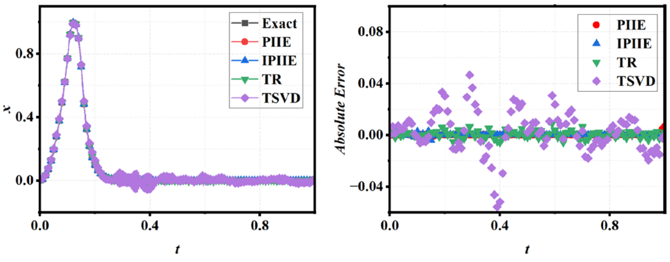

Fig. 5. The numerical results and absolute errors of the PIIE, IPIIE, TR and TSVD methods.

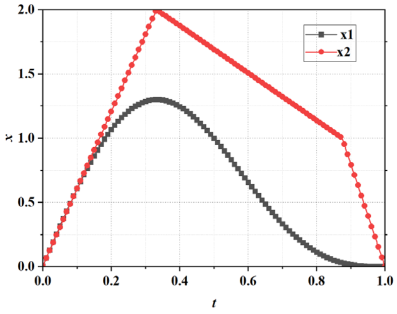

Fig. 6. The two desired solutions $\boldsymbol{x}$.

When $\boldsymbol{x} = \boldsymbol{x}_1$, the optimal regularization parameter for the TR method is 1e-11, the optimal number of truncated singular values for the TSVD method are 22, and the optimal termination iteration parameter for the PIIE method is 2. Fig. 7 illustrates that all four methods yield satisfactory computational results. However, the relative errors for the numerical results of PIIE, IPIIE, TR, and TSVD are 1.23e-06,1.22e-06, 3.24e-04 and 1.62e-06, respectively. From these results, it is evident that the computational accuracy of PIIE, IPIIE, and TSVD is comparable, while the TR method exhibits the lowest accuracy.

When $\boldsymbol{x} = \boldsymbol{x}_2$, the optimal regularization parameter for the TR method is determined to be 1e-12, while the optimal number of truncated singular values for the TSVD method is identified as 24. In addition, the optimal termination iteration parameter of the PIIE method is 3. Fig. 8 shows that there is a certain deviation between the numerical results of the four methods and the exact solution. However, the relative errors associated with the numerical results of PIIE, IPIIE, TR, and TSVD are 1.93e-02, 4.93e-03,1.13e-02, and 6.21e-03, respectively. These findings indicate that the computational accuracy of IPIIE and TSVD is comparable, whereas TR and PIIE exhibit the lowest levels of accuracy.

4.2 Ill-conditioned problems in two space-dimensions

The test problem *blurs* that it is a linear discrete ill-posed problem. The blurring of the image is caused by a blurring function with parameters $\sigma = 3$ and $band = 6$. This blurring function can be determined using the regularization tool package (Hansen, 2007). In this example, the image size is 50×50, as shown in Fig. 9. The PIIE, TR and TSVD methods are compared. The optimal regularization parameter for the TR method has been determined to be 1e-12, while the optimal number of truncated singular values for the TSVD method is identified as 1441. Furthermore, the optimal termination iteration parameter for the PIIE method is established as 3. The numerical results are shown in Fig. 10. The relative errors associated with the numerical results of the PIIE, IPIIE, TR, and TSVD methods are 3.98e-01, 5.07e-02, 5.11e-02, and 1.55e-01, respectively. These findings suggest that the computational accuracy of the IPIIE and TR methods is comparable, whereas the TSVD and PIIE methods demonstrate the lowest levels of accuracy.

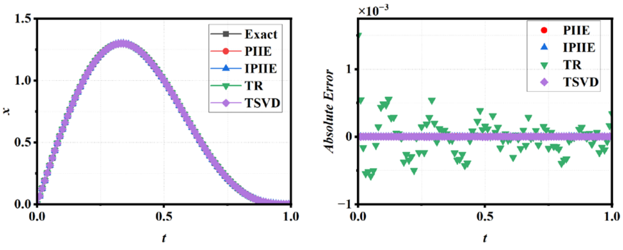

Fig. 7. When $\boldsymbol{x} = \boldsymbol{x}_1$, the numerical results and absolute errors of the PIIE, IPIIE, TR and TSVD methods.

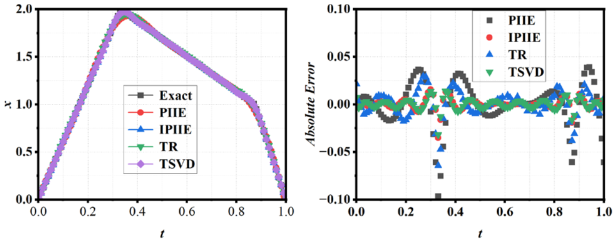

Fig. 8. When $\boldsymbol{x} = \boldsymbol{x}_2$, the numerical results and absolute errors of the PIIE, IPIIE, TR and TSVD methods.

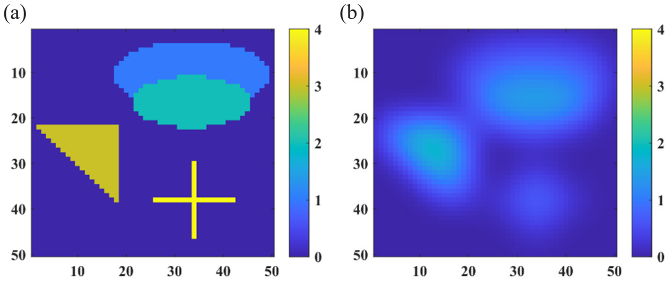

Fig. 9. (a) Original test image, (b) The deblurred image.

In summary, compared with the PIIE, TR, and TSVD methods, the IPIIE method does not require the selection of optimal normalization parameters. Furthermore, the IPIIE method demonstrates enhanced applicability in addressing complex problems.

5. Discussion

The PIIE and IPIIE algorithms use precise integration to calculate the matrix exponential $\exp(-\boldsymbol{A}2^k\tau)$ in Eq. (9). $\boldsymbol{T}_p^{(k+1)} = 2\boldsymbol{T}_p^{(k)} + \boldsymbol{T}_p^{(k)}\boldsymbol{T}_p^{(k)}$ in the Eq. (13) results in high computational complexity. If $\boldsymbol{T}_p^{(k)}$ is the sparse matrix, it can quickly densify the matrix during iteration, significantly increasing computational complexity and memory requirements. Consequently, when addressing large-scale, sparse, and ill-posed problems, the memory requirements for the PIIE and IPIIE methods increase significantly with a greater number of iterations. Fortunately, in the field of numerical solutions for differential equations, researchers have addressed similar limitations by proposing an exponential matrix calculation method that utilizes Krylov subspace techniques (Bergamaschi and Vianello, 2000; Botchev and Knizhnerman, 2020; Druskin and Simoncini, 2011; Fung and Chen, 2006; Suman and Kumar, 2022).

The Krylov subspace method transforms the original matrix exponential-vector product, as presented in the Eq. (9), into an m-dimensional matrix-vector product. The construction of the Krylov subspace basis through the Arnoldi process necessitates only matrix-vector product calculations. Furthermore, the dimension of the Krylov subspace is generally much smaller than the scale of the original problem. Consequently, this method can substantially decrease both computational complexity and memory requirements. For clarity, the PIIE and IPIIE based on the Krylov subspace are referred to as PIIE-K and IPIIE-K, respectively.

Considering the typical simple positive definite system with tridiagonal matrix $\boldsymbol{A}_m = (a_{ij})_{m\times m}$.

$$\boldsymbol{A}_m = \begin{bmatrix} 2 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 2 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 2 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 2 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 2 \end{bmatrix} \tag{47}$$

$$\boldsymbol{A}_m\boldsymbol{x} = \boldsymbol{b} \tag{48}$$

In this case, m is 5000. Assume that the exact solution of the Eq. (32) are

$$\boldsymbol{x} = \sin(0.02 \cdot \boldsymbol{y}) \tag{49}$$

with $\boldsymbol{y} = [1,2,\cdots,5000]^T$.

It can be seen from Fig. 11 that the number of non-zero elements in matrices $\boldsymbol{T}_p^{(0)}$ and $\boldsymbol{T}_p^{(10)}$ is 44,980 and 657,444, respectively. The memory consumption of matrices $\boldsymbol{T}_p^{(0)}$ and $\boldsymbol{T}_p^{(10)}$ in MATLAB is 1.258 MB and 178.100 MB, respectively. After 10 iterations, the number of non-zero elements in matrix $\boldsymbol{T}_p^{(10)}$ increases by approximately 146 times, while the memory usage increases by approximately 141 times.

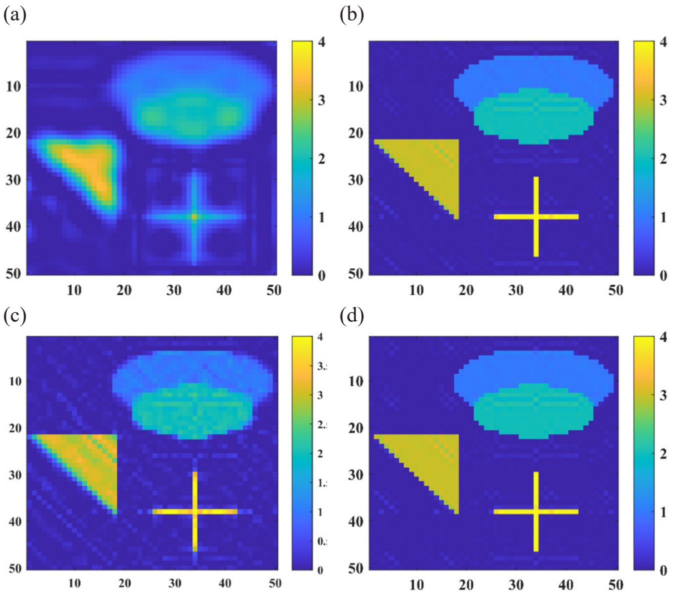

Fig. 10. The calculation results of different methods. (a) PIIE, (b) TR, (c) TSVD and (d) IPIIE.

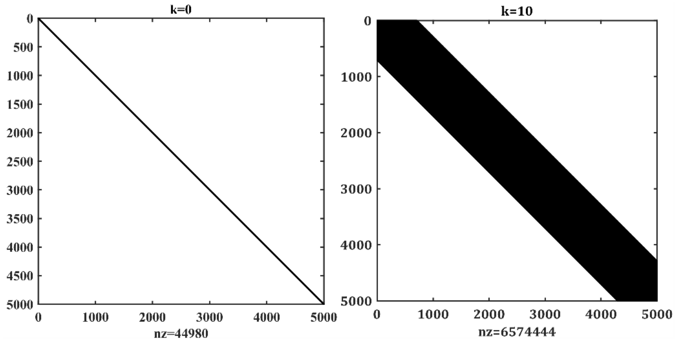

Fig. 11. A visual illustration of the number of non-zero elements in $\boldsymbol{T}_p^{(k)}$, the superscript k denotes the number of iterations. $\boldsymbol{nz}$ is the number of non-zero elements, k is iteration times.

Table 1.
The time of solving the Eq. (48) by PIIE, PIIE-K, IPIIE and IPIIE-K with different order $\boldsymbol{A}_m$ is taken. The m is the order of matrix $\boldsymbol{A}$ and the dimension of Krylov-subspace is 30, n is terminating iteration parameters.

Method	*m*=2000		*m*=5000	
	RE	Time	RE	Time
PIIE(*n*=2)	2.000e-01	7.223s	2.000e-01	14.041s
PIIE-K(*n*=2)	1.126e-02	0.057s	1.126e-02	0.304s
IPIIE	4.298e-07	9.184s	4.012e-07	20.520s
IPIIE-K	2.319e-07	0.303s	1.728e-07	1.683s

As illustrated in Table 1, when the termination iteration parameter n is set to 2, the computational time for the IPIIE method is significantly greater than that of the IPIIR-K method. Specifically, for m = 2000, the computation time for IPIIE is approximately 30 times longer than that of IPIIR-K, whereas for m = 5000, the computation time for IPIIE is about 12 times that of IPIIR-K. In terms of computational accuracy, at m = 2000, the relative errors for the PIIE and PIIE-K methods are 0.200 and 0.01126, respectively, while the relative errors for the IPIIE and IPIIE-K methods are 4.298e-07 and 2.319e-07, respectively. For m = 5000, the relative errors for the PIIE and PIIE-K methods remain at 0.200 and 0.01126, while the relative errors for the IPIIE and IPIIE-K methods are 4.012e-07 and 1.728e-07, respectively.

6. Conclusions

This paper presents the IPIIE algorithms, which combine the iterative refinement method with the PIIE algorithms. Based on numerical experiments, the following conclusions have been drawn:

1. The PIIE is not only applicable to symmetric positive definite linear systems but also to asymmetric positive definite linear systems.
2. The IPIIE algorithm effectively alleviates the influence of observations on the optimal termination iteration parameters in the PIIE algorithm. An advantage of IPIIE is its avoidance of the need for selecting regularization parameters compared with TR and TSVD method.
3. The IPIIE-K method effectively reduces the high complexity and large memory consumption inherent in the PIIE method when solving large-scale, sparse problems, demonstrating significant advantages.

Additionally, the second-order dynamic system method for solving linear equations is ultimately reduced to solving a first-order dynamic system. Therefore, the application of the IPIIE algorithm within the second-order dynamic system method is further examined and discussed.

CRediT authorship contribution statement

Jingtao Sun: Conceptualization, Methodology, Investigation, Writing – original draft, Software; **Jiulong Cheng:** Supervision, Writing – review & editing; **Lei Qin:** Resources, Writing – review & editing.

Declaration of competing interest

We declare that this research was conducted in the absence of any commercial or financial relationship that could be construed as a potential conflict of interest.

Declaration of Generative AI and AI-assisted technologies in the writing process

The authors confirm that there was no use of artificial intelligence (AI)-assisted technology for assisting in the writing or editing of the manuscript and no images were manipulated using AI.

Acknowledgement

We would like to express our gratitude to the three anonymous reviewers for their thorough evaluation of our papers, as well as for their constructive feedback and valuable suggestions regarding the manuscripts.

References

Bai, X., Huang, G-Xin., Lei, X-Jun., Reichel, L., Yin, F., 2021. A novel modified TRSVD method for large-scale linear discrete ill-posed problems. Appl. Math. Sci. 164, 72-88. https://doi.org/10.1016/j.apnum.2020.08.019

Beik, F.P.A., Ahmadi-Asl, S., Ameri, A., 2018. On the iterative refinement of the solution of ill-conditioned linear system of equations. Int. J. Comput. Math. 95, 427-443. https://doi.org/10.1080/00207160.2017.1290436

Bell, H.E., 1965. Gershgorin's theorem and the zeros of polynomials. Am. Math. Mon. 72, 292. https://doi.org/10.2307/2313703

Benning, M., Burger, M., 2018. Modern regularization methods for inverse problems. Acta Numerica 27, 1-111. https://doi.org/10.1017/s0962492918000016

Bergamaschi, L., Vianello, M., 2000. Efficient computation of the exponential operator for large, sparse, symmetric matrices. Numer. Linear Algebra Appl. 7, 27-45. https://doi.org/10.1002/(sici)1099-1506(200001/02)7:1<27::aid-nla185>3.0.co;2-4

Botchev, M.A., Knizhnerman, L.A., 2020. ART: Adaptive residual-time restarting for krylov subspace matrix exponential evaluations. J. Comput. Appl. Math. 364, 112311. https://doi.org/10.1016/j.cam.2019.06.027

Boyce, W.E., DiPrima, R.C., 1986. Elementary differential equations and boundary value problems - Fourth edition.

Chang, W., D'Ascenzo, N., Xie, Q., 2024. A relaxed iterated tikhonov regularization for linear ill-posed inverse problems. J. Math. Anal. Appl. 530, 127754. https://doi.org/10.1016/j.jmaa.2023.127754

Chen, Z., Chan, T.H.T., 2017. A truncated generalized singular value decomposition algorithm for moving force identification with ill-posed problems. J. Sound Vib. 401, 297-310. https://doi.org/10.1016/j.jsv.2017.05.004

Chu, M.T., 1988. On the continuous realization of iterative processes. SIAM Rev. 30, 375-387. https://doi.org/10.1137/1030090

Cui, L-Bin., Hu, W-Li., Yuan, J-Yun., 2023. Iterative refinement method by higher-order singular value decomposition for solving multi-linear systems. Appl. Math. Lett. 146, 108819. https://doi.org/10.1016/j.aml.2023.108819

Dmytryshyn, A., Fasi, M., Gulliksson, Mårten., 2022. The dynamical functional particle method for multi-term linear matrix equations. Appl. Math. Comput. 435, 127458. https://doi.org/10.1016/j.amc.2022.127458

Druskin, V., Simoncini, V., 2011. Adaptive rational krylov subspaces for large-scale dynamical systems. Sys. Control Letters 60, 546-560. https://doi.org/10.1016/j.sysconle.2011.04.013

Edvardsson, S., Neuman, M., Edström, P., Olin, H., 2015. Solvings through particle dynamics. Comput. Phys. Commun. 197, 169-181. https://doi.org/10.1016/j.cpc.2015.08.028

Fu, M., Li, Y., 2018. A Preconditioned precise integration method for solving Ill-conditioned linear equations. Appl. Math. Mech. (Engl. Ed.) 39. https://doi.org/10.21656/1000-0887.380206

Fu, M., Li, Y., Wenzhi, Z., 2018. A precise form for solving ill-conditioned algebraic equations and its iteration stopping criterion. Chin. J. Appl. Mech. 35. https://doi.org/10.11776/cjam.35.02.D008

Fu, M., Zhang, W., 2011. Precise integration method for solving il l-conditioned algebraic equations. Chin. J. Comput. Mech. 28. https://doi.org/10.7511/jslx201104007

Fung, T.C., Chen, Z.L., 2006. Krylov precise time-step integration method. Numerical Meth Engineering 68, 1115-1136. https://doi.org/10.1002/nme.1737

Hansen, P.C., 2007. Regularization tools version 4.0 for matlab 7.3. Numer. Algor. 46, 189-194. https://doi.org/10.1007/s11075-007-9136-9

Hoang, N.S., Ramm, A.G., 2010. Dynamical systems gradient method for solving ill-Conditioned linear algebraic systems. Acta Appl. Math. 111, 189-204. https://doi.org/10.1007/s10440-009-9540-3

Huang, C., Wang, L., Fu, M., Lu, Z-Rong., Chen, Y., 2020. A novel iterative integration regularization method for ill-posed inverse problems. Eng. Comput.

Huang, G., Liu, Y., Yin, F., 2022. Tikhonov regularization with MTRSVD method for solving large-scale discrete ill-posed problems. J. Comput. Appl. Math. 405, 113969. https://doi.org/10.1016/j.cam.2021.113969

Li, W., Xue, J., Li, X., 2024. Development of multi resolution transient electromagnetic virtual wave-field migration imaging method. J. Appl. Geophys. 222, 105314. https://doi.org/10.1016/j.jappgeo.2024.105314

Lu, K., Fan, Y'nan., Li, X., Zhou, J., Fan, K., 2023. Transient electromagnetic pseudo wavefield imaging based on the sweep-time preconditioned precise integration algorithm. J. Appl. Geophys. 209, 104937. https://doi.org/10.1016/j.jappgeo.2023.104937

Mittal, G., Giri, A.K., 2021. Iteratively regularized landweber iteration method: Convergence analysis via hölder stability. Appl. Math. Comput. 392, 125744. https://doi.org/10.1016/j.amc.2020.125744

Miyatake, Y., Sogabe, T., Zhang, S-Liang., 2018. On the equivalence between SOR-type methods for linear systems and the discrete gradient methods for gradient systems. J. Comput. Appl. Math. 342, 58-69. https://doi.org/10.1016/j.cam.2018.04.013

Pan, Y., Dai, W., Huang, L., Li, Z., Mikkola, A., 2021. Iterative refinement algorithm for efficient velocities and accelerations solutions in closed-loop multibody dynamics. Mech. Syst. Signal Process. 152, 107463. https://doi.org/10.1016/j.ymssp.2020.107463

Ramm, A.G., 2004. Dynamical systems method for solving operator equations. Commun. Nonlinear Sci. Numer. Simul. 9, 383-402. https://doi.org/10.1016/s1007-5704(03)00006-6

Salkuyeh, D.K., Fahim, A., 2011. A new iterative refinement of the solution of ill-conditioned linear system of equations. Int. J. Comput. Math. 88, 950-956. https://doi.org/10.1080/00207161003713907

Suman, S.K., Kumar, A., 2022. Investigation and implementation of model order reduction technique for large scale dynamical systems. Arch. Computat. Methods Eng. 29, 3087-3108. https://doi.org/10.1007/s11831-021-09690-8

Wang, J., Mao, M., Xiang, R., Wang, H., Lian, H., 2024. A memory-Efficient PITD method for multiscale electromagnetic simulations. IEEE Microw. Wireless Tech. Lett. 34, 967-970. https://doi.org/10.1109/lmwt.2024.3408457

Wang, W., Qu, G., Song, C., Ge, Y., Liu, Y., 2024. Tikhonov regularization with conjugate gradient least squares method for large-scale discrete ill-posed problem in image restoration. Appl. Math. Sci. 204, 147-161. https://doi.org/10.1016/j.apnum.2024.06.010

Wu, X., 2023. An exponential approach to highly ill-conditioned linear systems. Appl. Math. Lett. 140, 108560. https://doi.org/10.1016/j.aml.2022.108560

Wu, X., Shao, R., Zhu, Y., 2002. New iterative improvement of a solution for an ill-conditioned system of linear equations based on a linear dynamic system. Comput. Math. Appl. A 44, 1109-1116. https://doi.org/10.1016/s0898-1221(02)00219-5

Wu, Y., 2012. Parametric inverse of severely ill-conditioned hermitian matrices in signal processing. J. Franklin Institute 349, 1048-1060. https://doi.org/10.1016/j.jfranklin.2011.12.006

Xu, G., Gan, J., Liu, H., Wu, W., 2024. Shape function method based on truncated singular value decomposition regularization for hull longitudinal bending moment identification. Measurement 232, 114703. https://doi.org/10.1016/j.measurement.2024.114703

Xu, P., 1998. Truncated SVD methods for discrete linear ill-posed problems. Geophys. J. Int. 135, 505-514. https://doi.org/10.1046/j.1365-246x.1998.00652.x

Yang, L., Zhang, J., He, R., Lin, W., 2021. A dual interpolation precise integration boundary face method to solve two-dimensional transient heat conduction problems. Eng. Anal. Bound. Elem. 122, 75-84. https://doi.org/10.1016/j.enganabound.2020.09.014

Yang, X.-M., Deng, Z.-L., 2017. A data assimilation process for linear ill-posed problems. Math Methods App. Sci. 40, 5831-5840. https://doi.org/10.1002/mma.4432

Yu, Y., Yang, L., Shen, Y., Wang, W., Li, B., Chen, Q., 2024. An iterative and shrinking generalized ridge regression for ill-conditioned geodetic observation equations. J. Geod. 98. https://doi.org/10.1007/s00190-023-01795-1

Wan-Xie, Z., 2004. On precise integration method. J. Computational Applied Math 163, 59-78. https://doi.org/10.1016/j.cam.2003.08.053

Zhu, L., Ye, K., Huang, D., Wu, F., Zhong, W., 2023. An adaptively filtered precise integration method considering perturbation for stochastic dynamics problems. Acta Mech. Solida Sin. 36, 317-326. https://doi.org/10.1007/s10338-023-00381-4

Zhu, X., Ma, X., Shao, J., 2021. Low-Memory implementation of PITD method using a thresholding scheme. IEEE Microw. Wireless Compon. Lett. 31, 537-540. https://doi.org/10.1109/lmwc.2021.3069643