Filomat xx (2025), zzz–zzz
DOI

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Exploring the potential of inertial type algorithms in solving coincidence point/value problems: Theoretical analysis and practical applications

Ayşegül Keten Çopur[a], Emirhan Hacıoğlu[b], Faik Gürsoy[c], Gradimir V. Milovanović[d,e]

[a]*Department of Mathematics and Computer Science, Necmettin Erbakan University, Konya, 42090, Türkiye*
[b]*Department of Mathematics, Trakya University, Edirne, 22030, Türkiye*
[c]*Department of Mathematics, Adıyaman University, Adıyaman, 02040, Türkiye*
[d]*Serbian Academy of Sciences and Arts, 11000 Belgrade, Serbia*
[e]*University of Niš, Faculty of Sciences and Mathematics, P.O. Box 224, 18000 Niš, Serbia*

Abstract. In this study, we present novel inertial type algorithms designed to address a research gap and provide a solution to an open problem. We focus on a high-performing "Jungck normal-S inertial type algorithm", which we thoroughly examine for its convergence, stability and data dependency properties. To substantiate our theoretical findings, we include thoughtfully selected numerical examples covering both differential and integral equations. Furthermore, we show case the versatility of these algorithms by employing them to create aesthetically pleasing polynomiographs, enhancing both the educational and artistic dimensions of our work.

1. Introduction, basic concepts and some algorithms

A plethora of systematic methods, which can be broadly classified into iteration and direct methods, exists in the literature to solve a wide range of problems encountered in research fields. While direct methods are limited in dealing with inherently nonlinear problems, iteration algorithms become indispensable for approximating solutions to these problems. In recent years many algorithms have been introduced, aiming for higher convergence rates and possessing various qualitative properties such as data dependency, stability and convergence (see, [1, 11–13, 18, 19, 24]). As these algorithms find applications in diverse fields, the demand for purpose-specific designs grows. When addressing challenges in research, existing algorithms are refined or novel ones are designed to cater to unique problems. The chosen algorithm should adhere to specific criteria, notably rapid and accurate computations, which hold significant importance in applied and computational research domains. In this context, accelerating iteration algorithms can offer extensive benefits across various domains, particularly in optimization, machine learning, numerical analysis and scientific computing. They enhance efficient handling of increasingly complex problems, optimize computation time and resources, and facilitate real-time and large-scale applications. However, it is essential to

2010 *Mathematics Subject Classification*. Primary 47H09; Secondary 47H10, 47J25
Keywords. Convergence; Coincidence point/value; Data dependency; Inertial type algorithm; Stability; Polynomigrahy.

Communicated by (name of the Editor, mandatory)
Research of G.V.M. was partly supported by the Serbian Academy of Sciences and Arts (Project Φ-96).

Email addresses: aketen@erbakan.edu.tr (Ayşegül Keten Çopur), emirhanhacioglu@trakya.edu.tr (Emirhan Hacıoğlu), fgursoy@adiyaman.edu.tr (Faik Gürsoy), gvm@mi.sanu.ac.rs (Gradimir V. Milovanović)

strike a balance between speed and accuracy, as overly accelerated algorithms may compromise precision or convergence stability (see, e.g., [19]). To achieve efficient solutions for increasingly complex problems, researchers have devised diverse acceleration techniques, such as relaxation, inertial type extrapolation, and variants of accelerated gradient descent like Nesterov accelerated gradient. These techniques effectively accelerate iteration algorithms (for a more compherensive understanding of acceleration techniques, refer to [6]). However, it is essential to consider that not all methods may yield desired improvements in every scenario, as their effectiveness depends on the nature of the problem and the characteristics of the fixed point iteration algorithm being used. Hence, as always, experimentation and careful analysis on their impact on convergence behavior are key to identfying the most effective approach to a specific problem and achieve faster and more robust convergence. The objective of this paper is twofold: to fill a gap in the literature by designing inertial type algorithms for the pairs of mappings in the context of the coincidence point/value theory and thus addressing an problem in the literature in a positive manner. To achieve this goal, we shall first provide an overview of inertial iteration algorithms and their formulations within the context of coincidence point/value theory; second, we shall conduct a thorough analysis of their convergence, stability and data dependency properties; finally, we shall demonstrate their effectiveness through empirical evaluations on a range of challenging numerical problems.

Polyak [29] initially introduced an inertial extrapolation, relying on the principles of two-order time dynamical system's heavy-ball method, as an acceleration technique to address the challenge of smooth minimization. His approach generates a sequence $\{x_n\}_{n=0}^{\infty}$ by the following algorithm

```
**Polyak's heavy-ball method**
**Input:** An objective function $f$, initial points $x_{-1}, x_0$ and budget $N$.
**for** $n = 0, 1, 2, \dots, N$ **do**
    $x_{n+1} = x_n - \alpha_n \nabla f(x_n) + \beta_n (x_n - x_{n-1})$
**end for**
**Output:** Approximate solution $x_N$
```

in which ∇f denotes the gradient of the objective function f, x_n represents the current iterate at iteration n, $\beta_n (x_n - x_{n-1})$ is the momentum term (also called inertial term), which enables the algorithm to maintain a memory of previous iterations and incorporate their influence into the current update, facilitating faster convergence and better solution quality, α_n is the step size (learning rate) parameter at iteration n, which determines the size of the gradient descent update, and β_n is the momentum parameter at iteration n, controlling the influence of the previous update $x_n - x_{n-1}$ on the current update.

In recent times, a considerable number of scholars have devised fast iteration algorithms through the utilization of inertial extrapolation. These methods encompass the inertial proximal method [2, 27], the inertial forward-backward algorithm [21], the inertial proximal ADMM [5], the fast iteration shrinkage thresholding algorithm (FISTA) [3] and inertial KM-type algorithm [23].

Consider the following coincidence point/value problem:

$$\text{Find } (x, p) \in \mathbb{X}' \times \mathbb{X} \text{ such that } \mathbb{T}_1 x = \mathbb{T}_2 x = p, \tag{1}$$

in which $\mathbb{X}' \neq \emptyset$ is an arbitrary set, $\mathbb{X}$ is a Banach space, and $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ are two mappings. A point $x \in \mathbb{X}'$ is called (i) a fixed point (provided that $\mathbb{X}' = \mathbb{X}$) of $\mathbb{T}_1$ (or $\mathbb{T}_2$) if $\mathbb{T}_1 x = x$ (or $\mathbb{T}_2 x = x$); (ii) a common fixed point (provided that $\mathbb{X}' = \mathbb{X}$) of the pair $(\mathbb{T}_1, \mathbb{T}_2)$ if $\mathbb{T}_1 x = \mathbb{T}_2 x = x$; (iii) a coincidence point of the pair $(\mathbb{T}_1, \mathbb{T}_2)$ if $\mathbb{T}_1 x = \mathbb{T}_2 x$. If $\mathbb{T}_1 x = \mathbb{T}_2 x = p$ for some x in $\mathbb{X}'$, then p is called a coincidence value of $(\mathbb{T}_1, \mathbb{T}_2)$. It is obvious that finding the solutions of problems that can be modelled with equations of type (1), also called coincidence point equations, is equivalent to finding the coincidence points of the corresponding pair of mappings $(\mathbb{T}_1, \mathbb{T}_2)$.

The following result demonstrates an equivalent relationship between a coincidence point problem of the form (1) and a fixed point problem of the form

$$\text{Find } r^* \in \mathbb{X}' \text{ such that } \mathbb{T} r^* = r^*, \tag{2}$$

in which $\mathbb{T} : \mathbb{X}' \to \mathbb{X}'$ is a mapping. The set of all solutions of problems (1) and (2) are denoted by $C(\mathbb{T}_1, \mathbb{T}_2) \subset \mathbb{X}' \times \mathbb{X}$ and $\mathcal{F}(\mathbb{T})$, respectively.

Lemma 1.1 (see [26]). *Let $(\mathbb{X}', \rho_{\mathbb{X}'})$ and $(\mathbb{X}, \rho_{\mathbb{X}})$ be two metric spaces and $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ be two mappings such that $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$ and $\mathbb{T}_1$ is an injective mapping. Then, $\mathbb{T}_1$ has a left inverse $\mathbb{T}_{1,l}^{-1} : \mathbb{T}_1(\mathbb{X}') \to \mathbb{X}'$. Define a function $\mathbb{F} : \mathbb{X}' \times \mathbb{T}_1(\mathbb{X}') \to \mathbb{X}' \times \mathbb{T}_1(\mathbb{X}')$ by $\mathbb{F}(x, y) = (\mathbb{T}_{1,l}^{-1}(y), \mathbb{T}_2(x))$. Then, $\mathcal{F}(\mathbb{F}) = C(\mathbb{T}_1, \mathbb{T}_2)$.*

An illustrative example, found in [17], demonstrates how this lemma establishes a connection between fixed point and coincide point problems.

It appears that the initial investigations concerning the existence and uniqueness of the solution to the coincidence point/value problem (1), using a constructive method similar to the Banach Contraction Principle, were conducted by Machuca [22] in 1967. In the subsequent year, Goebel [9] further developed Machuca's findings and applied them to derive solutions of some differential equations. Jungck [15] extended Machuca's method by incorporating the concept of commuting mappings, thereby generalizing the Banach contraction principle. These studies, considered fundamental in coincidence point/value theory, have sparked numerous significant contributions from various researchers working with diverse mathematical structures in this field.

In 2008, Olatinwo [28] designed the following Classical Jungck-Ishikawa algorithm for a pair of mappings that satisfy a general contractive condition, and obtained some theoretical results concerning the strong convergence and stability of this algorithm:

```
Classical Jungck-Ishikawa algorithm (CJI algorithm)
Input: Two non-self mappings $\mathbb{T}_1$ and $\mathbb{T}_2$, initial point $s_0$, $\{\varphi_n\}_{n=0}^{\infty}, \{\psi_n\}_{n=0}^{\infty} \subset [0,1]$ and budget $N$.
1: for $n = 0, 1, 2, \ldots, N$ do
2:    $\mathbb{T}_1 r_n = (1 - \varphi_n)\mathbb{T}_1 s_n + \varphi_n \mathbb{T}_2 s_n$
      $\mathbb{T}_1 s_{n+1} = (1 - \psi_n)\mathbb{T}_1 s_n + \psi_n \mathbb{T}_2 r_n$
3: end for
Output: Approximate solution $\mathbb{T}_1 s_N$
```

CJI algorithm encompasses several noteworthy special cases: the Classical Jungck-Picard algorithm (CJP algorithm) [15] when $\varphi_n = 0$ and $\psi_n = 1$ for all $n \in \mathbb{N}_0$ ($\mathbb{N} \cup \{0\} = \{0, 1, 2, \ldots\}$), and the Classical Jungck-Mann algorithm (CJM algorithm) [34] when $\varphi_n = 0$ for all $n \in \mathbb{N}_0$.

In 2013, Hussain et al. [14] defined various iteration algorithms to investigate the distinct properties of the coincidence points/values of the pairs of mappings satisfying certain contractive conditions. Among these, the following extension of the original normal-S algorithm, introduced by Sahu [32], which has gained significant attention from numerous researchers owing to its exceptional ability to approximate solutions effectively across a diverse set of problems, stands out:

```
Classical Jungck normal-S algorithm (CJNS algorithm)
Input: Two non-self mappings $\mathbb{T}_1$ and $\mathbb{T}_2$, initial point $s_0$, $\{\varphi_n\}_{n=0}^{\infty} \subset [0,1]$ and budget $N$.
1: for $n = 0, 1, 2, \ldots, N$ do
2:    $\mathbb{T}_1 r_n = (1 - \varphi_n)\mathbb{T}_1 s_n + \varphi_n \mathbb{T}_2 s_n$
      $\mathbb{T}_1 s_{n+1} = \mathbb{T}_2 r_n$
3: end for
Output: Approximate solution $\mathbb{T}_1 s_N$
```

It is straightforward to notice that CJNS algorithm is a special case of CJI algorithm. It stands independently from the CJM algorithm and reduces to the CJP algorithm when $\varphi_n = 0$ for all $n \in \mathbb{N}_0$.

To the best of our knowledge, there is no study in the literature that uses an inertial algorithm designed for a pair of mappings to approximate solutions to the coincidence point/value problems given in (1). The main motivation for the work [17] was primarily to fill this gap in the literature. During the preparation of [17], the authors designed the following algorithm by incorporating a momentum (inertial) term into the Jungck normal-*S* algorithm with the aim of accelerating it (however, as explained below, this algorithm could not be used in [17]):

```
Jungck normal-S inertial type algorithm (JNSI algorithm, version 1)
Input: Two non-self mappings $\mathbb{T}_1$ and $\mathbb{T}_2$, initial points $x_{-1}, x_0$, $\{\alpha_n\}_{n=0}^{\infty} \subset [0,1]$ and budget $N$.
for $n = 0, 1, 2, \ldots, N$ do
    $\mathbb{T}_1 z_n = \mathbb{T}_1 x_n + \theta_n (\mathbb{T}_1 x_n - \mathbb{T}_1 x_{n-1})$
    $\mathbb{T}_1 y_n = (1 - \alpha_n) \mathbb{T}_1 z_n + \alpha_n \mathbb{T}_2 z_n$
    $\mathbb{T}_1 x_{n+1} = \mathbb{T}_2 y_n$
end for
Output: Approximate solution $\mathbb{T}_1 x_N$
```

Here, the sequence $\{\theta_n\}_{n=0}^{\infty}$ has been modified as follows, inspired by the choosing in [33, p. 881].

$$0 \leq \theta_n \leq \overline{\theta}_n := \begin{cases} \min\left\{ \dfrac{n}{n+\eta}, \dfrac{\epsilon_n}{\|\mathbb{T}_1 x_n - \mathbb{T}_1 x_{n-1}\|} \right\}, & \text{if } \mathbb{T}_1 x_n \neq \mathbb{T}_1 x_{n-1}, \\ \dfrac{n}{n+\eta}, & \text{if } \mathbb{T}_1 x_n = \mathbb{T}_1 x_{n-1}, \end{cases} \tag{3}$$

for some $\eta \geq 2$ and $\{\epsilon_n\}_{n=0}^{\infty} \subset (0, \infty)$. However, as could be seen from numerous numerical experiments (see also Table 4 in this paper), contrary to the expectations that the momentum term $\theta_n (\mathbb{T}_1 x_n - \mathbb{T}_1 x_{n-1})$ incorporated into CJNS would effectively speed up the iteration, it had a negligible effect, so that an acceleration of the JNSI algorithm (version 1) in the work [17] was presented as an open problem. In particular, the question remained whether it is possible to design an inertial Jungck-type iteration algorithm that is more efficient in terms of convergence rate than its classical counterpart? In addition, how does this algorithm behave in terms of data dependence and stability properties? We are now able to provide a positive answer to this problem and as a result propose the following algorithm:

```
Jungck normal-inertial type algorithm (JNSI algorithm, version 2)
Input: Two non-self mappings $\mathbb{T}_1$ and $\mathbb{T}_2$, initial points $x_{-1}$, $x_0$, $\{\alpha_n\}_{n=0}^{\infty} \subset [0,1]$ and budget $N$.
for $n = 0, 1, 2, \ldots, N$ do
    $\mathbb{T}_1 z_n = \mathbb{T}_2 x_n + \theta_n (\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1})$
    $\mathbb{T}_1 y_n = (1 - \alpha_n) \mathbb{T}_1 z_n + \alpha_n \mathbb{T}_2 z_n$
    $\mathbb{T}_1 x_{n+1} = \mathbb{T}_2 y_n$
end for
Output: Approximate solution $\mathbb{T}_1 x_N$
```

in which the sequence $\{\theta_n\}_{n=0}^{\infty}$ is defined the following (We note that this sequence has been modified, inspired by the choosing in [33, p. 881]),

$$0 \leq \theta_n \leq \overline{\theta}_n := \begin{cases} \min\left\{ \dfrac{n}{n+\eta}, \dfrac{\epsilon_n}{\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\|} \right\}, & \text{if } \mathbb{T}_2 x_n \neq \mathbb{T}_2 x_{n-1}, \\ \dfrac{n}{n+\eta}, & \text{if } \mathbb{T}_2 x_n = \mathbb{T}_2 x_{n-1}, \end{cases} \tag{4}$$

for some $\eta \geq 2$ and $\{\epsilon_n\}_{n=0}^{\infty} \subset (0, \infty)$. Similar to JNSI algorithm (version 2) with (4), Jungck-Ishikawa inertial type algorithm (JII algorithm version 2) for the pairs of mappings can be defined as follows.

```
Jungck-Ishikawa inertial type algorithm (JII algorithm, version 2)
Input: Two non-self mappings $\mathbb{T}_1$ and $\mathbb{T}_2$, initial points $x_{-1}$, $x_0$, $\{\alpha_n\}_{n=0}^{\infty}, \{\beta_n\}_{n=0}^{\infty} \subset [0,1]$ and budget $N$.
for $n = 0, 1, 2, \ldots, N$ do
    $\mathbb{T}_1 z_n = \mathbb{T}_2 x_n + \theta_n (\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1})$
    $\mathbb{T}_1 y_n = (1 - \alpha_n) \mathbb{T}_1 z_n + \alpha_n \mathbb{T}_2 z_n$
    $\mathbb{T}_1 x_{n+1} = (1 - \beta_n) \mathbb{T}_1 z_n + \beta_n \mathbb{T}_2 y_n$
end for
Output: Approximate solution $\mathbb{T}_1 x_N$
```

in which the sequence $\{\theta_n\}_{n=0}^{\infty}$ as in (4).

In this article, we are employing three distinct types of algorithms. To prevent any potential confusion, we are categorizing these algorithms in the following manner:

(i) Algorithms that lack an inertial term will be denoted as classical algorithms.

(ii) Algorithms that incorporate an inertial term $\theta_n(\mathbb{T}_1x_n - \mathbb{T}_1x_{n-1})$ will be identified as version 1 algorithms.

(iii) Algorithms that incorporate an inertial term $\theta_n(\mathbb{T}_2x_n - \mathbb{T}_2x_{n-1})$ will be referred to as version 2 algorithms.

Remark 1.1. Let $\mathbb{X}' \neq \emptyset$ be an arbitrary set and $\mathbb{X}$ be a Banach space and $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ be two mappings. If the following conditions are satisfied, then JNSI algorithm (ver. 2) and JII algorithm (ver. 2) are well-defined.

(i) $(1+\lambda)\mathbb{T}_2(\mathbb{X}') - \lambda\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_2(\mathbb{X}')$, for all $\lambda \in [0,1]$. In particular, if $\mathbb{T}_2(\mathbb{X}')$ is a vector subspace in $\mathbb{X}$, then the condition is satisfied.

(ii) $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$.

(iii) $\mathbb{T}_1$ is injective and $\mathbb{T}_1(\mathbb{X}')$ is a convex set.

Remark 1.2. If the conditions (i)–(iii) in Remark 1.1 are satisfied by replacing $\mathbb{T}_2$ with $\mathbb{T}_1$ in the condition (i), then JNSI algorithm (version 1) is well-defined. If the conditions (ii)–(iii) in Remark 1.1 are satisfied, then CJI and CJNS algorithms are well-defined.

Remark 1.3. The following special cases of JII algorithm (ver. 2) are worth mentioning.

(i) When $\alpha_n = 0$ and $\beta_n = 1$ for all $n \in \mathbb{N}_0$, it gives rise to a Jungck-Picard inertial type algorithm (JPI algorithm, ver. 2).

(ii) When $\alpha_n = 0$ for all $n \in \mathbb{N}_0$, it yields a Jungck-Mann inertial type algorithm (JMI algorithm, ver. 2).

(iii) When $\beta_n = 1$ for all $n \in \mathbb{N}_0$, it reduced to JNSI algorithm, ver. 2.

In general, the pair of mappings $(\mathbb{T}_1, \mathbb{T}_2)$ in (1) is expected to satisfy certain contractive conditions so that the coincidence point technique can be applied in order find solutions of (1), that is, coincidence points of $(\mathbb{T}_1, \mathbb{T}_2)$, and hence, the classes of mappings that satisfies various contractive conditions are important in the studies of coincidence points. We shall recall some of them in the following.

Definition 1.1. A pair of mappings $(\mathbb{T}_1, \mathbb{T}_2)$ with $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ and $C(\mathbb{T}_1, \mathbb{T}_2) \neq \emptyset$ is said to be

(i) a quasi contractive if there is a number $\delta \in [0,1)$ satisfying

$$\left\|\mathbb{T}_2 r - p\right\| \leq \delta \left\|\mathbb{T}_1 r - p\right\|,$$

for all $(r,(x,p)) \in \mathbb{X}' \times C(\mathbb{T}_1, \mathbb{T}_2)$;

(ii) a quasi (L,δ)-contractive if there are the numbers $\delta \in [0,1)$ and $L \geq 0$ satisfying

$$\left\|\mathbb{T}_2 r - p\right\| \leq \delta \left\|\mathbb{T}_1 r - p\right\| + L\left\|\mathbb{T}_1 r - \mathbb{T}_2 r\right\|,$$

for all $(r,(x,p)) \in \mathbb{X}' \times C(\mathbb{T}_1, \mathbb{T}_2)$.

Keten Çopur et al. [17] obtained an inclusion relation between the pair of mappings in (i) and (ii) of Definition 1.1 (for detail, see [17]).

We proceed with the following organization for the rest of the paper. In Section 2, we demonstrate that JNSI algorithm (versions 2 and 1) converges to the coincidence value of the pair $(\mathbb{T}_1, \mathbb{T}_2)$ of mappings satisfying the quasi (L,δ)-contractive condition. Furthermore, we show that JNSI algorithm (ver. 2) is weakly $(\mathbb{T}_1, \mathbb{T}_2)$-stable, and we construct a data dependency result related to the coincidence value of the pair $(\mathbb{T}_1, \mathbb{T}_2)$ of mappings satisfying the quasi (L,δ)-contractive condition. In Section 3, in order to evaluate the reliability, accuracy and effectiveness of the previous theoretical findings and to demonstrate the superior performance of JNSI algorithm (ver. 2) in solving complex problems, particularly when compared to various algorithms, including its classical counterpart, we present academic numerical examples in infinite dimensional spaces. In Section 4, additionally to Section 3, we present various examples related to integral and differential equations, each of which can be considered as an application. Moreover, we use a variety of algorithms, including JNSI algorithm (version 2), to create polynomigraphs that hold educational and artistic value and are visually appealing.

The definitions and lemma given below play an important role in obtaining our results.

Definition 1.2 ([4]). Let $\{x_n\}_{n=0}^{\infty}$ be a sequence in a metric space (X, d). If for any $k \in \mathbb{N}$, there exists a $\zeta = \zeta(k)$ satisfying

$$(\forall m \geq k) \quad d(x_m, y_m) \leq \zeta,$$

then, the sequence $\{y_n\}_{n=0}^{\infty}$ in X is called an approximate sequence of $\{x_n\}_{n=0}^{\infty}$.

Definition 1.3 ([35]). Consider a metric space $(\mathbb{X}, d)$, a non-empty set $\mathbb{X}'$ and two mappings $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$. Let $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$ and p be a coincidence value of $(\mathbb{T}_1, \mathbb{T}_2)$. Additionally, consider an iteration sequence $\{\mathbb{T}_1 x_n\}_{n=0}^{\infty}$ which is obtained by an algorithm in the form given below

$$\begin{aligned} & x_0 \in \mathbb{X}', \\ (\forall n \in \mathbb{N}_0) \quad \mathbb{T}_1 x_{n+1} &= f(\mathbb{T}_1, \mathbb{T}_2, x_n), \end{aligned} \tag{5}$$

where f is a function and x_0 is the initial approximation. If $\{\mathbb{T}_1 x_n\}_{n=0}^{\infty}$ converges to p, and for any approximate sequence $\{\mathbb{T}_1 y_n\}_{n=0}^{\infty}$ of $\{\mathbb{T}_1 x_n\}_{n=0}^{\infty}$, the condition $\lim\limits_{n\to\infty} d(\mathbb{T}_1 y_{n+1}, f(\mathbb{T}_1, \mathbb{T}_2, y_n)) = 0$ implies $\lim\limits_{n\to\infty} \mathbb{T}_1 y_n = p$, then (5) is called weakly stable with respect to $(\mathbb{T}_1, \mathbb{T}_2)$ or, briefly, weakly $(\mathbb{T}_1, \mathbb{T}_2)$-stable.

Lemma 1.2 (see [20]). *Let $\left\{\Phi_n^i\right\}_{n=0}^{\infty}$, $i = 1, 2$, be two real sequences satisfying $\Phi_n^1 \geq 0$, $\Phi_n^2 \geq 0$, $\lim\limits_{n\to\infty} \Phi_n^2 = 0$ and*

$$(\forall n \in \mathbb{N}_0) \quad \Phi_{n+1}^1 \leq \varrho \Phi_n^1 + \Phi_n^2,$$

in which $\varrho \in [0, 1)$ is a constant. Then, $\lim\limits_{n\to\infty} \Phi_n^1 = 0$.

2. Main results

In this section, we assume that $\mathbb{X}' \neq \emptyset$ is an arbitrary set, $\mathbb{X}$ is a Banach space, $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ is a pair of quasi (L, δ)-contractive mappings such that $\Theta := (\delta + L)/(1 - L) < 1$ and the pair satisfies the conditions (i)–(iii) of Remark 1.1. Also, we assume that $p \in \mathbb{X}$ is a coincidence value of $(\mathbb{T}_1, \mathbb{T}_2)$.

Theorem 2.1. *Let $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ be the sequence generated by JNSI algorithm (version 2). If $\sum\limits_{n=0}^{\infty} \epsilon_n < \infty$, then $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ converges to p.*

Proof. By JNSI algorithm (ver. 2) and Definition 1.1 (ii), we obtain

$$(\forall n \in \mathbb{N}_0) \quad \left\|\mathbb{T}_1 x_{n+1} - p\right\| \leq (\delta + L) \left\|\mathbb{T}_1 y_n - p\right\| + L \left\|p - \mathbb{T}_2 y_n\right\|$$

and, thus for all $n \in \mathbb{N}_0$

$$\left\|\mathbb{T}_1 x_{n+1} - p\right\| \leq \Theta \left\|\mathbb{T}_1 y_n - p\right\|. \tag{6}$$

Also, by JNSI algorithm (version 2), we obtain the below inequalities

$$\begin{aligned} \left\|\mathbb{T}_1 y_n - p\right\| &\leq (1 - \alpha_n) \left\|\mathbb{T}_1 z_n - p\right\| + \alpha_n \left\|\mathbb{T}_2 z_n - p\right\|, \\ \left\|\mathbb{T}_1 z_n - p\right\| &\leq \left\|\mathbb{T}_2 x_n - p\right\| + \theta_n \left\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\right\|. \end{aligned} \tag{7}$$

On the other hand, by Definition 1.1 (ii), we get, for all $n \in \mathbb{N}_0$

$$\left\|\mathbb{T}_2 z_n - p\right\| \leq \Theta \left\|\mathbb{T}_1 z_n - p\right\| \quad \text{and} \quad \left\|\mathbb{T}_2 x_n - p\right\| \leq \Theta \left\|\mathbb{T}_1 x_n - p\right\|. \tag{8}$$

Combining (6)–(8), we obtain, for all $n \in \mathbb{N}$

$$\left\|\mathbb{T}_1 x_{n+1} - p\right\| \leq \Theta^2\Big(1 - \alpha_n (1 - \Theta)\Big) \left\|\mathbb{T}_1 x_n - p\right\| + \Theta\Big(1 - \alpha_n (1 - \Theta)\Big)\theta_n \left\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\right\|. \tag{9}$$

Since $\Theta\left[1 - \alpha_n (1 - \Theta)\right] < 1$, the inequality in (9) implies that

$$\left\|\mathbb{T}_1 x_{n+1} - p\right\| \leq \Theta \left\|\mathbb{T}_1 x_n - p\right\| + \theta_n \left\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\right\| \tag{10}$$

for all $n \in \mathbb{N}$. By the definition of $\{\theta_n\}_{n=0}^{\infty}$ in (4), the condition $\sum\limits_{n=0}^{\infty} \epsilon_n < \infty$ and the squeeze principle, we get $\lim\limits_{n\to\infty} \theta_n \left\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\right\| = 0$. Now, an application of Lemma 1.2 to (10) yields that $\mathbb{T}_1 x_n \to p$ as $n \to \infty$. □

Remark 2.1. From the proof of Theorem 2.1, we observe that if the condition $\sum\limits_{n=0}^{\infty} \epsilon_n < \infty$ is replaced by $\lim\limits_{n\to\infty} \epsilon_n = 0$, then Theorem 2.1 remains true.

Remark 2.2. We observed with Example 3.1 that when the condition $\delta + 2L < 1$ is removed, the convergence result in Theorem 2.1 may not be guaranteed.

Theorem 2.2. *Let $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ be the sequence generated by JII algorithm (version 2) with $\{\theta_n\}_{n=0}^{\infty}$ in* (4)*. If $\sum\limits_{n=0}^{\infty} \epsilon_n < \infty$, then $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ converges to p.*

Proof. The proof of this theorem is omitted as similar to Theorem 2.1. □

By considering Remark 1.3 and Theorem 2.2 simultaneously, one can easily reach the following conclusion.

Corollary 2.1. *Under the assumptions of Theorem 2.2, each of the sequences generated by algorithms JPI (version 2) and JMI (version 2) converges to p.*

Theorem 2.3. *Let $a \in (0, 1]$ and $\{\beta_n\}_{n=0}^{\infty}$ be a sequence such that $a \leq \beta_n \leq 1$, for all $n \in \mathbb{N}$ and $\sum\limits_{n=0}^{\infty} \epsilon_n < \infty$. Then, the sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JII algorithm (version 1) converges to p.*

Proof. Using the definition of JII algorithm (version 1), by similar operations in Theorem 2.1, we obtain the following inequality for all $n \in \mathbb{N}$

$$\begin{aligned}
\left\|\mathbb{T}_1 x_{n+1} - p\right\| &\leq \left[1 - \beta_n + \beta_n \Theta \left\{1 - \alpha_n (1 - \Theta)\right\}\right] \left\|\mathbb{T}_1 x_n - p\right\| \\
&\quad + \left[1 - \beta_n + \beta_n \Theta \left\{1 - \alpha_n (1 - \Theta)\right\}\right] \theta_n \left\|\mathbb{T}_1 x_n - \mathbb{T}_1 x_{n-1}\right\| \\
&\leq \left[1 - \beta_n (1 - \Theta) - \Theta \alpha_n \beta_n (1 - \Theta)\right] \left\|\mathbb{T}_1 x_n - p\right\| + \left[1 - \beta_n (1 - \Theta) - \Theta \alpha_n \beta_n (1 - \Theta)\right] \epsilon_n \\
&= \left[1 - \beta_n (1 - \Theta)(1 + \Theta \alpha_n)\right] \left\|\mathbb{T}_1 x_n - p\right\| + \left[1 - \beta_n (1 - \Theta)(1 + \Theta \alpha_n)\right] \epsilon_n. \qquad (11)
\end{aligned}$$

Since $1 - \beta_n(1 - \Theta^2) \leq 1 - \beta_n(1 - \Theta)(1 + \Theta\alpha_n) \leq 1 - \beta_n(1 - \Theta) \leq 1 - a(1 - \Theta)$, for all $n \in \mathbb{N}$, the inequality in (11) implies that, for all $n \in \mathbb{N}$,

$$\left\|\mathbb{T}_1 x_{n+1} - p\right\| \leq \Big(1 - a(1 - \Theta)\Big) \left\|\mathbb{T}_1 x_n - p\right\| + \Big(1 - a(1 - \Theta)\Big) \epsilon_n. \tag{12}$$

Since $1 - a(1 - \Theta) < 1$ and $\sum\limits_{n=0}^{\infty} \epsilon_n < \infty$, an application of Lemma 1.2 to (12) yields that $\mathbb{T}_1 x_n \to p$ as $n \to \infty$. □

Considering Theorem 2.3 and the version 1 of Remark 1.3 simultaneously, one can easily reach the following conclusion.

Corollary 2.2. *Under the assumptions of Theorem 2.3, each of the sequences generated by algorithms JPI (version 1) and JMI (version 1) and JNSI (version 1) converges to p.*

Theorem 2.4. *Let $\{\mathbb{T}_1 x_n\}_{n=0}^{\infty}$ be the sequence generated by JNSI algorithm (version 2) under the conditions in Theorem 2.1 and $\{\mathbb{T}_1 u_n\}_{n=0}^{\infty}$ be an approximate sequence of $\{\mathbb{T}_1 x_n\}_{n=0}^{\infty}$. We define a sequence $\{\varepsilon_n\}_{n=1}^{\infty} \subset \mathbb{R}^+$ as follows*

$$\begin{aligned}
\varepsilon_n &= \|\mathbb{T}_1 u_{n+1} - \mathbb{T}_2 v_n\|, \\
\mathbb{T}_1 v_n &= (1-\alpha_n)\,\mathbb{T}_1 w_n + \alpha_n \mathbb{T}_2 w_n, \\
\mathbb{T}_1 w_n &= \mathbb{T}_2 u_n + \widetilde{\theta}_n\,(\mathbb{T}_2 u_n - \mathbb{T}_2 u_{n-1}),
\end{aligned} \tag{13}$$

in which $\{\alpha_n\}_{n=1}^{\infty}$ is as in Theorem 2.1 and

$$0 \le \widetilde{\theta}_n \le \overline{\overline{\theta}}_n := \begin{cases} \min\left\{\dfrac{n}{n+\eta}, \dfrac{\widetilde{\epsilon}_n}{\|\mathbb{T}_2 u_n - \mathbb{T}_2 u_{n-1}\|}\right\}, & \text{if } \mathbb{T}_2 u_n \neq \mathbb{T}_2 u_{n-1}, \\ \dfrac{n}{n+\eta}, & \text{if } \mathbb{T}_2 u_n = \mathbb{T}_2 u_{n-1}, \end{cases} \tag{14}$$

for some $\eta \ge 2$ and $\{\widetilde{\epsilon}_n\}_{n=1}^{\infty} \subset (0,\infty)$. If $\sum_{n=1}^{\infty} \widetilde{\epsilon}_n < \infty$, then $\lim_{n\to\infty} \varepsilon_n = 0$ if and only if that $\lim_{n\to\infty} \mathbb{T}_1 u_n = p$. In particular, JNSI algorithm (version 2) is weakly $(\mathbb{T}_1, \mathbb{T}_2)$-stable.

Proof. Assume that $\lim_{n\to\infty} \varepsilon_n = 0$. We shall show that $\lim_{n\to\infty} \mathbb{T}_1 u_n = p$. It follows from Definition 1.1 (ii), JNSI algorithn (version 2) and (13), for all $n \in \mathbb{N}$,

$$\begin{aligned}
\left\|\mathbb{T}_1 u_{n+1} - p\right\| &\le \varepsilon_n + \left\|\mathbb{T}_2 v_n - p\right\| + \left\|p - \mathbb{T}_2 y_n\right\| + \left\|\mathbb{T}_1 x_{n+1} - p\right\| \\
&\le \varepsilon_n + \delta\left\|\mathbb{T}_1 v_n - p\right\| + L\left\|\mathbb{T}_2 v_n - \mathbb{T}_1 v_n\right\| \\
&\quad + \delta\left\|\mathbb{T}_1 y_n - p\right\| + L\left\|\mathbb{T}_2 y_n - \mathbb{T}_1 y_n\right\| + \left\|\mathbb{T}_1 x_{n+1} - p\right\|.
\end{aligned} \tag{15}$$

On the other hand, by Definition 1.1 (ii), we have, for all $n \in \mathbb{N}_0$,

$$\|\mathbb{T}_2 v_n - \mathbb{T}_1 v_n\| \le \kappa\left\|\mathbb{T}_1 v_n - p\right\| \quad \text{and} \quad \left\|\mathbb{T}_2 y_n - \mathbb{T}_1 y_n\right\| \le \kappa\left\|\mathbb{T}_1 y_n - p\right\|. \tag{16}$$

in which $\kappa = (\delta+1)/(1-L)$. Inserting (16) into (15), we get, for all $n \in \mathbb{N}$,

$$\left\|\mathbb{T}_1 u_{n+1} - p\right\| \le \varepsilon_n + \Theta\left(\left\|\mathbb{T}_1 v_n - p\right\| + \left\|\mathbb{T}_1 y_n - p\right\|\right) + \left\|\mathbb{T}_1 x_{n+1} - p\right\|. \tag{17}$$

Following the same lines as in the proof of Theorem 2.1, we obtain

$$\begin{aligned}
&\left\|\mathbb{T}_1 y_n - p\right\| \le \left(1 - \alpha_n (1-\Theta)\right)\left(\Theta\left\|\mathbb{T}_1 x_n - p\right\| + \theta_n\left\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\right\|\right), \\
&\left\|\mathbb{T}_1 v_n - p\right\| \le \left(1 - \alpha_n (1-\Theta)\right)\left(\Theta\left\|\mathbb{T}_1 u_n - p\right\| + \widetilde{\theta}_n\left\|\mathbb{T}_2 u_n - \mathbb{T}_2 u_{n-1}\right\|\right).
\end{aligned} \tag{18}$$

Substituting the inequalities in (18) into (17), we get, for all $n \in \mathbb{N}$,

$$\begin{aligned}
\left\|\mathbb{T}_1 u_{n+1} - p\right\| &\le \varepsilon_n + \Theta^2\left(1 - \alpha_n (1-\Theta)\right)\left(\left\|\mathbb{T}_1 u_n - p\right\| + \left\|\mathbb{T}_1 x_n - p\right\|\right) \\
&\quad + \Theta\left(1 - \alpha_n (1-\Theta)\right)\widetilde{\theta}_n \|\mathbb{T}_2 u_n - \mathbb{T}_2 u_{n-1}\| \\
&\quad + \Theta\left(1 - \alpha_n (1-\Theta)\right)\theta_n \|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\| + \left\|\mathbb{T}_1 x_{n+1} - p\right\|.
\end{aligned}$$

Then, using $\Theta < 1$, $[1 - \alpha_n (1-\Theta)] < 1$, (4) and (14), we get

$$\left\|\mathbb{T}_1 u_{n+1} - p\right\| \le \Theta\left\|\mathbb{T}_1 u_n - p\right\| + \left\|\mathbb{T}_1 x_n - p\right\| + \left\|\mathbb{T}_1 x_{n+1} - p\right\| + \varepsilon_n + \epsilon_n + \widetilde{\epsilon}_n. \tag{19}$$

By the hypotheses, since

$$\lim_{n\to\infty}\left\|\mathbb{T}_1 x_n - p\right\| = \lim_{n\to\infty} \varepsilon_n = \lim_{n\to\infty} \epsilon_n = \lim_{n\to\infty} \widetilde{\epsilon}_n = 0,$$

by Lemma 1.2, we get $\lim\limits_{n\to\infty}\left\|\mathbb{T}_1 u_n - p\right\| = 0$, that is, $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ is weakly $(\mathbb{T}_1, \mathbb{T}_2)$-stable.

Suppose on the contrary that $\lim\limits_{n\to\infty}\mathbb{T}_1 u_n = p$. We shall show that $\lim\limits_{n\to\infty}\varepsilon_n = 0$. By (13), we get

$$\varepsilon_n \leq \|\mathbb{T}_1 u_{n+1} - \mathbb{T}_1 x_{n+1}\| + \left\|\mathbb{T}_2 y_n - p\right\| + \left\|p - \mathbb{T}_2 v_n\right\|, \text{ for all } n \in \mathbb{N}.$$

Following the same lines as in first part of the proof, we get for all $n \in \mathbb{N}$

$$\begin{aligned}
\varepsilon_n &\leq \|\mathbb{T}_1 u_{n+1} - \mathbb{T}_1 x_{n+1}\| + \Theta\left(\left\|\mathbb{T}_1 y_n - p\right\| + \left\|\mathbb{T}_1 v_n - p\right\|\right) \\
&\leq \|\mathbb{T}_1 u_{n+1} - \mathbb{T}_1 x_{n+1}\| + \Theta^2\Big(1 - \alpha_n(1-\Theta)\Big)\left\{\left\|\mathbb{T}_1 x_n - p\right\| + \left\|\mathbb{T}_1 u_n - p\right\|\right\} \\
&\quad + \Theta\Big(1 - \alpha_n(1-\Theta)\Big)\left(\theta_n\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\| + \widetilde{\theta}_n\|\mathbb{T}_2 u_n - \mathbb{T}_2 u_{n-1}\|\right).
\end{aligned}$$

Using (4), (14), $\Theta < 1$, $[1 - \alpha_n(1-\Theta)] < 1$, we get, for all $n \in \mathbb{N}$

$$\varepsilon_n \leq \|\mathbb{T}_1 u_{n+1} - \mathbb{T}_1 x_{n+1}\| + \Theta^2\left\{\left\|\mathbb{T}_1 x_n - p\right\| + \left\|\mathbb{T}_1 u_n - p\right\|\right\} + \Theta\Big(\epsilon_n + \widetilde{\epsilon}_n\Big). \tag{20}$$

By the hypotheses, since

$$\lim_{n\to\infty}\left\|\mathbb{T}_1 x_n - p\right\| = \lim_{n\to\infty}\left\|\mathbb{T}_1 u_n - p\right\| = \lim_{n\to\infty}\epsilon_n = \lim_{n\to\infty}\widetilde{\epsilon}_n = 0,$$

passing to the limit in (20), we get $\lim\limits_{n\to\infty}\varepsilon_n = 0$. □

Theorem 2.5. *Let $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ be the sequence generated by JNSI algorithm (version 2). Let $\widetilde{\mathbb{T}}_1, \widetilde{\mathbb{T}}_2 : \mathbb{X}' \to \mathbb{X}$ be two mappings satisfying the conditions of* (i)–(iii) *of Remark 1.1, $\widetilde{p}$ be a coincidence value of $(\widetilde{\mathbb{T}}_1, \widetilde{\mathbb{T}}_2)$ and initial points $\widetilde{x}_{-1}, \widetilde{x}_0 \in \mathbb{X}'$ be given. We consider the sequence $\{\widetilde{\mathbb{T}}_1\widetilde{x}_n\}_{n=1}^{\infty}$ generated by*

$$\begin{aligned}
\widetilde{\mathbb{T}}_1\widetilde{z}_n &= \widetilde{\mathbb{T}}_2\widetilde{x}_n + \widetilde{\theta}_n(\widetilde{\mathbb{T}}_2\widetilde{x}_n - \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}), \\
\widetilde{\mathbb{T}}_1\widetilde{y}_n &= (1-\alpha_n)\widetilde{\mathbb{T}}_1\widetilde{z}_n + \alpha_n\widetilde{\mathbb{T}}_2\widetilde{z}_n, \\
\widetilde{\mathbb{T}}_1\widetilde{x}_{n+1} &= \widetilde{\mathbb{T}}_2\widetilde{y}_n,
\end{aligned} \tag{21}$$

in which

$$0 \leq \widetilde{\theta}_n \leq \overline{\widetilde{\theta}}_n := \begin{cases} \min\left\{\dfrac{n}{n+\eta}, \dfrac{\widetilde{\epsilon}_n}{\left\|\widetilde{\mathbb{T}}_2\widetilde{x}_n - \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}\right\|}\right\}, & \text{if } \widetilde{\mathbb{T}}_2\widetilde{x}_n \neq \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}, \\ \dfrac{n}{n+\eta}, & \text{if } \widetilde{\mathbb{T}}_2\widetilde{x}_n = \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}, \end{cases} \tag{22}$$

for some $\eta \geq 2$ and $\{\widetilde{\epsilon}_n\}_{n=0}^{\infty} \subset (0, \infty)$.

Suppose that all the hypotheses of Theorem 2.1 hold. In addition, if

(C1) *there exist maximum admissible errors μ_1, $\mu_2 > 0$ such that $\|\mathbb{T}_1 r - \widetilde{\mathbb{T}}_1 r\| \leq \mu_1$ and $\|\mathbb{T}_2 r - \widetilde{\mathbb{T}}_2 r\| \leq \mu_2$, for all $r \in \mathbb{X}'$,*

(C2) $\sum\limits_{n=0}^{\infty}\widetilde{\epsilon}_n < \infty$,

(C3) $\lim\limits_{n\to\infty}\widetilde{\mathbb{T}}_1\widetilde{x}_n = \widetilde{p}$,

then it holds that

$$\left\|\widetilde{p} - p\right\| \leq \frac{\Big(\mu_2 + \Theta\mu_1\Big)\Big(\Theta^2 + \Theta + 1\Big)}{1 - \Theta^2}. \tag{23}$$

Proof. It follows from JNSI algorithm (version 2), (21) and (C1) that for all $n \in \mathbb{N}_0$

$$\begin{aligned} \left\|\widetilde{\mathbb{T}}_1\widetilde{x}_{n+1} - \mathbb{T}_1 x_{n+1}\right\| &\leq \left\|\widetilde{\mathbb{T}}_2\widetilde{y}_n - \mathbb{T}_2\widetilde{y}_n\right\| + \left\|\mathbb{T}_2\widetilde{y}_n - \mathbb{T}_2 y_n\right\| \\ &\leq \mu_2 + \left\|\mathbb{T}_2\widetilde{y}_n - p\right\| + \left\|p - \mathbb{T}_2 y_n\right\|. \end{aligned} \tag{24}$$

Using Definition 1.1 (ii) and (C1) in (24), we get, for all $n \in \mathbb{N}_0$,

$$\begin{aligned} \left\|\widetilde{\mathbb{T}}_1\widetilde{x}_{n+1} - \mathbb{T}_1 x_{n+1}\right\| &\leq \mu_2 + \Theta\left(\|\mathbb{T}_1\widetilde{y}_n - \widetilde{\mathbb{T}}_1\widetilde{y}_n\| + \|\widetilde{\mathbb{T}}_1\widetilde{y}_n - p\| + \|\mathbb{T}_1 y_n - p\|\right) \\ &\leq \mu_2 + \Theta\mu_1 + \Theta\left(\|\widetilde{\mathbb{T}}_1\widetilde{y}_n - p\| + \|\mathbb{T}_1 y_n - p\|\right). \end{aligned} \tag{25}$$

On the other hand, by (21), (C1) and Definition 1.1 (ii), we have

$$\begin{aligned} \left\|\widetilde{\mathbb{T}}_1\widetilde{y}_n - p\right\| &\leq (1-\alpha_n)\,\|\widetilde{\mathbb{T}}_1\widetilde{z}_n - p\| + \alpha_n\|\widetilde{\mathbb{T}}_2\widetilde{z}_n - \mathbb{T}_2\widetilde{z}_n\| + \alpha_n\|\mathbb{T}_2\widetilde{z}_n - p\| \\ &\leq [1-\alpha_n(1-\Theta)]\,\|\widetilde{\mathbb{T}}_1\widetilde{z}_n - p\| + \alpha_n\mu_2 + \alpha_n\Theta\mu_1 \\ &\leq [1-\alpha_n(1-\Theta)]\left(\mu_2 + \|\mathbb{T}_2\widetilde{x}_n - p\| + \widetilde{\theta}_n\left\|\widetilde{\mathbb{T}}_2\widetilde{x}_n - \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}\right\|\right) + \alpha_n(\mu_2 + \Theta\mu_1) \\ &\leq [1-\alpha_n(1-\Theta)]\left(\mu_2 + \Theta\|\mathbb{T}_1\widetilde{x}_n - p\| + \widetilde{\theta}_n\|\widetilde{\mathbb{T}}_2\widetilde{x}_n - \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}\|\right) + \alpha_n(\mu_2 + \Theta\mu_1) \\ &\leq [1-\alpha_n(1-\Theta)]\,\Theta\|\widetilde{\mathbb{T}}_1\widetilde{x}_n - p\| + [1-\alpha_n(1-\Theta)]\,[\mu_2 + \Theta\mu_1] \\ &\quad + [1-\alpha_n(1-\Theta)]\,\widetilde{\theta}_n\|\widetilde{\mathbb{T}}_2\widetilde{x}_n - \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}\| + \alpha_n[\mu_2 + \Theta\mu_1]. \end{aligned} \tag{26}$$

Inserting (18) and (26) into (25) and using $\Theta < 1$, $\alpha_n < 1$, $[1-\alpha_n(1-\Theta)] < 1$, we get for all $n \in \mathbb{N}_0$

$$\begin{aligned} \|\widetilde{\mathbb{T}}_1\widetilde{x}_{n+1} - \mathbb{T}_1 x_{n+1}\| &\leq \mu_2 + \Theta\mu_1 + \Theta^2[1-\alpha_n(1-\Theta)]\,\|\widetilde{\mathbb{T}}_1\widetilde{x}_n - p\| + \Theta[1-\alpha_n(1-\Theta)]\,[\mu_2 + \Theta\mu_1] \\ &\quad + \Theta[1-\alpha_n(1-\Theta)]\,\widetilde{\theta}_n\|\widetilde{\mathbb{T}}_2\widetilde{x}_n - \widetilde{\mathbb{T}}_2\widetilde{x}_{n-1}\| + \Theta\alpha_n[\mu_2 + \Theta\mu_1] \\ &\quad + \Theta^2[1-\alpha_n(1-\Theta)]\,\|\mathbb{T}_1 x_n - p\| + \Theta[1-\alpha_n(1-\Theta)]\,\theta_n\,\|\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}\| \\ &\leq \mu_2 + \Theta\mu_1 + \Theta^2\left\|\widetilde{\mathbb{T}}_1\widetilde{x}_n - p\right\| + (\Theta + \Theta^2)(\mu_2 + \Theta\mu_1) \\ &\quad + \Theta\widetilde{\epsilon}_n + \Theta^2\left\|\mathbb{T}_1 x_n - p\right\| + \Theta\,\epsilon_n. \end{aligned} \tag{27}$$

Passing to the limit in (27) and utilizing conditions (C2) and (C3), we get

$$\left\|\widetilde{p} - p\right\| \leq \frac{\left(\mu_2 + \Theta\mu_1\right)\left(\Theta^2 + \Theta + 1\right)}{1 - \Theta^2}.$$

This completes the proof. □

3. Numerical examples

In this section, we provide a range of intricate numerical examples that serve a dual purpose: reinforcing the theoretical conclusions drawn in the preceding section and highlighting the performance superiority of JNSI algorithm (version 2) in computational tasks compared to various algorithms, including its own classical counterpart.

Unless otherwise stated throughout this section and Section 4.1, we take

$$(\forall n \in \mathbb{N}_0) \qquad \alpha_n = \varphi_n = \psi_n = \beta_n = \frac{n+1}{n+2}, \quad \epsilon_n = \frac{1}{(n+1)^2}, \quad \theta_n = \frac{1}{(n+1)^{10}+1}\overline{\theta}_n,$$

and $\eta = 2$.

The following example demonstrates that the result in Theorem 2.1 might not hold true if the condition $\Theta := (\delta + L)/(1 - L) < 1$ is omitted from the main assumptions.

Example 3.1. Let $\ell_1 = \left\{\{\xi_n\}_{n=0}^{\infty} : \sum\limits_{n=0}^{\infty} |\xi_n| < \infty\right\}$, with the norm $\|\{\xi_n\}_n\| = \sum\limits_{n=0}^{\infty} |\xi_n|$ and $\mathbb{X}' = \mathbb{X} = \ell_1$. Define two mappings $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ by

$$\mathbb{T}_1\left(\{\xi_i\}_{i=0}^{\infty}\right) = \{\psi_i\}_{i=0}^{\infty}, \quad \psi_i = \begin{cases} \dfrac{\xi_i}{2}, & i = 0, \\ \dfrac{\xi_i}{3}, & i \geq 1, \end{cases}$$

and

$$\mathbb{T}_2\left(\{\xi_i\}_{i=0}^{\infty}\right) = \{\rho_i\}_{i=0}^{\infty}, \quad \rho_i = \begin{cases} \dfrac{\xi_i}{3}, & i = 0, \\ \dfrac{\xi_{i-1} + 2\xi_i}{6}, & i \geq 1. \end{cases}$$

It can be easily seen that $\mathbb{T}_1$ and $\mathbb{T}_2$ are well-defined and $\mathbb{T}_2$ is linear. Thus, $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{X}$ is a vector subspace. It is clear that $\mathbb{T}_1$ is linear and injective. Hence, $\mathbb{T}_1(\mathbb{X}') \subset \mathbb{X}$ is a convex subspace.

Now, we show that $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$, and therefore we take $\xi \in \mathbb{T}_2(\mathbb{X}')$. Then, there exists a $\{\xi_n^1\}_{n=0}^{\infty} \in \mathbb{X}'$ such that $\xi = \mathbb{T}_2\left(\{\xi_n^1\}_{n=0}^{\infty}\right)$. If we choose the sequence $\{\xi_n^2\}_{n=0}^{\infty}$ as follows

$$\xi_0^2 := \frac{2\xi_0^1}{3} \quad \text{and} \quad \xi_n^2 := \frac{\xi_{n-1}^1 + 2\xi_n^1}{2}, \quad \text{for all } \; n \geq 1,$$

then, $\left\{\xi_n^2\right\}_{n=0}^{\infty} \in \mathbb{X}'$ and $\xi = \mathbb{T}_1\left(\left\{\xi_n^2\right\}_{n=0}^{\infty}\right)$. Thus, $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$. For $x = \{0, 0, \ldots\} \in \mathbb{X}'$, we have $\mathbb{T}_1 x = \mathbb{T}_2 x$. Thus, $p = \{0, 0, \ldots\} \in \mathbb{X}$ is the coincidence value of the pair $(\mathbb{T}_1, \mathbb{T}_2)$. On the other hand, any $x = \{\xi_n\}_{n=0}^{\infty} \in \mathbb{X}'$, so we have

$$\left\|\mathbb{T}_2 x - p\right\| = \frac{|\xi_0|}{3} + \sum_{i=1}^{\infty} \left|\frac{\xi_{i-1} + 2\xi_i}{6}\right|, \tag{28}$$

$$\left\|\mathbb{T}_1 x - p\right\| = \frac{|\xi_0|}{2} + \sum_{i=1}^{\infty} \frac{|\xi_i|}{3} \quad \text{and} \quad \|\mathbb{T}_2 x - \mathbb{T}_1 x\| = \frac{1}{3}|\xi_0| + \sum_{i=1}^{\infty} \frac{|\xi_i|}{6}. \tag{29}$$

Utilizing (28) and (29), we get, for all $x = \{\xi_n\}_{n=0}^{\infty} \in \mathbb{X}'$,

$$\begin{aligned} \left\|\mathbb{T}_2 x - p\right\| &\leq \frac{|\xi_0|}{3} + \sum_{i=0}^{\infty} \frac{|\xi_i|}{6} + \sum_{i=0}^{\infty} \frac{|2\xi_{i+1}|}{6} \\ &= \frac{1}{2}\left(\frac{|\xi_0|}{3} + \frac{|\xi_1|}{3} + \frac{|\xi_2|}{3} + \cdots\right) + 2\left(\frac{|\xi_0|}{6} + \frac{|\xi_1|}{6} + \frac{|\xi_2|}{6} + \cdots\right) \\ &\leq \frac{1}{2}\left(\frac{|\xi_0|}{2} + \sum_{i=1}^{\infty} \frac{|\xi_i|}{3}\right) + 2\left(\frac{|\xi_0|}{3} + \sum_{i=1}^{\infty} \frac{|\xi_i|}{6}\right). \end{aligned}$$

Thus, for all $x \in \mathbb{X}'$,

$$\left\|\mathbb{T}_2 x - p\right\| \leq \delta \left\|\mathbb{T}_1 x - p\right\| + L\left\|\mathbb{T}_2 x - \mathbb{T}_1 x\right\|, \tag{30}$$

in which $\delta = 1/2$ and $L = 2$. So, the $(\mathbb{T}_1, \mathbb{T}_2)$ is a pair of quasi (L, δ)-contractive mappings. On the other hand, inequality (30) is not satisfied for any numbers $\delta \geq 0$ and $L \geq 0$ satisfying the condition $\delta + 2L < 1$. Indeed, let $x = \left\{1/2^{i+1}\right\}_{i=0}^{\infty}$. In this case, $x \in \ell_1$ and

$$\left\|\mathbb{T}_2 x - p\right\| = \frac{1}{2}, \quad \left\|\mathbb{T}_1 x - p\right\| = \frac{5}{12} \quad \text{and} \quad \left\|\mathbb{T}_2 x - \mathbb{T}_1 x\right\| = \frac{1}{4}.$$

Assume that inequality (30) is satisfied for any numbers $\delta \geq 0$ and $L \geq 0$ satisfying the condition $\delta + 2L < 1$. In this case, by (30), we have

$$\frac{1}{2} \leq \delta \frac{5}{12} + L \frac{1}{4}.$$

That is, $6 \leq 5\delta + 3L$. However, it cannot be $6 \leq 5\delta + 3L$ as $\delta + 2L < 1$. Thus, inequality (30) is not satisfied for $\delta + 2L < 1$.

Now, we take $x_{-1} = x_0 = 1/3^{n+1}$, for all $n \in \mathbb{N}_0$. Hence, all conditions except the condition $\delta + 2L < 1$ in Theorem 2.1 are satisfied. We see in Figure 1 that the sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (version 2) does not converge to p. That is, if the condition $\delta + 2L < 1$ is removed from the main assumptions, then convergence of $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ to point p may not be guaranteed.

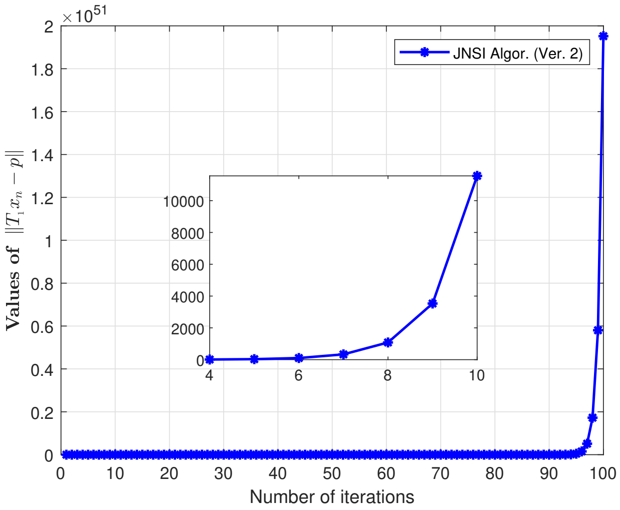


Figure 1: Graph denotes the convergence state of JNSI algorithm (version 2) for $n = 1, 2, \ldots, 100$

Now, we present an illustrative example that meets all the criteria outlined in Theorem 2.1, thereby corroborating the theorem's outcome.

Example 3.2. Let $\mathbb{X}' = \mathbb{X} = \ell_1$. Define two mappings by

$$\begin{aligned} \mathbb{T}_1 \ &: \ \mathbb{X}' \longrightarrow \mathbb{X} \\ &\quad \{\xi_n\}_{n=0}^{\infty} \longrightarrow \left\{ \frac{\xi_0}{2}, \frac{\xi_1}{2}, \frac{\xi_2}{2}, \ldots \right\} \end{aligned}$$

and

$$\begin{aligned} \mathbb{T}_2 \ &: \ \mathbb{X}' \longrightarrow \mathbb{X} \\ &\quad \{\xi_n\}_{n=0}^{\infty} \longrightarrow \left\{ \frac{\xi_0}{8}, \frac{\xi_1}{8}, \frac{\xi_2}{8}, \ldots \right\}. \end{aligned}$$

Following similar arguments as in Example 3.1, it can be easily shown that $\mathbb{T}_1$ and $\mathbb{T}_2$ are well-defined and the conditions (i)–(iii) of Remark 1.1 are satisfied. For $x = \{0, 0, \ldots\} \in \mathbb{X}'$, $p = \{0, 0, \ldots\} \in \mathbb{X}$ is the coincidence value of the pair $(\mathbb{T}_1, \mathbb{T}_2)$. On the other hand, for any $x = \{\xi_n\}_{n=0}^{\infty} \in \mathbb{X}'$, we have

$$\left\| \mathbb{T}_2 x - p \right\| = \frac{1}{8} \left\| \{\xi_n\}_{n=0}^{\infty} \right\|, \quad \left\| \mathbb{T}_1 x - p \right\| = \frac{1}{2} \left\| \{\xi_n\}_{n=0}^{\infty} \right\| \tag{31}$$

and

$$\|\mathbb{T}_2x-\mathbb{T}_1x\| = \frac{3}{8}\left\|\{\xi_n\}_{n=0}^{\infty}\right\|. \tag{32}$$

Utilizing (31) and (32), we get, for all $x=\{\xi_n\}_{n=0}^{\infty}\in\mathbb{X}'$

$$\begin{aligned}\left\|\mathbb{T}_2x-p\right\| &\leq \frac{9}{40}\left\|\{\xi_n\}_{n=0}^{\infty}\right\| \\ &= \frac{1}{5}\left(\frac{1}{2}\left\|\{\xi_n\}_{n=0}^{\infty}\right\|\right)+\frac{1}{3}\left(\frac{3}{8}\|\{\xi_n\}_{n=0}^{\infty}\|\right)\end{aligned}$$

that is, for all $x\in\mathbb{X}'$,

$$\left\|\mathbb{T}_2x-p\right\|\leq\delta\left\|\mathbb{T}_1x-p\right\|+L\|\mathbb{T}_2x-\mathbb{T}_1x\|,$$

in which $\delta=1/5$ and $L=1/3$. So, the $(\mathbb{T}_1,\mathbb{T}_2)$ is a pair of quasi (L,δ)-contractive mappings with $\delta+2L=13/15<1$. We take $x_{-1}=x_0=s_0=1/3^{n+1}$, for all $n\in\mathbb{N}$. Thus, all hypotheses in Theorem 2.1 are satisfied, and the sequence $\{\mathbb{T}_1x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) converges to p. We see this case and convergence state of some other algorithms in Figure 2.

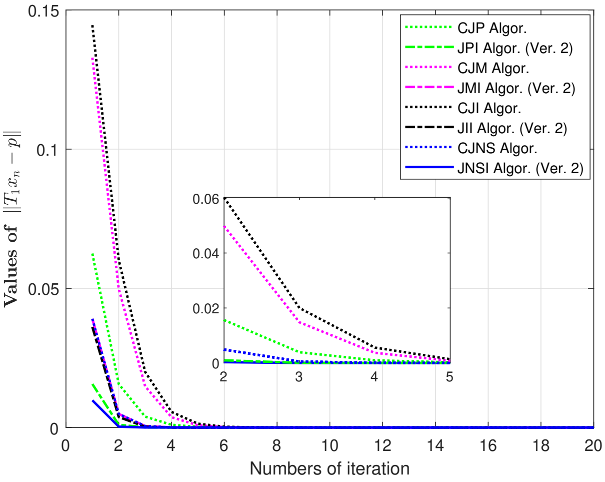


Figure 2: Graphs show the convergence states of algorithms for $n=1,2,\ldots,20$

The subsequent example demonstrates that JNSI algorithm (version 2) is weakly $(\mathbb{T}_1,\mathbb{T}_2)$-stable.

Example 3.3. Let $\mathbb{X}'$, $\mathbb{X}$, $\mathbb{T}_1$, $\mathbb{T}_2$, x_{-1} and x_0 be as in Example 3.2, and the sequence $\{\mathbb{T}_1x_n\}_{n=0}^{\infty}$ be generated by JNSI algorithm (ver. 2). We take the sequence $\{\mathbb{T}_1u_n\}_{n=0}^{\infty}$ in X as follows

$$(\forall n\in\mathbb{N})\qquad \mathbb{T}_1u_n=\left\{\mathbb{T}_1u_k^n\right\}_{k=0}^{\infty},\quad \mathbb{T}_1u_k^n=\begin{cases}0, & \text{if } k<n,\\ \dfrac{1}{(k+1)^4+\log(k+1)}, & \text{if } k\geq n.\end{cases}$$

The sequence $\{\mathbb{T}_1u_n\}_{n=0}^{\infty}$ is an approximate sequence of $\{\mathbb{T}_1x_n\}_{n=0}^{\infty}$ as can be seen in Figure 3 (a). We also see in Figure 3 (b) that $\lim_{n\to\infty}\mathbb{T}_1u_n=p$ if and only if $\lim_{n\to\infty}\varepsilon_n=0$. In particularly, JNSI algorithm (ver. 2) is weakly $(\mathbb{T}_1,\mathbb{T}_2)$-stable.

In the following example, we illustrate the data dependency between the values p and $\widetilde{p}$ in Theorem 2.5. The example shows the practical application of the estimate mentioned in (23).

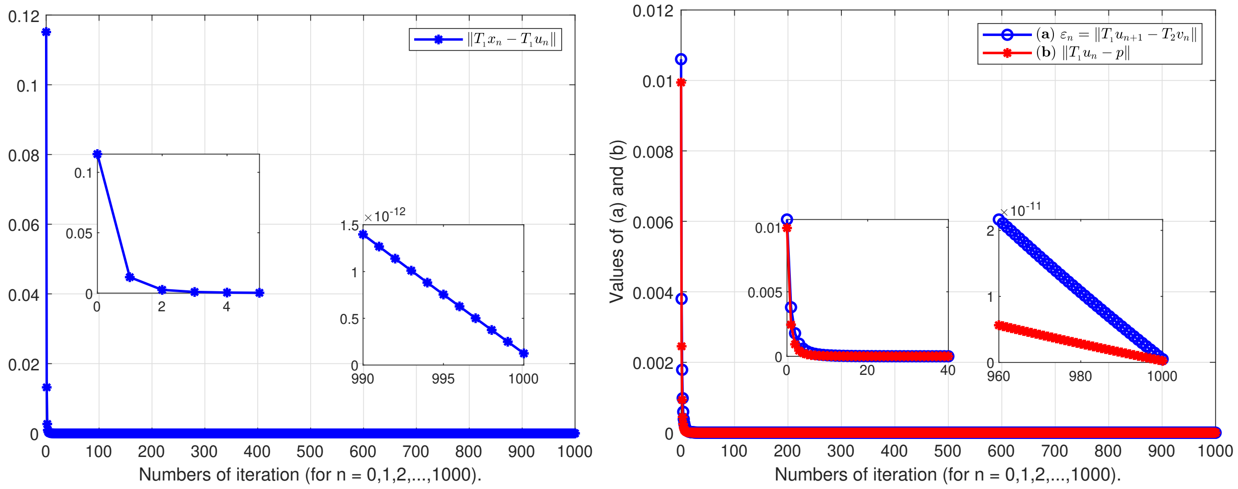


Figure 3: Graphs show that values of $\|T_1x_n - T_1u_n\|$, ε_n generated by (13) and $\|T_1u_n - p\|$

Example 3.4. Let $\mathbb{X}'$, $\mathbb{X}$, $\mathbb{T}_1$, $\mathbb{T}_2$, x_{-1} and x_0 be as in Example 3.2. Define the following mappings

$$\begin{aligned}\widetilde{\mathbb{T}}_1 \quad &: \quad \mathbb{X}' \to \mathbb{X}\\ &\quad \{\xi_n\}_{n=0}^{\infty} \to \widetilde{\mathbb{T}}_1\left(\{\xi_n\}_{n=0}^{\infty}\right) = \left\{\frac{\xi_n}{2} + \frac{1}{10^{n+1}}\right\}_{n=0}^{\infty}\end{aligned}$$

and

$$\begin{aligned}\widetilde{\mathbb{T}}_2 \quad &: \quad \mathbb{X}' \to \mathbb{X}\\ &\quad \{\xi_n\}_{n=0}^{\infty} \to \widetilde{\mathbb{T}}_2\left(\{\xi_n\}_{n=0}^{\infty}\right) = \left\{\frac{\xi_n}{8} + \frac{1}{10^{n+1}}\right\}_{n=0}^{\infty}.\end{aligned}$$

It can be easily seen as in Example 3.2 that $\widetilde{\mathbb{T}}_1$ is injective, $\widetilde{\mathbb{T}}_1(\mathbb{X}')$ is a convex set and $\widetilde{\mathbb{T}}_2(\mathbb{X}') \subset \widetilde{\mathbb{T}}_1(\mathbb{X}')$ and $(1+\lambda)\widetilde{\mathbb{T}}_2(\mathbb{X}') - \lambda\widetilde{\mathbb{T}}_2(\mathbb{X}') \subset \widetilde{\mathbb{T}}_2(\mathbb{X}')$.

For $x = \{0, 0, \ldots\}$, $\widetilde{\mathbb{T}}_1 x = \widetilde{\mathbb{T}}_2 x = \left\{1/10^{n+1}\right\}_{n=0}^{\infty} = \widetilde{p}$. Thus, $\widetilde{p}$ is the unique coincidence value of $(\mathbb{T}_1, \mathbb{T}_2)$. An easy computation yields that

$$\|\mathbb{T}_1 x - \widetilde{\mathbb{T}}_1 x\| = \|\mathbb{T}_2 x - \widetilde{\mathbb{T}}_2 x\| = \sum_{i=0}^{\infty} \frac{1}{10^{i+1}} = \frac{1}{9} = \mu_1 = \mu_2,$$

for all $x = \{\xi_n\}_{n=0}^{\infty} \in \mathbb{X}'$.

The sequence $\{\widetilde{\mathbb{T}}_1 \widetilde{x}_n\}_{n=1}^{\infty}$ generated by (21) with $\tilde{x}_{-1} = \tilde{x}_0 = 1/3^{n+1}$, $\tilde{\varepsilon}_n = 1/(n+1)^2$, for all $n \in \mathbb{N}$, converges to $\widetilde{p}$ as we can see in Figure 4. Now, by (23), we have

$$\left\|p - \widetilde{p}\right\| = \frac{1}{9} \leq \frac{\left(\frac{1}{9} + \frac{4}{5}\cdot\frac{1}{9}\right)\left(\frac{16}{25} + \frac{4}{5} + 1\right)}{1 - \frac{16}{25}} = \frac{61}{45}.$$

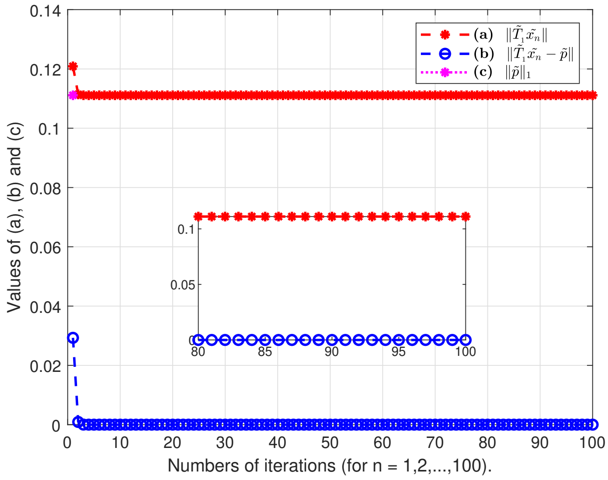


Figure 4: Graphs denote that values of the $\|\widetilde{T}_1\widetilde{x}_n\|$, $\|\widetilde{T}_1\widetilde{x}_n - \tilde{p}\|$ and $\|\tilde{p}\|$

4. Applications

In this section, practical applications of the results obtained in this study are provided as examples, not only by testing the hypotheses but also by providing examples of how the iteration algorithms defined in this study can be useful in solving problems directly without testing the hypotheses.

4.1. *Applications to integral and differential equations*

Example 4.1. Let us contemplate the integral equation provided as

$$2(t+1)x(t) + t^2 = \int_0^t (1-s^2)x(s)\,\mathrm{d}s, \tag{33}$$

which has a solution

$$z(t) = -\frac{\mathrm{e}^{-(t-1)^2/4}}{t+1}\left\{\sqrt{\pi}\left[\operatorname{erfi}\left(\frac{1}{2}\right) + \operatorname{erfi}\left(\frac{t-1}{2}\right)\right] + 2\mathrm{e}^{(t-1)^2/4} - 2\sqrt[4]{\mathrm{e}}\right\},$$

where

$$\operatorname{erfi}(z) = -\mathrm{i}\operatorname{erf}(\mathrm{i}z) \quad \text{and} \quad \operatorname{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z \mathrm{e}^{-t^2}\,\mathrm{d}t.$$

It is a well-known fact that the set of real-valued continuous functions on the closed interval $[0,1]$, denoted as $C[0,1]$, is a Banach space with the norm $\|x\| = \max_{t\in[0,1]} |x(t)|$. Let $\mathbb{X} = \mathbb{X}' = (C[0,1], \|\cdot\|)$. If we define the operators $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ as

$$\mathbb{T}_1(x) = (t+1)\,x(t) + \frac{t^2}{2}, \quad \mathbb{T}_2(x) = \frac{1}{2}\int_0^t (1-s^2)x(s)\,\mathrm{d}s$$

subsequently, we can satisfy all the conditions stated in Theorem 2.1. In fact,

i) Since $\mathbb{T}_1$ is bijective on $\mathbb{X}'$, we have $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$.

ii) Since $\mathbb{T}_2$ is linear, $\mathbb{T}_2(\mathbb{X}')$ is a subspace in $\mathbb{X}$.

iii) It is readily apparent that $\mathbb{T}_1(\mathbb{X}')$ forms a closed and convex subset of $\mathbb{X}$.

iv) Let $\mathbb{T}_2(z(t)) = \mathbb{T}_1(z(t)) = p(t)$. Since $1 - s^2 \leq 1 + s$ for $s \geq 0$ then, for any $L > 0$ and $\delta = 1/2$, we have

$$\left\|\mathbb{T}_2 x - p\right\| \leq \frac{1}{2}\left\|\mathbb{T}_1 x - p\right\| + L\left\|\mathbb{T}_1 x - \mathbb{T}_2 x\right\|.$$

Let choose $\mathbb{T}_2(x_{-1}(t)) = \mathbb{T}_2(1)$ and $\mathbb{T}_2(x_0(t)) = \mathbb{T}_2(1.001)$. Then, by Theorem 2.1, the sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (version 2) converges to $p(t) = \mathbb{T}_2(z(t)) = \mathbb{T}_1(z(t))$. Moreover, the sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) converges faster than algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2), CJI, JII (ver. 2), and CJNS. This can be easily verified in Tables 1, 2, 3 and Figure 5.

Table 1: The errors $\left|\mathbb{T}_1(x_1) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	5.01×10^{-1}	0.	5.01×10^{-1}	0.	0.	0.
0.1	5.00×10^{-2}	1.17×10^{-3}	5.78×10^{-1}	2.56×10^{-2}	5.66×10^{-1}	2.53×10^{-2}	2.56×10^{-2}	5.93×10^{-4}
0.2	9.93×10^{-2}	4.32×10^{-3}	6.61×10^{-1}	5.18×10^{-2}	6.37×10^{-1}	5.08×10^{-2}	5.18×10^{-2}	2.22×10^{-3}
0.3	1.47×10^{-1}	8.93×10^{-3}	7.48×10^{-1}	7.82×10^{-2}	7.13×10^{-1}	7.60×10^{-2}	7.82×10^{-2}	4.65×10^{-3}
0.4	1.93×10^{-1}	1.45×10^{-2}	8.39×10^{-1}	1.04×10^{-1}	7.95×10^{-1}	1.01×10^{-1}	1.04×10^{-1}	7.60×10^{-3}
0.5	2.36×10^{-1}	2.04×10^{-2}	9.35×10^{-1}	1.28×10^{-1}	8.81×10^{-1}	1.23×10^{-1}	1.28×10^{-1}	1.08×10^{-2}
0.6	2.75×10^{-1}	2.61×10^{-2}	1.03	1.50×10^{-1}	9.71×10^{-1}	1.44×10^{-1}	1.50×10^{-1}	1.39×10^{-2}
0.7	3.07×10^{-1}	3.12×10^{-2}	1.13	1.69×10^{-1}	1.07	1.62×10^{-1}	1.69×10^{-1}	1.67×10^{-2}
0.8	3.33×10^{-1}	3.52×10^{-2}	1.24	1.84×10^{-1}	1.16	1.76×10^{-1}	1.84×10^{-1}	1.89×10^{-2}
0.9	3.50×10^{-1}	3.77×10^{-2}	1.34	1.94×10^{-1}	1.26	1.85×10^{-1}	1.94×10^{-1}	2.03×10^{-2}
1.0	3.56×10^{-1}	3.86×10^{-2}	1.44	1.97×10^{-1}	1.36	1.88×10^{-1}	1.97×10^{-1}	2.08×10^{-2}

Table 2: The errors $\left|\mathbb{T}_1(x_3) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	4.17×10^{-2}	0.	4.17×10^{-2}	0.	0.	0.
0.1	1.83×10^{-5}	5.60×10^{-11}	5.91×10^{-2}	7.83×10^{-7}	5.03×10^{-2}	7.48×10^{-7}	8.18×10^{-7}	5.68×10^{-12}
0.2	1.28×10^{-4}	1.89×10^{-10}	7.78×10^{-2}	5.91×10^{-6}	5.98×10^{-2}	5.43×10^{-6}	6.05×10^{-6}	5.31×10^{-11}
0.3	3.70×10^{-4}	2.32×10^{-9}	9.72×10^{-2}	1.82×10^{-5}	7.00×10^{-2}	1.62×10^{-5}	1.85×10^{-5}	8.22×10^{-11}
0.4	7.47×10^{-4}	1.47×10^{-8}	1.17×10^{-1}	3.84×10^{-5}	8.07×10^{-2}	3.32×10^{-5}	3.89×10^{-5}	2.06×10^{-10}
0.5	1.22×10^{-3}	4.53×10^{-8}	1.37×10^{-1}	6.52×10^{-5}	9.17×10^{-2}	5.53×10^{-5}	6.59×10^{-5}	1.23×10^{-9}
0.6	1.74×10^{-3}	9.76×10^{-8}	1.56×10^{-1}	9.57×10^{-5}	1.03×10^{-1}	7.98×10^{-5}	9.67×10^{-5}	3.25×10^{-9}
0.7	2.24×10^{-3}	1.66×10^{-7}	1.73×10^{-1}	1.26×10^{-4}	1.14×10^{-1}	1.04×10^{-4}	1.27×10^{-4}	6.09×10^{-9}
0.8	2.66×10^{-3}	2.35×10^{-7}	1.89×10^{-1}	1.51×10^{-4}	1.24×10^{-1}	1.24×10^{-4}	1.53×10^{-4}	9.10×10^{-9}
0.9	2.93×10^{-3}	2.87×10^{-7}	2.02×10^{-1}	1.69×10^{-4}	1.34×10^{-1}	1.37×10^{-4}	1.70×10^{-4}	1.14×10^{-8}
1.0	3.03×10^{-3}	3.07×10^{-7}	2.13×10^{-1}	1.75×10^{-4}	1.43×10^{-1}	1.42×10^{-4}	1.76×10^{-4}	1.23×10^{-8}

It is obvious that the newly defined inertial step increases the accuracy, as well as the convergence speeds of Jungck type algorithms. However, the classical inertial step has no affect on convergence speeds of Junck type algorithms (see Table 4 for errors in the iterations of the algorithms CJNS and JNSI (ver. 1) for $n = 1, 2, 3, 4$).

Table 3: The errors $\left|\mathbb{T}_1(x_5) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	1.39×10^{-3}	0.	1.39×10^{-3}	0.	0.	0.
0.1	2.05×10^{-9}	1.98×10^{-19}	2.72×10^{-3}	2.97×10^{-12}	1.81×10^{-3}	2.72×10^{-12}	3.20×10^{-12}	5.93×10^{-22}
0.2	5.08×10^{-8}	2.65×10^{-17}	4.32×10^{-3}	8.42×10^{-11}	2.29×10^{-3}	7.17×10^{-11}	8.76×10^{-11}	1.00×10^{-19}
0.3	2.94×10^{-7}	2.75×10^{-16}	6.12×10^{-3}	5.36×10^{-10}	2.82×10^{-3}	4.30×10^{-10}	5.51×10^{-10}	1.49×10^{-18}
0.4	9.26×10^{-7}	5.68×10^{-16}	8.04×10^{-3}	1.82×10^{-9}	3.39×10^{-3}	1.39×10^{-9}	1.87×10^{-9}	7.79×10^{-18}
0.5	2.07×10^{-6}	2.12×10^{-15}	1.00×10^{-2}	4.34×10^{-9}	3.96×10^{-3}	3.17×10^{-9}	4.42×10^{-9}	2.17×10^{-17}
0.6	3.68×10^{-6}	1.49×10^{-14}	1.19×10^{-2}	8.10×10^{-9}	4.54×10^{-3}	5.74×10^{-9}	8.24×10^{-9}	3.85×10^{-17}
0.7	5.53×10^{-6}	4.42×10^{-14}	1.36×10^{-2}	1.26×10^{-8}	5.08×10^{-3}	8.73×10^{-9}	1.28×10^{-8}	4.63×10^{-17}
0.8	7.27×10^{-6}	8.66×10^{-14}	1.50×10^{-2}	1.70×10^{-8}	5.57×10^{-3}	1.16×10^{-8}	1.73×10^{-8}	3.81×10^{-17}
0.9	8.50×10^{-6}	1.26×10^{-13}	1.60×10^{-2}	2.03×10^{-8}	5.98×10^{-3}	1.36×10^{-8}	2.06×10^{-8}	2.15×10^{-17}
1.0	8.95×10^{-6}	1.42×10^{-13}	1.65×10^{-2}	2.14×10^{-8}	6.31×10^{-3}	1.44×10^{-8}	2.17×10^{-8}	1.31×10^{-17}

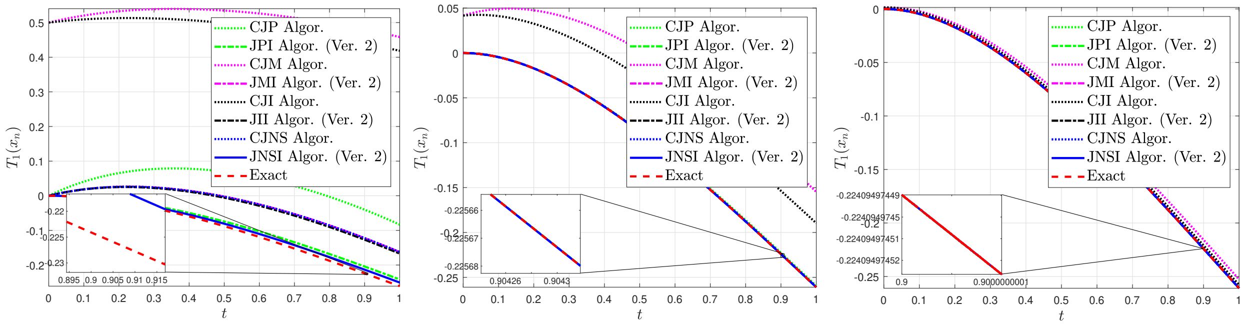


Figure 5: Comparisons of the convergence of JNSI algorithm (ver. 2) with others, for $n = 1, 3,$ and 5

Table 4: The errors $\left|\mathbb{T}_1(x_1) - p(t)\right|$ in the iterations of the algorithms CJNS and JNSI (ver. 1) for $n = 1, 2, 3, 4$

Iter.	n = 1		n = 2		n = 3		n = 4	
t	CJNS	JNSI (Ver.1)	CJNS	JNSI (Ver.1)	CJNS	JNSI (Ver.1)	CJNS	JNSI (Ver.1)
0.0	0	0	0	0	0	0	0	0
0.1	2.56×10^{-2}	2.56×10^{-2}	2.04×10^{-4}	2.02×10^{-4}	8.18×10^{-7}	8.06×10^{-7}	1.98×10^{-9}	1.94×10^{-9}
0.2	5.18×10^{-2}	5.18×10^{-2}	7.85×10^{-4}	7.81×10^{-4}	6.05×10^{-6}	6.00×10^{-6}	2.81×10^{-8}	2.78×10^{-8}
0.3	7.82×10^{-2}	7.82×10^{-2}	1.68×10^{-3}	1.67×10^{-3}	1.85×10^{-5}	1.84×10^{-5}	1.23×10^{-7}	1.22×10^{-7}
0.4	1.04×10^{-1}	1.04×10^{-1}	2.79×10^{-3}	2.79×10^{-3}	3.89×10^{-5}	3.87×10^{-5}	3.29×10^{-7}	3.27×10^{-7}
0.5	1.28×10^{-1}	1.28×10^{-1}	4.02×10^{-3}	4.01×10^{-3}	6.59×10^{-5}	6.57×10^{-5}	6.59×10^{-7}	6.55×10^{-7}
0.6	1.50×10^{-1}	1.50×10^{-1}	5.25×10^{-3}	5.24×10^{-3}	9.67×10^{-5}	9.63×10^{-5}	1.09×10^{-6}	1.08×10^{-6}
0.7	1.69×10^{-1}	1.69×10^{-1}	6.36×10^{-3}	6.35×10^{-3}	1.27×10^{-4}	1.27×10^{-4}	1.56×10^{-6}	1.55×10^{-6}
0.8	1.84×10^{-1}	1.84×10^{-1}	7.25×10^{-3}	7.23×10^{-3}	1.53×10^{-4}	1.52×10^{-4}	1.98×10^{-6}	1.97×10^{-6}
0.9	1.94×10^{-1}	1.94×10^{-1}	7.82×10^{-3}	7.80×10^{-3}	1.70×10^{-4}	1.70×10^{-4}	2.28×10^{-6}	2.27×10^{-6}
1.0	1.97×10^{-1}	1.97×10^{-1}	8.01×10^{-3}	8.00×10^{-3}	1.76×10^{-4}	1.76×10^{-4}	2.38×10^{-6}	2.37×10^{-6}

Since we have confirmed the validity of Theorems 2.1, we can now verify Theorems 2.4 and 2.5. Let us begin with Theorem 2.4.

Let $\{u_n\}_{n=0}^{\infty}$ be $\left\{z(t)\left(\frac{n^{10}+1}{n^{10}+2}\right)\right\}_{n=0}^{\infty}$ and $\{\varepsilon_n\}_{n=0}^{\infty}$ be the sequence generated by (13). Then, Table 5 indicates that $\lim\limits_{n\to\infty} \varepsilon_n = 0$ and it is obvious that the sequence $\{\mathbb{T}_1 u_n\}_{n=1}^{\infty}$ converges to $\mathbb{T}_2\left(z\left(t\right)\right) = \mathbb{T}_1\left(z\left(t\right)\right) = p\left(t\right)$. Hence, JNSI algorithm (ver. 2) is weakly $(\mathbb{T}_1, \mathbb{T}_2)$-stable.

Table 5: The errors $\varepsilon_n = \|\mathbb{T}_1 u_{n+1} - \mathbb{T}_2 v_n\|$

n	1	2	3	4	5	10	20
ε_n	3.77×10^{-4}	8.54×10^{-6}	4.93×10^{-7}	5.32×10^{-8}	8.61×10^{-9}	5.22×10^{-11}	3.13×10^{-14}

In conclusion, let us confirm the validity of Theorem 2.5 by examining the following integral equation

$$2(t+1)x(t) + t^2 + \tau = \int_0^t \left[\left(1 - s^2\right)x(s) + \tau\right] \mathrm{d}s \tag{34}$$

where τ is a sufficiently small real number. The solution to equation (34) is given by

$$\widetilde{z}(t) = \frac{\mathrm{e}^{-(t-1)^2/4}}{2(t+1)} \left\{ \sqrt{\pi}\,(\tau - 2)\left[\operatorname{erfi}\left(\frac{t-1}{2}\right) + \operatorname{erfi}\left(\frac{1}{2}\right)\right] - \sqrt[4]{\mathrm{e}}(\tau - 4) - 4\mathrm{e}^{(t-1)^2/4} \right\}.$$

Define the following operators

$$\widetilde{\mathbb{T}}_1(x) = (t+1)x(t) + \frac{t^2 + \tau}{2}, \quad \widetilde{\mathbb{T}}_2(x) = \frac{1}{2}\int_0^t \left[\left(1 - s^2\right)x(s) + \tau\right] \mathrm{d}s$$

then, we have

$$\left\|\mathbb{T}_1(x) - \widetilde{\mathbb{T}}_1(x)\right\| = \max_{t \in [0,1]} \left|\frac{\tau}{2}\right| = \frac{|\tau|}{2} = \mu_1, \quad \text{for all } x \in \mathbb{X}',$$

and

$$\left\|\mathbb{T}_2(x) - \widetilde{\mathbb{T}}_2(x)\right\| = \max_{t \in [0,1]} \left|-\frac{\tau t}{2}\right| = \frac{|\tau|}{2} = \mu_2, \quad \text{for all } x \in \mathbb{X}'.$$

Assuming τ taken as 10^{-6}, Table 6 illustrates that the sequence $\left\{\widetilde{\mathbb{T}}_1 \widetilde{x}_n\right\}_{n=1}^{\infty}$, generated by (21), with $\widetilde{\mathbb{T}}_2(\widetilde{x}_{-1}(t)) = \widetilde{\mathbb{T}}_2(-\tau/2)$ and $\widetilde{\mathbb{T}}_2(\widetilde{x}_0(t)) = \widetilde{\mathbb{T}}_2(-\tau/2 + 0.001)$, converges to $\widetilde{p}(t) = \widetilde{\mathbb{T}}_2(\widetilde{z}(t)) = \widetilde{\mathbb{T}}_1(\widetilde{z}(t))$. Hence,

$$\left\|p(t) - \widetilde{p}(t)\right\| = \frac{\sqrt{\pi}\operatorname{erfi}\left(\frac{1}{2}\right) + 1 - \sqrt[4]{e}}{2000000} = 4.02974395839751\ldots \times 10^{-7}.$$

On the other hand, without knowing the value of $\widetilde{p}(t)$ (or without computing it), by (23), we have the following

$$\left\|\widetilde{p}(t) - p(t)\right\| = 4.03 \times 10^{-7} < 1.06 \times 10^{-5} = \frac{\left(\mu_2 + \left(\frac{\delta + L}{1 - L}\right)\mu_1\right)\left(\left(\frac{\delta + L}{1 - L}\right)^2 + \left(\frac{\delta + L}{1 - L}\right) + 1\right)}{1 - \left(\frac{\delta + L}{1 - L}\right)^2}.$$

Remark 4.1. For solving the integral equation (33) (or its perturbed form (34)), we can use discrete versions of the operators $\mathbb{T}_1$ and $\mathbb{T}_2$, with high precision methods presented in [25].

Table 6: The errors $\left|\widetilde{\mathbb{T}_1}(\widetilde{x}_n) - \widetilde{p}(t)\right|$ for $n = 1, 2, \dots, 6$

Iter.	t=0	t=0.1	t=0.2	t=0.3	t=0.4	t=0.5	t=0.6	t=0.7	t=0.8	t=0.9	t=1
n=1	0	1.05×10^{-6}	8.60×10^{-6}	3.23×10^{-5}	8.12×10^{-5}	1.59×10^{-4}	2.62×10^{-4}	3.76×10^{-4}	4.84×10^{-4}	5.61×10^{-4}	5.89×10^{-4}
n=2	0	1.82×10^{-10}	8.42×10^{-10}	2.76×10^{-10}	7.10×10^{-9}	2.88×10^{-8}	6.92×10^{-8}	1.25×10^{-7}	1.84×10^{-7}	2.29×10^{-7}	2.46×10^{-7}
n=3	0	7.51×10^{-15}	1.30×10^{-13}	5.76×10^{-13}	1.04×10^{-12}	3.31×10^{-14}	4.97×10^{-12}	1.49×10^{-11}	2.79×10^{-11}	3.92×10^{-11}	4.37×10^{-11}
n=4	0	1.03×10^{-19}	6.17×10^{-18}	5.99×10^{-17}	2.48×10^{-16}	5.87×10^{-16}	8.66×10^{-16}	7.45×10^{-16}	1.09×10^{-16}	6.86×10^{-16}	1.05×10^{-15}
n=5	0	6.51×10^{-25}	1.38×10^{-22}	2.70×10^{-21}	1.88×10^{-20}	7.09×10^{-20}	1.74×10^{-19}	3.12×10^{-19}	4.41×10^{-19}	5.24×10^{-19}	5.51×10^{-19}
n=6	0	2.21×10^{-30}	1.70×10^{-27}	6.65×10^{-26}	7.42×10^{-25}	3.99×10^{-24}	1.31×10^{-23}	2.98×10^{-23}	5.11×10^{-23}	6.91×10^{-23}	7.62×10^{-23}

Example 4.2. We are tasked with solving the following second-order differential equation (DE) subject to homogeneous Dirichlet boundary conditions:

$$\left.\begin{aligned} x''(t) &= x(t) + t^3 - 2t^2 + 2t - 7, \quad t \in [0,1], \\ x(0) &= x(1) = 0. \end{aligned}\right\} \tag{35}$$

To proceed with solving equation (35), we require an appropriate function space. Let $\mathbb{X} = C[0,1]$. Additionally, let $\mathbb{X}'$ be the subset of $\mathbb{X}$ defined as $\mathbb{X}' := \left\{x(t) \in C^2[0,1] : x(0) = x(1) = 0\right\}$ with norm $\|x\|_c$, in which $C^2[0,1]$ is the space of real functions with continuous second derivatives on $[0,1]$ and $\|x\|_c = \max\left\{\|x\|, \|x'\|, \|x''\|\right\}$. It has been established in [7] that for all $x \in \mathbb{X}'$, the inequalities

$$\|x\| \leq \frac{1}{2}\|x'\| \leq \frac{1}{2}\|x''\|$$

hold. Define operators $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ as

$$\mathbb{T}_1(x) = x'', \quad \mathbb{T}_2(x) = x(t) + t^3 - 2t^2 + 2t - 7.$$

It is worth noting that $\mathbb{T}_1$ is an injective map and the inverse of $\mathbb{T}_1$ can be expressed in the form

$$\mathbb{T}_1^{-1}(x) = \int_0^1 G(t,s)\,x(s)\,\mathrm{d}s,$$

where $G(t,s)$ is the Green's function corresponding to the boundary value problem. Thus, in this case,

$$G(t,s) = \begin{cases} s(t-1), & 0 \leq s \leq t, \\ t(s-1), & t < s \leq 1, \end{cases}$$

and

$$\mathbb{T}_1^{-1}(x) = (t-1)\int_0^t s\,x(s)\,\mathrm{d}s + t\int_t^1 (s-1)x(s)\,\mathrm{d}s.$$

Also, the exact solution is

$$z(t) = -t^3 + 2t^2 - 8t + \frac{(4-11\mathrm{e})\mathrm{e}^{1-t} + (11-4\mathrm{e})\mathrm{e}^t}{\mathrm{e}^2 - 1} + 11,$$

as well as

$$p(z) = \mathbb{T}_1(z) = \mathbb{T}_2(z) = \frac{(4-11\mathrm{e})\mathrm{e}^{1-t} + (11-4\mathrm{e})\mathrm{e}^t}{\mathrm{e}^2 - 1} - 6t + 4.$$

Furthermore, it can be established that $\mathbb{T}_2(\mathbb{X}') \subset \mathbb{T}_1(\mathbb{X}')$ (for more details, refer to [7]).

Given the inequality

$$\left|\mathbb{T}_2\left(x\right) - \mathbb{T}_2\left(y\right)\right| \leq \left|x - y\right|,$$

we can apply [7, Lemma 4.4] to deduce

$$\left\|\mathbb{T}_2(x) - \mathbb{T}_2(y)\right\| \leq \frac{1}{2}\left\|\mathbb{T}_1(x) - \mathbb{T}_1(y)\right\|.$$

Consequently, all the assumptions of Theorem 2.1 are met. Therefore, by selecting $\mathbb{T}_2\left(x_{-1}(t)\right) = \mathbb{T}_2(t(1-t))$ and $\mathbb{T}_2\left(x_0(t)\right) = \mathbb{T}_2\Big(t(1-t^2)\Big)$, the sequence $\{\mathbb{T}_1\left(x_n\right)\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) not only converges to $\mathbb{T}_2\left(z\left(t\right)\right) = \mathbb{T}_1\left(z\left(t\right)\right) = p\left(t\right)$, but also demonstrates a significantly higher convergence rate when compared to the algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2), CJI, JII (ver. 2) and CJNS. This comparison is detailed in Tables 7, 8, 9, as well as in Figure 6.

Table 7: The errors $\left|\mathbb{T}_1(x_1) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	3.50	0.	3.50	0.	0.	0.
0.1	1.71×10^{-1}	1.24×10^{-2}	2.89	7.92×10^{-2}	2.94	8.26×10^{-2}	7.92×10^{-2}	5.61×10^{-3}
0.2	2.82×10^{-1}	2.31×10^{-2}	2.36	1.29×10^{-1}	2.43	1.36×10^{-1}	1.29×10^{-1}	1.04×10^{-2}
0.3	3.43×10^{-1}	3.11×10^{-2}	1.90	1.56×10^{-1}	1.99	1.65×10^{-1}	1.56×10^{-1}	1.40×10^{-2}
0.4	3.63×10^{-1}	3.56×10^{-2}	1.50	1.64×10^{-1}	1.60	1.73×10^{-1}	1.64×10^{-1}	1.60×10^{-2}
0.5	3.49×10^{-1}	3.66×10^{-2}	1.15	1.56×10^{-1}	1.25	1.66×10^{-1}	1.56×10^{-1}	1.64×10^{-2}
0.6	3.08×10^{-1}	3.40×10^{-2}	8.52×10^{-1}	1.37×10^{-1}	9.37×10^{-1}	1.47×10^{-1}	1.37×10^{-1}	1.53×10^{-2}
0.7	2.48×10^{-1}	2.85×10^{-2}	5.92×10^{-1}	1.10×10^{-1}	6.61×10^{-1}	1.17×10^{-1}	1.10×10^{-1}	1.27×10^{-2}
0.8	1.73×10^{-1}	2.04×10^{-2}	3.67×10^{-1}	7.61×10^{-2}	4.16×10^{-1}	8.18×10^{-2}	7.61×10^{-2}	9.13×10^{-3}
0.9	8.85×10^{-2}	1.06×10^{-2}	1.71×10^{-1}	3.89×10^{-2}	1.96×10^{-1}	4.19×10^{-2}	3.89×10^{-2}	4.76×10^{-3}
1.0	0.	0.	0.	0.	0.	0.	0.	0.

Table 8: The errors $\left|\mathbb{T}_1(x_3) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	2.92×10^{-1}	0.	2.92×10^{-1}	0.	0.	0.
0.1	1.17×10^{-3}	1.16×10^{-6}	2.11×10^{-1}	2.49×10^{-5}	2.37×10^{-1}	4.22×10^{-5}	2.47×10^{-5}	2.30×10^{-8}
0.2	2.22×10^{-3}	2.20×10^{-6}	1.48×10^{-1}	4.71×10^{-5}	1.90×10^{-1}	8.00×10^{-5}	4.67×10^{-5}	4.38×10^{-8}
0.3	3.04×10^{-3}	3.03×10^{-6}	1.00×10^{-1}	6.43×10^{-5}	1.51×10^{-1}	1.09×10^{-4}	6.38×10^{-5}	6.03×10^{-8}
0.4	3.56×10^{-3}	3.56×10^{-6}	6.46×10^{-2}	7.48×10^{-5}	1.18×10^{-1}	1.28×10^{-4}	7.42×10^{-5}	7.09×10^{-8}
0.5	3.72×10^{-3}	3.75×10^{-6}	3.91×10^{-2}	7.79×10^{-5}	8.99×10^{-2}	1.34×10^{-4}	7.73×10^{-5}	7.46×10^{-8}
0.6	3.52×10^{-3}	3.56×10^{-6}	2.15×10^{-2}	7.33×10^{-5}	6.60×10^{-2}	1.26×10^{-4}	7.28×10^{-5}	7.09×10^{-8}
0.7	2.98×10^{-3}	3.03×10^{-6}	1.01×10^{-2}	6.19×10^{-5}	4.56×10^{-2}	1.07×10^{-4}	6.14×10^{-5}	6.03×10^{-8}
0.8	2.16×10^{-3}	2.20×10^{-6}	3.46×10^{-3}	4.47×10^{-5}	2.81×10^{-2}	7.72×10^{-5}	4.43×10^{-5}	4.38×10^{-8}
0.9	1.13×10^{-3}	1.16×10^{-6}	4.51×10^{-4}	2.34×10^{-5}	1.30×10^{-2}	4.05×10^{-5}	2.32×10^{-5}	2.30×10^{-8}
1.0	0.	0.	0.	0.	0.	0.	0.	0.

Table 9: The errors $\left|\mathbb{T}_1(x_5) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	9.72×10^{-3}	0.	9.72×10^{-3}	0.	0.	0.
0.1	1.18×10^{-5}	1.22×10^{-10}	5.84×10^{-3}	2.43×10^{-9}	7.66×10^{-3}	1.29×10^{-8}	2.41×10^{-9}	2.37×10^{-14}
0.2	2.25×10^{-5}	2.32×10^{-10}	3.18×10^{-3}	4.62×10^{-9}	5.99×10^{-3}	2.45×10^{-8}	4.58×10^{-9}	4.52×10^{-14}
0.3	3.09×10^{-5}	3.19×10^{-10}	1.42×10^{-3}	6.35×10^{-9}	4.65×10^{-3}	3.38×10^{-8}	6.30×10^{-9}	6.21×10^{-14}
0.4	3.63×10^{-5}	3.75×10^{-10}	3.27×10^{-4}	7.46×10^{-9}	3.55×10^{-3}	3.97×10^{-8}	7.40×10^{-9}	7.31×10^{-14}
0.5	3.82×10^{-5}	3.95×10^{-10}	2.95×10^{-4}	7.83×10^{-9}	2.65×10^{-3}	4.17×10^{-8}	7.77×10^{-9}	7.68×10^{-14}
0.6	3.63×10^{-5}	3.75×10^{-10}	5.80×10^{-4}	7.43×10^{-9}	1.92×10^{-3}	3.96×10^{-8}	7.37×10^{-9}	7.31×10^{-14}
0.7	3.09×10^{-5}	3.19×10^{-10}	6.31×10^{-4}	6.31×10^{-9}	1.30×10^{-3}	3.37×10^{-8}	6.26×10^{-9}	6.21×10^{-14}
0.8	2.24×10^{-5}	2.32×10^{-10}	5.20×10^{-4}	4.58×10^{-9}	7.92×10^{-4}	2.45×10^{-8}	4.54×10^{-9}	4.52×10^{-14}
0.9	1.18×10^{-5}	1.22×10^{-10}	2.99×10^{-4}	2.41×10^{-9}	3.63×10^{-4}	1.29×10^{-8}	2.39×10^{-9}	2.37×10^{-14}
1.0	0.	0.	0.	0.	0.	0.	0.	0.

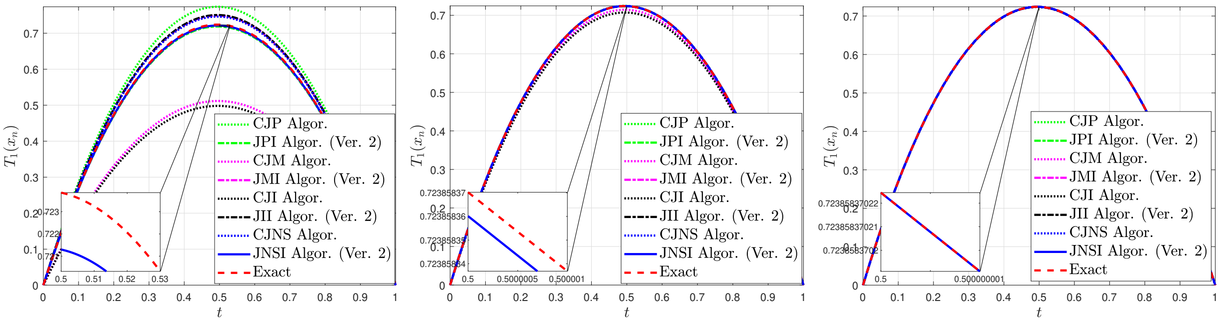


Figure 6: Comparisons of the convergence of JNSI algorithm (ver. 2) with others, for $n = 1, 3$ and 5

Remark 4.2. In certain instances, verifying conditions in the hypothesis of Theorem 2.1 may pose significant challenges. Nonetheless, it is feasible to identify examples where the practical utility of the result from Theorem 2.1 is not hindered. In the example given below, convergence can be attained using the JNSI algorithm (ver. 2) without the need to explicitly validate the conditions stipulated in Theorem 2.1.

Example 4.3. Consider the following first-order nonlinear DE:

$$\left.\begin{aligned} &3x^2(t)x'(t) = x^3(t) + (t^2+1)\mathrm{e}^t + t, \quad t \in [0,1], \\ &\qquad x(0) = 1. \end{aligned}\right\} \tag{36}$$

The exact solution for this equation is given by

$$z(t) = \left[\left(2 + t + \frac{t^3}{3}\right)\mathrm{e}^t - t - 1\right]^{1/3}.$$

Equation (36) can be transformed into the integral equation given below,

$$x^3(t) = 1 + \int_0^t \left[x^3(s) + (s^2+1)\mathrm{e}^s + s\right]\mathrm{d}s, \quad t \in [0,1].$$

Let $\mathbb{X} = C\left[0,1\right]$. We define $\mathbb{X}' \subset \mathbb{X}$ as $\mathbb{X}' := \left\{x\left(t\right) \in C\left[0,1\right] : x\left(t\right) \geq 0\right\}$ and the operators $\mathbb{T}_1, \mathbb{T}_2 : \mathbb{X}' \to \mathbb{X}$ as follows

$$\mathbb{T}_1\left(x\right) = x^3, \quad \mathbb{T}_2(x) = 1 + \int_0^t \left[x^3(s) + (s^2+1)\mathrm{e}^s + s\right] \mathrm{d}s.$$

If we take $\mathbb{T}_2\left(x_{-1}\left(t\right)\right) = \mathbb{T}_2\left(1\right)$ and $\mathbb{T}_2\left(x_0\left(t\right)\right) = \mathbb{T}_2\left(1.001\right)$, then by observing the data in Table 10, 11, 12 and Figure 7, it is evident that the sequence $\{\mathbb{T}_1\left(x_n\right)\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) converges to $\mathbb{T}_2\left(z\left(t\right)\right) = \mathbb{T}_1\left(z\left(t\right)\right) = p\left(t\right)$. Notably, this convergence is characterized by a faster rate compared to several other algorithms (CJP, JPI (ver. 2), CJM, JMI (ver. 2), CJI, JII (ver. 2) and CJNS).

Table 10: The errors $\left|\mathbb{T}_1(x_1) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	1.50×10^{-3}	0.	1.50×10^{-3}	0.	0.	0.
0.1	1.04×10^{-2}	3.36×10^{-4}	1.14×10^{-1}	5.37×10^{-3}	1.12×10^{-1}	5.28×10^{-3}	5.37×10^{-3}	1.72×10^{-4}
0.2	4.52×10^{-2}	2.89×10^{-3}	2.66×10^{-1}	2.41×10^{-2}	2.56×10^{-1}	2.34×10^{-2}	2.41×10^{-2}	1.52×10^{-3}
0.3	1.10×10^{-1}	1.04×10^{-2}	4.62×10^{-1}	6.01×10^{-2}	4.37×10^{-1}	5.77×10^{-2}	6.01×10^{-2}	5.56×10^{-3}
0.4	2.10×10^{-1}	2.60×10^{-2}	7.10×10^{-1}	1.18×10^{-1}	6.64×10^{-1}	1.12×10^{-1}	1.18×10^{-1}	1.43×10^{-2}
0.5	3.54×10^{-1}	5.39×10^{-2}	1.02	2.04×10^{-1}	9.46×10^{-1}	1.92×10^{-1}	2.04×10^{-1}	3.01×10^{-2}
0.6	5.51×10^{-1}	9.87×10^{-2}	1.41	3.25×10^{-1}	1.30	3.04×10^{-1}	3.25×10^{-1}	5.63×10^{-2}
0.7	8.12×10^{-1}	1.66×10^{-1}	1.89	4.89×10^{-1}	1.73	4.54×10^{-1}	4.89×10^{-1}	9.65×10^{-2}
0.8	1.15	2.64×10^{-1}	2.48	7.06×10^{-1}	2.26	6.52×10^{-1}	7.06×10^{-1}	1.56×10^{-1}
0.9	1.58	3.99×10^{-1}	3.20	9.89×10^{-1}	2.91	9.09×10^{-1}	9.89×10^{-1}	2.40×10^{-1}
1.0	2.12	5.83×10^{-1}	4.09	1.35	3.71	1.24	1.35	3.56×10^{-1}

Table 11: The errors $\left|\mathbb{T}_1(x_3) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	1.25×10^{-4}	0.	1.25×10^{-4}	0.	0.	0.
0.1	8.18×10^{-6}	1.56×10^{-11}	1.18×10^{-2}	3.77×10^{-7}	1.00×10^{-2}	3.50×10^{-7}	3.82×10^{-7}	2.65×10^{-13}
0.2	1.41×10^{-4}	4.36×10^{-9}	3.30×10^{-2}	7.28×10^{-6}	2.46×10^{-2}	6.30×10^{-6}	7.32×10^{-6}	1.74×10^{-10}
0.3	7.50×10^{-4}	8.49×10^{-8}	6.63×10^{-2}	4.31×10^{-5}	4.51×10^{-2}	3.51×10^{-5}	4.32×10^{-5}	4.06×10^{-9}
0.4	2.49×10^{-3}	6.78×10^{-7}	1.16×10^{-1}	1.57×10^{-4}	7.35×10^{-2}	1.22×10^{-4}	1.58×10^{-4}	3.59×10^{-8}
0.5	6.36×10^{-3}	3.38×10^{-6}	1.85×10^{-1}	4.41×10^{-4}	1.12×10^{-1}	3.25×10^{-4}	4.42×10^{-4}	1.93×10^{-7}
0.6	1.38×10^{-2}	1.26×10^{-5}	2.81×10^{-1}	1.04×10^{-3}	1.64×10^{-1}	7.38×10^{-4}	1.05×10^{-3}	7.71×10^{-7}
0.7	2.69×10^{-2}	3.83×10^{-5}	4.09×10^{-1}	2.20×10^{-3}	2.32×10^{-1}	1.50×10^{-3}	2.20×10^{-3}	2.50×10^{-6}
0.8	4.81×10^{-2}	1.01×10^{-4}	5.79×10^{-1}	4.25×10^{-3}	3.21×10^{-1}	2.80×10^{-3}	4.26×10^{-3}	7.00×10^{-6}
0.9	8.08×10^{-2}	2.38×10^{-4}	7.99×10^{-1}	7.69×10^{-3}	4.37×10^{-1}	4.93×10^{-3}	7.70×10^{-3}	1.75×10^{-5}
1.0	1.29×10^{-1}	5.13×10^{-4}	1.08	1.32×10^{-2}	5.85×10^{-1}	8.26×10^{-3}	1.32×10^{-2}	3.99×10^{-5}

Table 12: The errors $\left|\mathbb{T}_1(x_5) - p(t)\right|$

t	CJP	JPI (Ver.2)	CJM	JMI (Ver.2)	CJI	JII (Ver.2)	CJNS	JNSI (Ver.2)
0.0	0.	0.	4.17×10^{-6}	0.	4.17×10^{-6}	0.	0.	0.
0.1	2.61×10^{-9}	2.12×10^{-19}	5.62×10^{-4}	4.36×10^{-12}	3.62×10^{-4}	3.75×10^{-12}	4.44×10^{-12}	1.36×10^{-21}
0.2	1.80×10^{-7}	6.16×10^{-16}	2.01×10^{-3}	3.69×10^{-10}	9.74×10^{-4}	2.76×10^{-10}	3.72×10^{-10}	4.05×10^{-19}
0.3	2.15×10^{-6}	7.40×10^{-14}	4.89×10^{-3}	5.27×10^{-9}	1.95×10^{-3}	3.49×10^{-9}	5.31×10^{-9}	1.15×10^{-16}
0.4	1.25×10^{-5}	1.96×10^{-12}	9.99×10^{-3}	3.65×10^{-8}	3.46×10^{-3}	2.17×10^{-8}	3.67×10^{-8}	3.84×10^{-15}
0.5	4.96×10^{-5}	2.43×10^{-11}	1.83×10^{-2}	1.69×10^{-7}	5.71×10^{-3}	9.11×10^{-8}	1.70×10^{-7}	5.54×10^{-14}
0.6	1.53×10^{-4}	1.88×10^{-10}	3.12×10^{-2}	6.09×10^{-7}	9.00×10^{-3}	3.00×10^{-7}	6.11×10^{-7}	4.87×10^{-13}
0.7	4.00×10^{-4}	1.06×10^{-9}	5.05×10^{-2}	1.84×10^{-6}	1.37×10^{-2}	8.35×10^{-7}	1.84×10^{-6}	3.07×10^{-12}
0.8	9.22×10^{-4}	4.75×10^{-9}	7.83×10^{-2}	4.86×10^{-6}	2.03×10^{-2}	2.06×10^{-6}	4.87×10^{-6}	1.53×10^{-11}
0.9	1.93×10^{-3}	1.78×10^{-8}	1.17×10^{-1}	1.16×10^{-5}	2.95×10^{-2}	4.61×10^{-6}	1.16×10^{-5}	6.32×10^{-11}
1.0	3.77×10^{-3}	5.81×10^{-8}	1.72×10^{-1}	2.57×10^{-5}	4.20×10^{-2}	9.60×10^{-6}	2.57×10^{-5}	2.27×10^{-10}

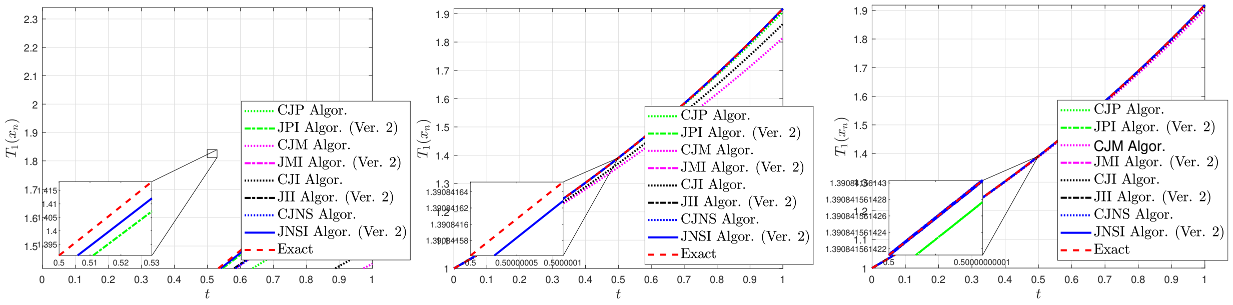


Figure 7: Comparisons of the convergence of JNSI algorithm (ver. 2) with others, for $n = 1, 3,$ and 5

4.2. *Applications for Finding Roots of Complex Polynomials and Creating Image Polynomiography*

Polynomiography is a unique mathematical and artistic technique developed by Kalantari in the late 1990s (cf. [16]). It involves visually representing complex polynomial equations in a two-dimensional plane, showcasing the roots of these equations as points. These points convey both the real and imaginary parts of the roots, while factors like color and size offer additional insights into the polynomial's properties. Polynomiography stands as a remarkable intersection where mathematics and artistic expression intertwine. Rooted in the art and science of visualizing the zeros of complex polynomials, polynomiography employs mathematical iteration functions and fractal imagery to unlock a realm of both scientific exploration and creative artistry. This discipline encapsulates a dual purpose: it serves as a tool for approximating polynomial roots, shedding light on a fundamental mathematical problem, while simultaneously offering a canvas for artists and enthusiasts to craft intricate and mesmerizing visual representations. The synergy between mathematics and computer technology inherent in polynomiography not only transforms how we perceive complex equations but also forges an innovative path that enriches both the scientific and artistic landscapes. We can mention [8], [10], [30], [36] for various studies on polynomiography.

Example 4.4. Let us consider the equation $P(z) = z^3-2z-1 = 0$, where $P(z)$ represents a complex polynomial. If we take

$$\mathbb{T}_2(z) = z^3 - 1 \quad \text{and} \quad \mathbb{T}_1(z) = 2z,$$

then we can establish the equivalence:

$$P(z) = 0 \;\Leftrightarrow\; \mathbb{T}_2(z) = \mathbb{T}_1(z).$$

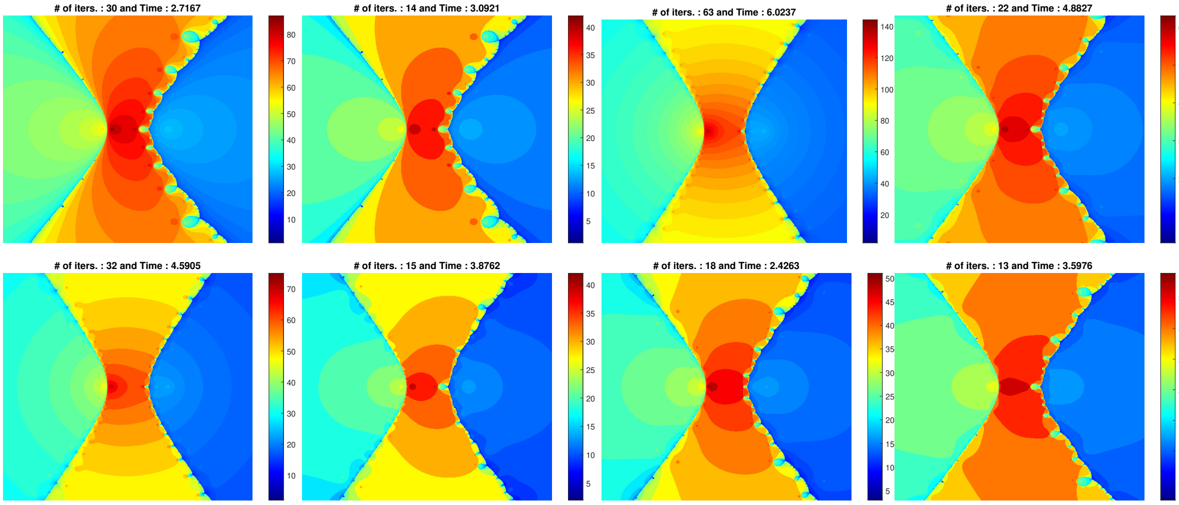


Figure 8: Example 4.4 (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)

For any value of $\omega \neq -1$, specifically chosen such that $\omega = -\mathbb{T}_1'(z)/\mathbb{T}_2'(z)$, we can deduce

$$\omega\mathbb{T}_2(z) = \omega\mathbb{T}_1(z) \;\Leftrightarrow\; \mathbb{T}_1(z) = \mathbb{T}_2^{\omega}(z) = \mathbb{T}_2(z) = \frac{\omega\mathbb{T}_2(z) + \mathbb{T}_1(z)}{\omega + 1}$$

(as detailed in [31]). To determine all the roots of $P(z)$ and generate polynomiographs as shown in Figure 8, we adopt $\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.5$, $\theta = 10^{-6}$, an accuracy of $eps = 10^{-7}$, and confine the range to the area $[-5,5]^2$. As depicted in Figure 8, it is evident that the inertial type algorithm are faster then original algorithms and the sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) converges faster than any other algorithms.

Example 4.5. Consider the problem of solving $P(z) = z^5 + 2x - \mathrm{i} + 1 = 0$, where $P(z)$ is a complex polynomial.

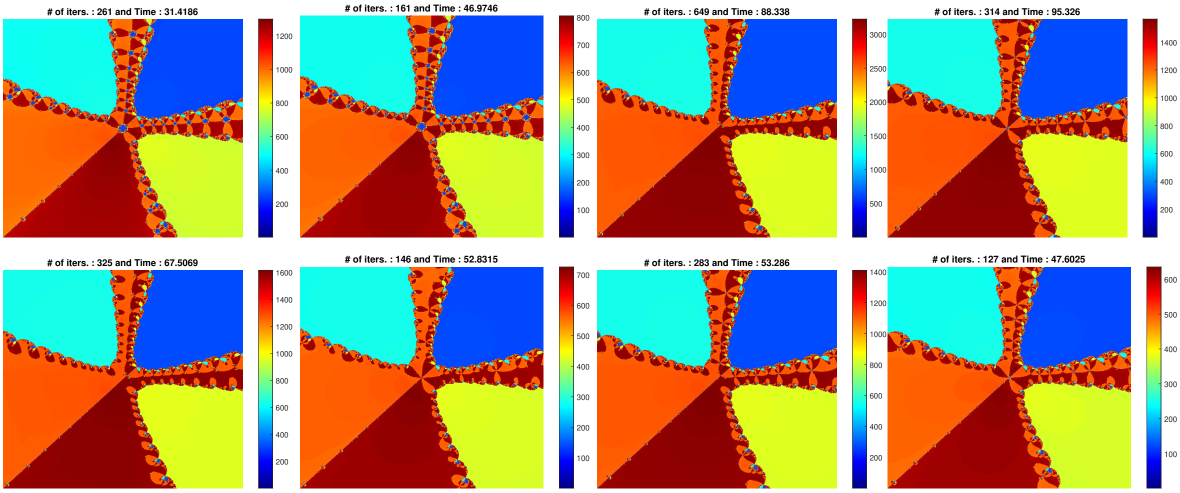


Figure 9: Example 4.5 (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)

We define the following transformations:

$$\mathbb{T}_2(z) = z^5 - \mathrm{i} + 1 \quad \text{and} \quad \mathbb{T}_1(z) = -2z.$$

Similar to the previous example, we then apply $\mathbb{T}_2^{\omega}(z)$. To determine all the roots of $P(z)$ and generate polynomiographs shown in Figure 9, we select parameters as follows:

$$\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.9, \quad \theta = 10^{-6},$$

accuracy $eps = 10^{-7}$ and domain $[-5, 5]^2$. As seen in Figure 9, it is evident that the inertial type algorithms are faster then original algorithms and the sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) exhibits faster convergence compared to all other algorithms.

Example 4.6. Let us consider the problem of solving

$$P(z) = z^7 + 2z^4 + z^3 - 4z^2 + 2z + 1 = 0,$$

where $P(z)$ is a complex polynomial.

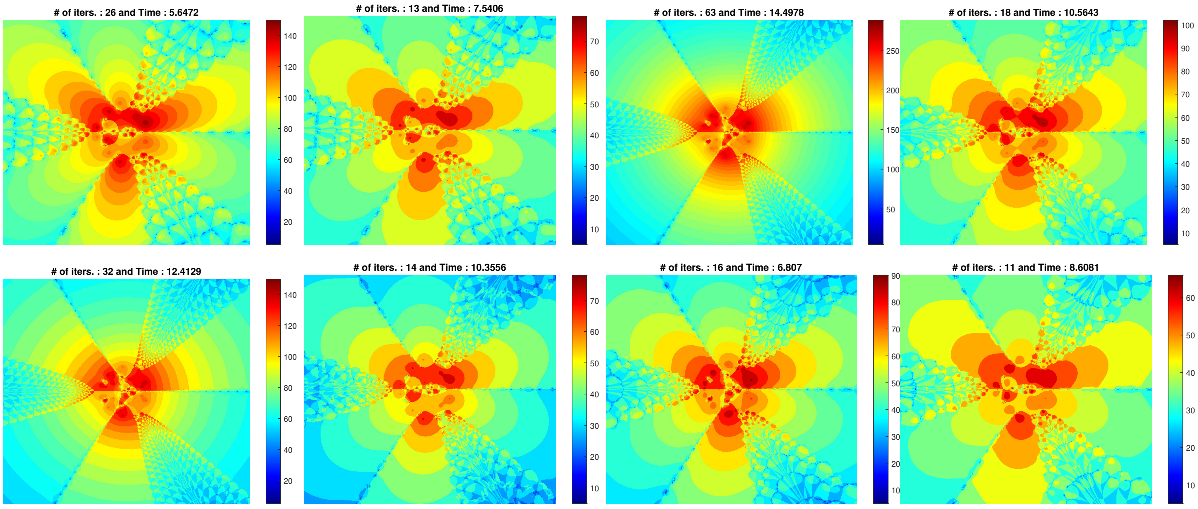


Figure 10: Example 4.6 (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)

We define the following transformations

$$\mathbb{T}_2(z) = z^7 + 2z^4 - 4z^2 + 2z \quad \text{and} \quad \mathbb{T}_1(z) = -z^3 - 1.$$

Similar to the previous example, we iterate $\mathbb{T}_2^{\omega}(z)$ as well. In order to determine all the roots of $P(z)$ and to obtain polynomiographs as shown in Figure 10, we set

$$\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.9, \quad \theta = 10^{-6},$$

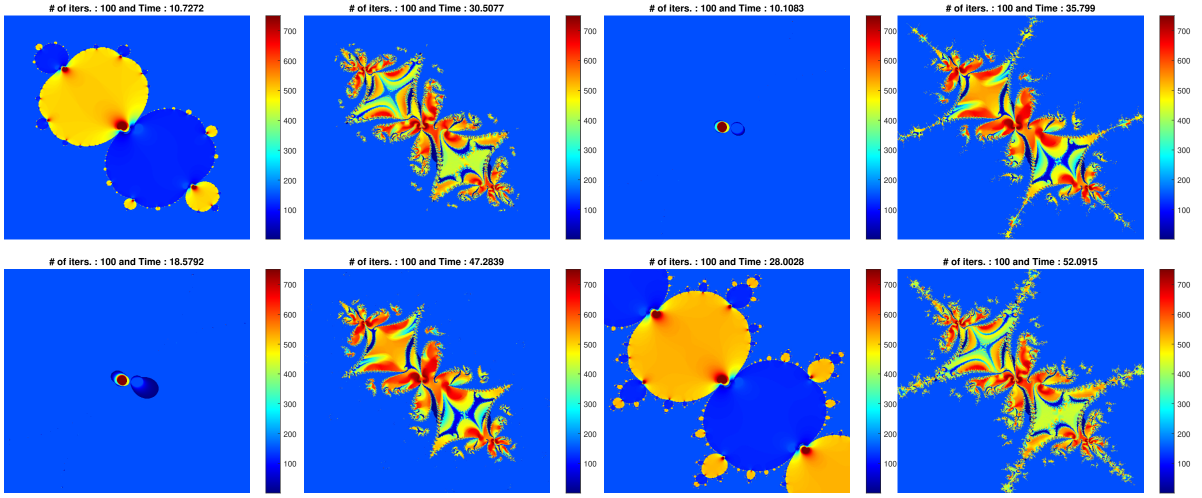
accuracy $eps = 10^{-7}$, and the region of interest as $[-5, 5]^2$.

From Figure 10, it is evident that the inertial type algorithms are faster then original algorithms. The sequence $\{\mathbb{T}_1 x_n\}_{n=1}^{\infty}$ generated by JNSI algorithm (ver. 2) demonstrates faster convergence in comparison to all the other algorithms.

In the following example, we show the affect of parameters to the images.

Example 4.7. Consider the polynomials $P(z)$, $\mathbb{T}_2(z)$, $\mathbb{T}_1(z)$, and $\mathbb{T}_2^{\omega}(z)$ as defined in Example 4.4.

To observe the effects of various parameters on polynomiographs, we generated, images which are shown in Figures 11 – 14, using different parameter values without concerning ourselves with the convergence of iterations.


# of iters. : 100 and Time : 10.7272

# of iters. : 100 and Time : 30.5077

# of iters. : 100 and Time : 10.1083

# of iters. : 100 and Time : 35.799

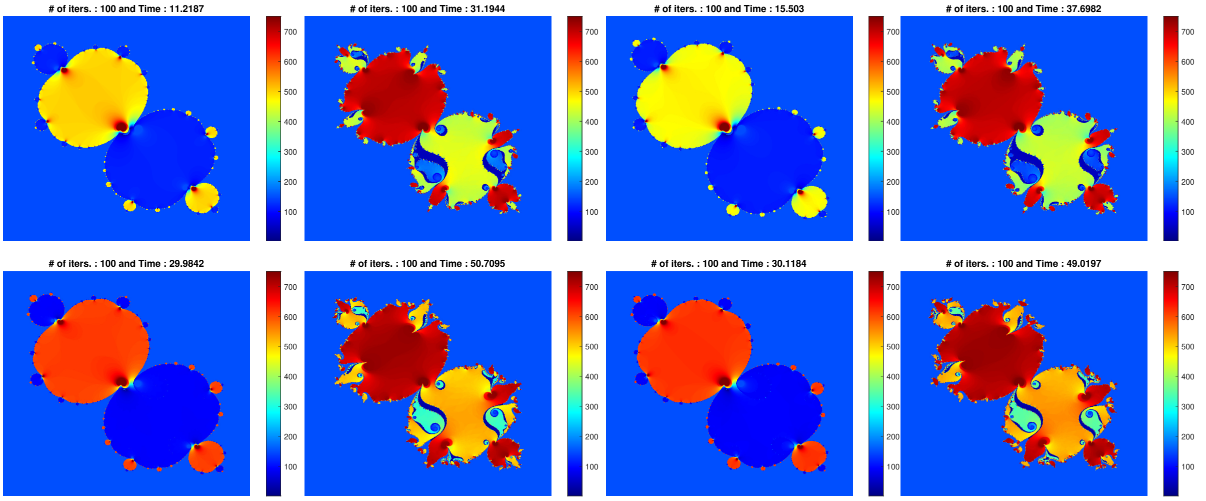
# of iters. : 100 and Time : 18.5792

# of iters. : 100 and Time : 47.2839

# of iters. : 100 and Time : 28.0028

# of iters. : 100 and Time : 52.0915

Figure 11: Example 4.7 for $\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.5$ for all $n \in \mathbb{N}_0$ and $i = 1, 2, 3$, $\theta = 0.9$, accuracy $eps = 1$, the area being $[-20, 20]^2$, the iteration count fixed at 100 and $\omega = 0.009i$: (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)


# of iters. : 100 and Time : 11.2187

# of iters. : 100 and Time : 31.1944

# of iters. : 100 and Time : 15.503

# of iters. : 100 and Time : 37.6982

# of iters. : 100 and Time : 29.9842

# of iters. : 100 and Time : 50.7095

# of iters. : 100 and Time : 30.1184

# of iters. : 100 and Time : 49.0197

Figure 12: Example 4.7 for $\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.9$ for all $n \in \mathbb{N}_0$, $\theta = 0.5$, accuracy $eps = 1$, the area being $[-20, 20]^2$, the iteration count fixed at 100 and $\omega = 0.009i$: (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)

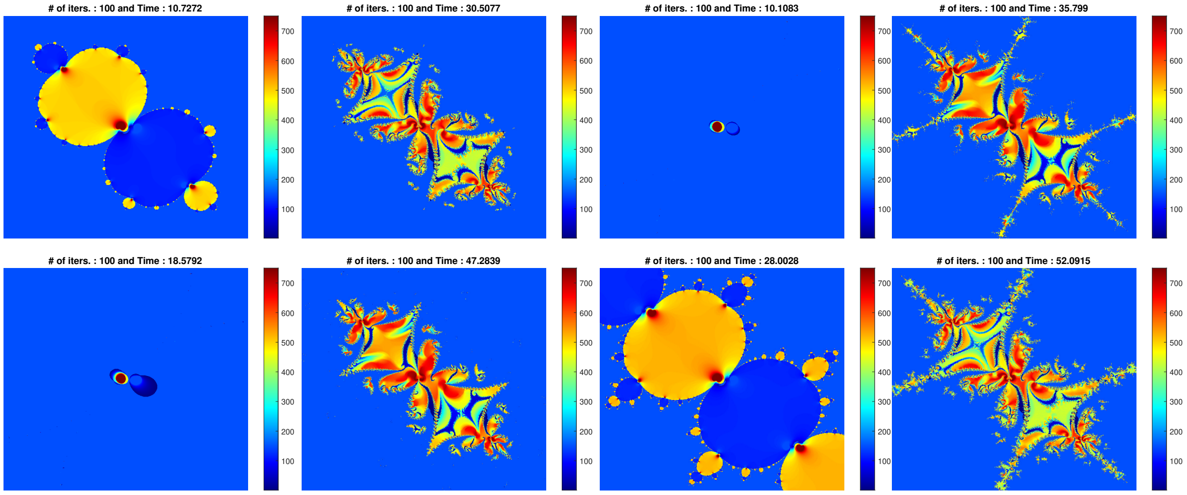


Figure 13: Example 4.7 for $\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.1$ for all $n \in \mathbb{N}_0$, $\theta = 0.9$, accuracy *eps* = 1, the area being $[-20, 20]^2$, the iteration count fixed at 100 and $\omega = 0.009i$: (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)

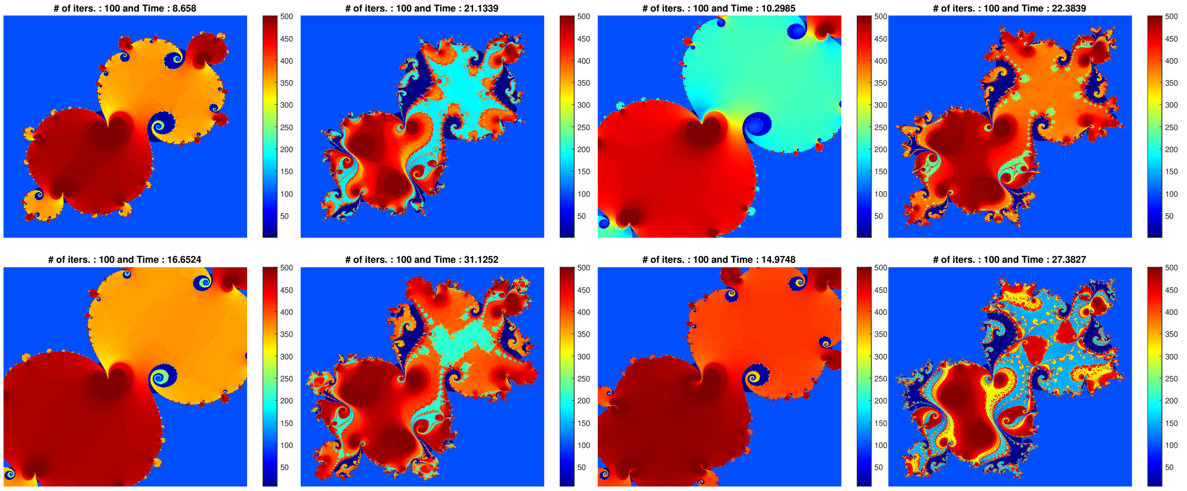


Figure 14: Example 4.7 for $\alpha_n = \alpha_n^1 = \alpha_n^2 = \alpha_n^3 = 0.5$ for all $n \in \mathbb{N}_0$, $\theta = 0.5$, accuracy *eps* = 1, the area being $[-6, 6]^2$, the iteration count fixed at 100 and $\omega = -0.0001 - 0.08i$: (from left to right): Algorithms CJP, JPI (ver. 2), CJM, JMI (ver. 2) (first line); Algorithms CJI, JII (ver. 2), CJNS, JNSI (ver. 2) (second line)

5. Conclusion

In this study, we would investigate the convergence of algorithms, assuming that the inertial algorithms available in the literature for the fixed point problem will work effectively for the pairs of mappings that satisfy quasi (L, δ)-contractive condition, with the intention of adapting them to the coincidence point problem. However, as we delved deeper into the study, we saw that, surprisingly, these were not effective in accelerating the classical Jungck algorithms due to their generally accepted structure in the literature. We were able to overcome this difficulty by defining an iteration with a somewhat contradictory inertial step $(\theta_n(\mathbb{T}_2 x_n - \mathbb{T}_2 x_{n-1}))$. We defined and studied effective algorithms in terms of convergence speed for the coincidence point problem called “Jungck inertial type algorithms”. These algorithms offered a different structural approach compared to classical inertial algorithms, resulting in increased performance and efficiency. In this context, we observed with examples that the inertial step in the proposed new structure also accelerates the classical Jungck-Picard, classical Jungck-Mann and classical Jungck-Ishikawa algorithms. We obtained extensive analyzes of the convergence of new inertial type algorithms and demonstrated the stability and data dependency properties for one of these algorithms. We tested the obtained theoretical results with non-obvious examples in finite dimensional spaces. Finally, in order to reveal how these theoretical results can be applied practically, we used them to find approximate solutions to the solutions of integral equations and differential equations, as well as to approximately find the roots of complex polynomials and obtained polynomiographs to turn them into visual art. We believe that the demonstrated advantages of these algorithms will facilitate their broader adoption and offer valuable insights for future developments in the field of fixed/coincidence point theory and related research areas.

References

[1] F. Ali, J. Ali, Convergence, stability, and data dependence of a new iterative algorithm with an application, Comput. Appl. Math. **39**(4) (2020) Paper No. 267, 15 pp.

[2] F. Alvarez, H. Attouch, An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping, Wellposedness in optimization and related topics (Gargnano, 1999) Set-Valued Anal. **9**, no. 1-2 (2001) 3–11.

[3] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. **2** (2009) 183–202.

[4] V. Berinde, Iterative Approximation of Fixed Points, Springer, Berlin–Heidelberg, 2007.

[5] C. Chen, R.H. Chan, S. Ma, J. Yang, Inertial proximal ADMM for linearly constrained separable convex optimization, SIAM J. Imaging Sci. **8**(4) (2015) 2239–2267.

[6] A. D'aspremont, D. Scieur, A. Taylor, Acceleration Methods, Foundations and Trends in Optimization **5**, No. 1-2 (2021), pp. 1–245; DOI:10.1561/2400000036.

[7] J. Garcia-Falset, O. Muñiz-Pérez, K. Sadarangani, Coincidence problems under contractive type conditions, Fixed Point Theory **18**(1) (2017) 213–222.

[8] K. Gdawiec, W. Kotarski, A. Lisowska, On the robust Newton's method with the Mann iteration and the artistic patterns from its dynamics, Nonlinear Dynam. **104** (2021) 297–331.

[9] K. Goebel, A coincidence theorem, Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys. **16** (1968) 733–735.

[10] I. Gościniak, K. Gdawiec, One more look on visualization of operation of a root-finding algorithm, Soft. Comput. **24** (2020) 14135–14155.

[11] F. Gürsoy, A.R. Khan, M. Ertürk, V. Karakaya, Weak w^2-stability and data dependence of Mann iteration method in Hilbert spaces, Rev. R. Acad. Cienc. Exactas Fís. Nat. Ser. A Mat. RACSAM, **113** (2019) 11–20.

[12] F. Gürsoy, A.R. Khan, M. Ertürk, V. Karakaya, Coincidences of nonself operators by a simpler algorithm. Numer. Funct. Anal. Optim. **41**(2) (2020) 192–208.

[13] E. Hacıoğlu, F. Gürsoy, S. Maldar, Y. Atalan, G.V. Milovanović, Iterative approximation of fixed points and applications to two-point second-order boundary value problems and to machine learning, Appl. Numer. Math. **167** (2021) 143–172.

[14] N. Hussain, V. Kumar, M.A. Kutbi, On the rate of convergence of Jungck-type iterative schemes, Abstr. Appl. Anal. 2013, Art. ID 132626, 15 pp.

[15] G. Jungck, Commuting mappings and fixed points, Amer. Math. Monthly **83**(4) (1976) 261–263.

[16] B. Kalantari, Polynomial Root-Finding and Polynomiography, World Scientific, Singapore, 2009.

[17] A. Keten Çopur, E. Hacıoğlu, F. Gürsoy, New insights on a pair of quasi-contractive operators in Banach spaces: Results on Jungck type iteration algorithms and proposed open problems, Math. Comput. Simulation **215** (2024) 476–497.

[18] A.R. Khan, F. Gürsoy, V. Karakaya, Jungck-Khan iterative scheme and higher convergence rate, Int. J. Comput. Math. **93**(12) (2016) 2092–2105.

[19] A.R. Khan, F. Gürsoy, V. Kumar, Stability and data dependence results for the Jungck-Khan iterative scheme. Turkish J. Math. **40**(3) (2016) 631–640.

[20] Q.-H. Liu, A convergence theorem of the sequence of Ishikawa iterates for quasi-contractive mappings, J. Math. Anal. Appl. **146**(2) (1990) 301–305.

[21] D.A. Lorenz, T. Pock, An inertial forward-backward algorithm for monotone inclusions, J. Math. Imaging Vision **51**(2) (2015) 311–325.

[22] R. Machuca, A coincidence theorem, Amer. Math. Monthly **74**(5) (1967) 569 https://doi.org/10.2307/2314896.

[23] P.E. Maingé, Convergence theorems for inertial KM-type algorithms, J. Comput. Appl. Math. **219** (2008) 223–236.

[24] S. Maldar, Iterative algorithms of generalized nonexpansive mappings and monotone operators with application to convex minimization problem, J. Appl. Math. Comput. **68**(3) (2022) 1841–1868.

[25] S. Micula, G. V. Milovanović, Iterative processes and integral equations of the second kind, In: Matrix and Operator Equations and Applications (M. S, Moslehian, ed.), pp. 661–711, Springer, Cham, 2023; https://doi.org/10.1007/16618_2023_59.

[26] O. Mleşniţe, Existence and Ulam-Hyers stability results for coincidence problems, J. Nonlinear Sci. Appl. **6**(2) (2013) 108–116.

[27] A. Moudafi, M. Oliny, Convergence of a splitting inertial proximal method formonotone operators, J. Comput. Appl. Math. **155**(2) (2003) 447–454.

[28] M.O. Olatinwo, Some stability and strong convergence results for the Jungck-Ishikawa iteration process, Creative Math. Inform. **17** (2008) 33–42.

[29] B.T. Polyak, Some methods of speeding up the convergence of iteration methods, USSR Comput. Math. Math. Phys. **4**(5) (1964) 1–17.

[30] S. Qureshi, I.K. Argyros, A. Soomro, K. Gdawiec, A.A. Shaikh, E. Hincal, A new optimal root-finding iterative algorithm: local and semilocal analysis with polynomiography, Numer. Algorithm **25**(4) (2024) 1715–1745.

[31] A. Rafiq, M. Tanveer, W. Nazeer, S.M. Kang, Polynomiography via modified Jungck, modified Jungck Mann and modified Jungck Ishikawa iteration schemes, PanAmer. Math. J. **24**(4) (2014) 66–95.

[32] D.R. Sahu, Applications of the S-iteration process to constrained minimization problems and split feasibility problems, Fixed Point Theory **12** (2011) 187–204.

[33] Y. Shehu, A. Gibali, S. Sagratella, Inertial projection-type methods for solving quasi-variational inequalities in real Hilbert spaces, J. Optim. Theory Appl. **184**(3) (2020) 877–894.

[34] S.L. Singh, C. Bhatnagar, S.N. Mishra, Stability of Jungck-type iterative procedures, Int. J. Math. Math. Sci. **19** (2005) 3035–3043.

[35] I. Timiş, V. Berinde, Weak stability of iterative procedures for some coincidence theorems, Creat. Math. Inform. **19**(1) (2010) 85–95.

[36] G.I. Usurelu, A. Bejenaru, M. Postolache, Newton-like methods and polynomiographic visualization of modified Thakur processes, Int. J. Comput. Math. **98**(5) (2021) 1049–1068.