J. Korean Soc. Ind. Appl. Math. Vol.26, No.3, 138–155, 2022 https://doi.org/10.12941/jksiam.2022.26.138

AN IMPROVED IMPLICIT EULER METHOD FOR SOLVING INITIAL VALUE PROBLEMS

BEONG IN YUN[1]

[1]DEPARTMENT OF MATHEMATICS, KUNSAN NATIONAL UNIVERSITY, GUNSAN, REPUBLIC OF KOREA
Email address: biyun@kunsan.ac.kr

ABSTRACT. To solve the initial value problem we present a new single-step implicit method based on the Euler method. We prove that the proposed method has convergence order 2. In practice, numerical results of the proposed method for some selected examples show an error tendency similar to the second-order Taylor method. It can also be found that this method is useful for stiff initial value problems, even when a small number of nodes are used. In addition, we extend the proposed method by using weighted averages with a parameter and show that its convergence order becomes 2 for the parameter near $\frac{1}{2}$. Moreover, it can be seen that the extended method with properly selected values of the parameter improves the approximation error more significantly.

1. INTRODUCTION

In this work we consider an initial value problem as follows

$$\begin{cases} y'(t) &= f(t,y), \qquad a \le t \le b \\ y(a) &= \alpha \end{cases} \tag{1.1}$$

where f is a real valued function with which this initial value problem has a unique solution $y = y(t)$, $a \le t \le b$.

Classical methods for numerical solutions for the initial value problem can be found in many literatures (see for example [1, 2, 3, 4]). Among them, the well-known Taylor method of order n is a typical single-step numerical method which has convergence order n for a step size $h > 0$. We are interested in the Euler method(or first-order Taylor method), which is the simplest and standard single-step numerical method. The Euler method is, however, less accurate than other higher order methods such as the Runge-Kutta method and the linear multistep method. On the other hand, the implicit Euler method(or backward Euler method) is also one of the most basic numerical methods for solving ordinary differential equations. In general, the implicit method is known to be effective for stiff initial value problems even when the step size is not very small

Received July 6 2022; Revised July 29 2022; Accepted in revised form August 6 2022; Published online September 25 2022.

2000 *Mathematics Subject Classification.* 65L05.

Key words and phrases. initial value problem, Euler method, implicit equation.

[5, 6, 7, 8]. Recently, a linear multistep method using the standard implicit Euler method and a time filter is introduced in [9].

In this work, we aim to propose a new single-step implicit method based on the Euler method which can improve convergence order of the original one. In the following section we develop an implicit Euler method including the mean of the approximate solutions y_i and y_{i+1} at the adjacent points t_i and t_{i+1}, respectively. It is proved that the proposed method enhances convergence order of the standard implicit Euler method from one to two. Solving the nonlinear implicit equation for the next approximation y_{i+1} with the Newton method yields an appropriate iterative formula.

Numerical results for some selected examples show that the proposed method causes approximation errors similar to the second-order Taylor method, as expected from the error analysis. It is also seen that the method is particularly effective for the stiff initial value problem.

In section 3 we propose a modified method using parameterized weighed averages, which is an extended version of the method provided in the previous section. Analysis of the error bound and the stability of the method, associated with a parameter $0 < \delta \leq 1$, is performed. It is demonstrated that the proposed method has convergence order of two for δ close enough to $\frac{1}{2}$ and is A-stable [10, 11, 12] for every $0 < \delta \leq \frac{1}{2}$. Numerical results of the proposed method for some examples show that the approximation error can be significantly improved by choosing appropriate values for the parameter.

2. An implicit Euler method associated with the Newton method

For the equidistant grid points

$$t_j = a + jh\,, \qquad j = 0, 1, 2, \ldots, N,$$

with step size $h = (b-a)/N$, we recall the following generalized Euler method[1] by which one can obtain approximations $\{y_j\}_{j=1}^N$ to the solutions $\{y(t_j)\}_{j=1}^N$ of the initial problem (1.1).

$$y_{i+1} = y_i + h\phi(t_i, y_i; h)\,, \qquad i = 0, 1, 2, \ldots, N-1, \tag{2.1}$$

where the function ϕ is given in terms of f, in the initial problem (1.1), and the step size h. The above explicit method is the prototype of the implicit method developed in this work, and its numerical results will be compared with the method (2.1) of some typical cases of ϕ. For example, the standard Euler method(or first-order Taylor method) has

$$\phi(t, y; h) = f(t, y)$$

and the modified Euler method(or second-order Runge-Kutta method) has

$$\phi(t, y; h) = \frac{1}{2}\left[f(t, y) + f(t+h,\, y + hf(t, y))\right].$$

In addition, the second-order Taylor method has

$$\phi(t, y; h) = f(t, y) + \frac{h}{2} f'(t,\, y)\,,$$

where f' is a derivative of $f(t, y(t))$ with respect to the variable t, that is,

$$f'(t,y) = \frac{d}{dt} f(t, y(t)) = f_t(t, y(t)) + f_y(t, y(t)) f(t, y(t)) \,.$$

In the following subsection we develop a new single-step method, based on an implicit version of the standard Euler method, whose error tendency turns to be the same as that of the typical second-order methods aforementioned.

2.1. **A second order implicit method.** We focus on the Euler method, the basic single-step method, such as

$$x \,=\, w \,+\, hf(t_i, w), \qquad i = 0, 1, 2, \ldots, N-1, \tag{2.2}$$

where $w \,:=\, y_i$ and $x \,:=\, y_{i+1}$ are the approximations to the solutions $y(t_i)$ and $y(t_{i+1})$, respectively.

For each i fixed, setting

$$\bar{t}_i \,=\, \frac{t_i + t_{i+1}}{2}\,, \qquad \overline{w} \,=\, \frac{w + x}{2} \tag{2.3}$$

and referring to the standard Euler method (2.2), we suggest an implicit equation to determine x as follows.

$$x \,=\, \overline{w} \,+\, \frac{h}{2} f(\bar{t}_i, \overline{w}).$$

From (2.3) this equation simply becomes

$$x \,=\, w \,+\, hf(\bar{t}_i, \overline{w}). \tag{2.4}$$

It can be seen that the method (2.4) coincides with the well known implicit midpoint method as a result. However, as can be seen from the preceding formula, the derivation of (2.4) is based on pivoting the mean $\bar{\omega} = (x + w)/2$ when applying the standard Euler method. This is the motivation behind the extended implicit method developed in Section 3 using a weighted average between w and x.

From the following theorem we can see that convergence order of the method proposed above is 2. The theorem is a special case of $\delta = \frac{1}{2}$ in Theorem 3.1 given in the next section, so the proof is omitted.

Theorem 2.1. *Suppose $f(t, y)$, in the initial value problem* (1.1)*, satisfies a Lipschitz condition in the variable y with a Lipschitz constant L on a set $D = \{(t, y) \mid a \leq t \leq b, \ -\infty < y < \infty\}$. Furthermore, let f be twice continuously differentiable in D. Then, for each $i = 0, 1, 2, \ldots, N-1$ the approximate solution $y_{i+1}\,(= x)$ to the exact solution $y(t_{i+1})$ obtained by the formula* (2.4) *satisfies*

$$|y\,(t_{i+1}) - y_{i+1}| \leq \frac{C}{L} h^2 \left\{ e^{(i+1)hL} - 1 \right\},$$

where C is a positive constant and the step size h is assumed to be small enough.

Since the implicit Eq.(2.4) cannot be solved algebraically for $x\,(=y_{i+1})$ in general, a numerical method to approximate x is required. In practice, if we define a function F by

$$F(x) \;=\; x - w \;-\; hf(\bar{t}_i, \overline{w}),$$

then the problem of finding x in the Eq.(2.4) becomes the root-finding for the equation $F(x) = 0$. Using the Newton method with an initial guess $x^{(0)} = w\ (= y_i)$, we consider the following iterative method.

$$\begin{aligned} x^{(k)} \;&=\; x^{(k-1)} \;-\; \frac{F(x^{(k-1)})}{F'(x^{(k-1)})} \\ &=\; x^{(k-1)} \;-\; \frac{x^{(k-1)} - w - hf\left(\bar{t}_i, \frac{w+x^{(k-1)}}{2}\right)}{1 - hf_x\left(\bar{t}_i, \frac{w+x^{(k-1)}}{2}\right)}, \qquad k = 1, 2, \ldots, k_{\max}, \end{aligned} \tag{2.5}$$

where f_x is a partial derivative defined, for $\eta = \overline{w}$, by

$$\begin{aligned} f_x(\bar{t}_i, \eta) \;&:=\; \frac{\partial}{\partial x} f\left(\bar{t}_i, \overline{w}\right) \\ &=\; \frac{\partial}{\partial x} f\left(\bar{t}_i, \frac{w+x}{2}\right) \;=\; \frac{1}{2}\left[\frac{\partial}{\partial \eta} f\left(\bar{t}_i, \eta\right)\right]. \end{aligned} \tag{2.6}$$

When, for some k, the iterate $x^{(k)}$ obtained by the formula (2.5) satisfies a proper tolerance error, then we set $x = y_{i+1} = x^{(k)}$.

If the step size h is small enough then the number of iterations $k_{\max}$ does not need to be large. As a special case of a single-iteration, that is, $k_{\max} = 1$ in (2.5) with $x^{(0)} = w$, we have the following simple formula.

$$x \;=\; x^{(1)} \;=\; w \;+\; \frac{hf(\bar{t}_i, w)}{1 - hf_x(\bar{t}_i, w)}. \tag{2.7}$$

On the other hand, if $f(t, y)$ is a linear map with respect to y in particular then it follows that $F(x^{(1)}) = 0$ and $x^{(2)} = x^{(1)}$. Therefore, we have $F\left(x^{(k)}\right) \;=\; 0$ for every $k \geq 1$. That is, the iteration (2.5) is independent of k. Thus, in this case the proposed method (2.5) is reduced to the single-iteration formula (2.7).

2.2. **Numerical examples.** To explore the availability of the proposed method, we select some typical examples.

Example 1.

$$\begin{cases} y'(t) &= (1-t)y^2, \qquad -2 \leq t \leq 2 \\ y(-2) &= 0.2 \end{cases}$$

whose exact solution is $y = \frac{2}{2-2t+t^2}$.

Table 1 includes numerical errors, $e_{i+1} := |y_{i+1} - y(t_{i+1})|$ of the approximates y_{i+1} obtained by the proposed method (2.5) with $N = 20$. Numerical errors of the standard Euler method and the second-order Taylor method are also included for comparison. From the table one can see that the proposed method (2.5) is competitive to the second-order Taylor method.

In addition, we denote by $E_{2,N}$ the l_2–norm error of the approximates $\{y_{i+1}\}_{i=0}^{N-1}$ defined as

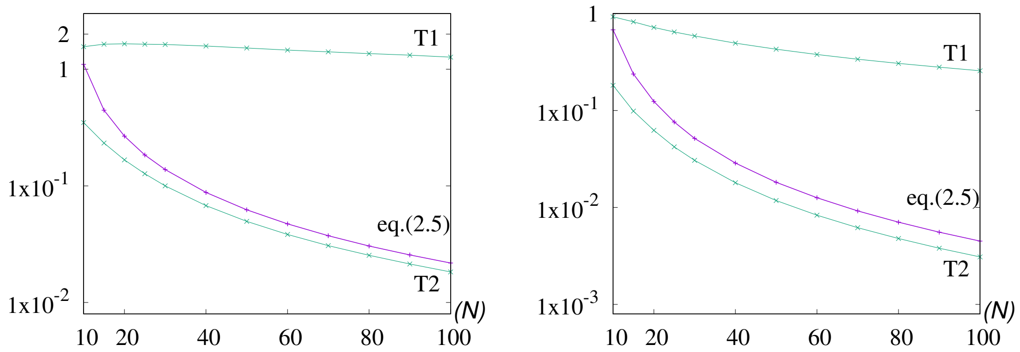
$$E_{2,N} = \left[\sum_{i=0}^{N-1} e_{i+1}^2\right]^{1/2}$$

and denote by $E_{\infty,N}$ the l_∞–norm error defined as

$$E_{\infty,N} = \max_{0\le i\le N-1} e_{i+1}.$$

TABLE 1. Differences between the approximate solutions $\{y_j\}_{j=0}^N$ and the exact solutions $\{y(t_j)\}_{j=0}^N$ for Example 1 ($N = 20$).

t_j	Existing methods: Euler method	Existing methods: Taylor method of order 2	Presented method: eqn. (2.5)	Presented method: eqn. (3.3) (with $\delta = \delta^*$)
-2.0	0	0	0	0
-1.8	2.2×10^{-3}	1.6×10^{-4}	1.1×10^{-4}	6.2×10^{-5}
-1.6	5.6×10^{-3}	4.3×10^{-4}	3.0×10^{-4}	1.5×10^{-4}
-1.4	1.1×10^{-2}	8.5×10^{-4}	6.1×10^{-4}	2.8×10^{-4}
-1.2	1.8×10^{-2}	1.5×10^{-3}	1.1×10^{-3}	4.6×10^{-4}
-1.0	3.0×10^{-2}	2.5×10^{-3}	1.9×10^{-3}	7.0×10^{-4}
-0.8	4.6×10^{-2}	4.2×10^{-3}	3.3×10^{-3}	1.0×10^{-3}
-0.6	7.1×10^{-2}	6.7×10^{-3}	5.6×10^{-3}	1.5×10^{-3}
-0.4	1.1×10^{-1}	1.1×10^{-2}	9.3×10^{-3}	2.1×10^{-3}
-0.2	1.6×10^{-1}	1.7×10^{-2}	1.6×10^{-2}	2.8×10^{-3}
0.0	2.4×10^{-1}	2.5×10^{-2}	2.6×10^{-2}	3.5×10^{-3}
0.2	3.4×10^{-1}	3.7×10^{-2}	4.1×10^{-2}	4.2×10^{-3}
0.4	4.7×10^{-1}	4.9×10^{-2}	6.4×10^{-2}	4.3×10^{-3}
0.6	6.0×10^{-1}	5.9×10^{-2}	9.1×10^{-2}	3.3×10^{-3}
0.8	7.0×10^{-1}	6.3×10^{-2}	1.1×10^{-1}	1.3×10^{-3}
1.0	7.2×10^{-1}	6.0×10^{-2}	1.2×10^{-1}	5.0×10^{-3}
1.2	6.4×10^{-1}	5.8×10^{-2}	1.1×10^{-1}	9.4×10^{-3}
1.4	5.1×10^{-1}	5.8×10^{-2}	9.1×10^{-2}	9.8×10^{-3}
1.6	3.7×10^{-1}	5.2×10^{-2}	6.4×10^{-2}	6.3×10^{-3}
1.8	2.7×10^{-1}	4.2×10^{-2}	4.1×10^{-2}	1.8×10^{-3}
2.0	1.9×10^{-1}	3.0×10^{-2}	2.6×10^{-2}	1.7×10^{-3}
l_2-error($E_{2,N}$)	1.70	0.17	0.27	0.018

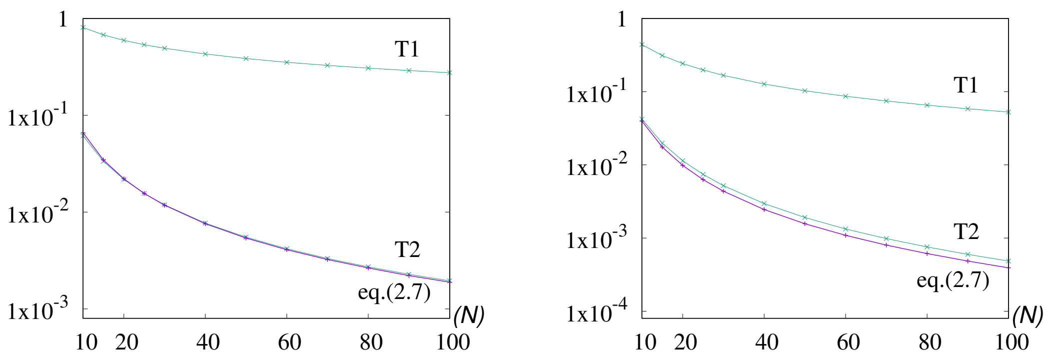


(a) $E_{2,N}$ (b) $E_{\infty,N}$

FIGURE 1. Graphs of the l_2-norm errors, $E_{2,N}$ in (a) and the l_∞-norm errors, $E_{\infty,N}$ in (b), $10 \le N \le 100$, for Example 1.

Figure 1 shows the distribution of the errors $E_{2,N}$ and $E_{\infty,N}$ for the approximates $\{y_{i+1}\}_{i=0}^{N-1}$ obtained by the proposed methods (2.5), compared with those of the original Euler method and the second-order Taylor method(indicated by T1 and T2, respectively), with respect to the number of nodes $10 \le N \le 100$.

Example 2. (Burden and Paires[3])



(a) $E_{2,N}$ (b) $E_{\infty,N}$

FIGURE 2. Graphs of the l_2-norm errors, $E_{2,N}$ in (a) and the l_∞-norm errors, $E_{\infty,N}$ in (b), $10 \le N \le 100$, for Example 2.

$$\begin{cases} y'(t) &= y - t^2 + 1, \qquad 0 \le t \le 2 \\ y(0) &= 0.5 \end{cases}$$

whose exact solution is $y = (t+1)^2 - \frac{1}{2}e^t$.

Since $f(t, y) = y - t^2 + 1$ is a linear map of y, the iteration (2.5) is reduced to the single-iteration formula (2.7). Table 2 includes numerical errors, $|y_{i+1} - y(t_{i+1})|$ of the approximates y_{i+1} obtained by the proposed method (2.7) with $N = 20$. As in the case of Example 1, the proposed method (2.7) shows similar error tendency to the Taylor method of order 2.

Figure 2 shows the distribution of the l_2-norm error $E_{2,N}$ and the l_∞-norm error $E_{\infty,N}$ for the approximates $\{y_{i+1}\}_{i=0}^{N-1}$ obtained by the proposed methods (2.7), the original Euler method and the second-order Taylor method, with respect to the number of nodes $10 \le N \le 100$.

TABLE 2. Differences between the approximate solutions $\{y_j\}_{j=0}^N$ and the exact solutions $\{y(t_j)\}_{j=0}^N$ for Example 2 ($N = 20$).

	Existing methods		Presented method	
t_j	Euler method	Taylor method of order 2	eqn. (2.7)	eqn. (3.4) (with $\delta = \delta^*$)
0.0	0	0	0	0
0.1	7.4×10^{-3}	8.6×10^{-5}	1.2×10^{-4}	4.2×10^{-5}
0.2	1.5×10^{-2}	1.9×10^{-4}	4.5×10^{-4}	8.3×10^{-5}
0.3	2.4×10^{-2}	3.1×10^{-4}	7.1×10^{-4}	1.2×10^{-4}
0.4	3.3×10^{-2}	4.6×10^{-4}	9.8×10^{-4}	1.6×10^{-4}
0.5	4.2×10^{-2}	6.4×10^{-4}	1.3×10^{-3}	2.0×10^{-4}
0.6	5.2×10^{-2}	8.5×10^{-4}	1.6×10^{-3}	2.3×10^{-4}
0.7	6.2×10^{-2}	1.1×10^{-3}	2.0×10^{-3}	2.6×10^{-4}
0.8	7.3×10^{-2}	1.4×10^{-3}	2.3×10^{-3}	2.8×10^{-4}
0.9	8.5×10^{-2}	1.7×10^{-3}	2.7×10^{-3}	3.0×10^{-4}
1.0	9.7×10^{-2}	2.1×10^{-3}	3.2×10^{-3}	3.0×10^{-4}
1.1	1.1×10^{-1}	2.6×10^{-3}	3.6×10^{-3}	3.0×10^{-4}
1.2	1.2×10^{-1}	3.1×10^{-3}	4.2×10^{-3}	2.8×10^{-4}
1.3	1.4×10^{-1}	3.7×10^{-3}	4.7×10^{-3}	2.4×10^{-4}
1.4	1.5×10^{-1}	4.4×10^{-3}	5.3×10^{-3}	1.9×10^{-4}
1.5	1.7×10^{-1}	5.2×10^{-3}	5.9×10^{-3}	1.1×10^{-4}
1.6	1.8×10^{-1}	6.1×10^{-3}	6.6×10^{-3}	7.1×10^{-7}
1.7	2.0×10^{-1}	7.2×10^{-3}	7.3×10^{-3}	1.4×10^{-4}
1.8	2.1×10^{-1}	8.4×10^{-3}	8.1×10^{-3}	3.2×10^{-4}
1.9	2.3×10^{-1}	9.8×10^{-3}	8.9×10^{-3}	5.4×10^{-4}
2.0	2.4×10^{-1}	1.2×10^{-2}	9.8×10^{-3}	8.2×10^{-4}
l_2-error($E_{2,N}$)	0.60	0.022	0.022	0.0014

Example 3.(Burden and Paires[3])

$$\begin{cases} y'(t) &= 5e^{5t}(t-y)^2+1, \qquad 0 \le t \le 2 \\ y(0) &= -1 \end{cases}$$

whose exact solution is $y = t - e^{-5t}$.

This example includes the so-called stiff differential equation as the exact solution involves fast decaying transient term. It is well-known that most numerical methods with not very small step-sizes are unstable when they are directly applied to the stiff differential equation [3, 12].

Numerical errors, $|y_{i+1} - y(t_{i+1})|$ of the approximates y_{i+1} obtained by the proposed method (2.5), the standard Euler method and the Taylor method of order 2, for small number of nodes $N = 5, 10$, are given in Table 3. Contrary to the proposed method (2.5), the errors of the existing methods blow up for $N = 5$ and/or $N = 10$. This implies that the proposed method is useful even with a large step-size for the stiff equation. Numerical errors with $N = 20$ are given in Table 4, and it can be seen that the proposed method results in slightly better errors than the second-order Taylor method.

TABLE 3. Differences between the approximate solutions $\{y_j\}_{j=0}^N$ and the exact solutions $\{y(t_j)\}_{j=0}^N$ for Example 3 ($N = 5,\ 10$).

N	t_j	Existing methods		Presented method	
		Euler method	Taylor method of order 2	eqn. (2.5)	eqn. (3.3) (with $\delta = \delta^*$)
	0.0	0	0	0	0
	0.4	1.1×10^{0}	8.7×10^{-1}	2.4×10^{-1}	4.3×10^{-2}
5	0.8	1.6×10^{1}	1.9×10^{2}	7.1×10^{-2}	7.7×10^{-3}
	1.2	2.7×10^{4}	8.2×10^{10}	4.9×10^{-2}	1.1×10^{-3}
	1.6	6.0×10^{11}	3.5×10^{38}	4.6×10^{-2}	1.5×10^{-4}
	2.0	2.1×10^{27}	1.0×10^{123}	4.2×10^{-2}	2.1×10^{-5}
	0.0	0	0	0	0
	0.2	3.7×10^{-1}	1.3×10^{-1}	6.6×10^{-2}	2.4×10^{-3}
	0.4	1.4×10^{-1}	2.7×10^{-2}	2.4×10^{-2}	1.2×10^{-3}
	0.6	5.0×10^{-2}	2.2×10^{0}	9.0×10^{-3}	4.7×10^{-4}
	0.8	1.8×10^{-2}	4.2×10^{3}	3.3×10^{-3}	1.8×10^{-4}
10	1.0	6.7×10^{-3}	2.1×10^{14}	1.2×10^{-3}	6.6×10^{-5}
	1.2	2.5×10^{-3}	2.2×10^{47}	4.5×10^{-4}	2.5×10^{-5}
	1.4	9.1×10^{-4}	1.6×10^{147}	1.6×10^{-4}	9.0×10^{-6}
	1.6	3.4×10^{-4}	1.0×10^{123}	6.1×10^{-5}	3.3×10^{-6}
	1.8	1.2×10^{-4}	∞	2.2×10^{-5}	1.2×10^{-6}
	2.0	4.5×10^{-5}	∞	8.2×10^{-6}	4.5×10^{-7}
l_2-error($E_{2,N}$)		0.40	–	0.071	0.0027

TABLE 4. Differences between the approximate solutions $\{y_j\}_{j=0}^N$ and the exact solutions $\{y(t_j)\}_{j=0}^N$ for Example 3 ($N = 20$).

t_j	Existing methods		Presented method	
	Euler method	Taylor method of order 2	eqn. (2.5)	eqn. (3.3) (with $\delta = \delta^*$)
0.0	0	0	0	0
0.1	1.1×10^{-1}	1.9×10^{-2}	1.4×10^{-2}	2.5×10^{-4}
0.2	7.4×10^{-2}	2.1×10^{-2}	1.3×10^{-2}	2.4×10^{-4}
0.3	4.7×10^{-2}	1.7×10^{-2}	9.3×10^{-3}	1.8×10^{-4}
0.4	2.9×10^{-2}	1.3×10^{-2}	6.1×10^{-3}	1.2×10^{-4}
0.5	1.7×10^{-2}	9.1×10^{-3}	3.9×10^{-3}	7.7×10^{-5}
0.6	1.1×10^{-2}	6.2×10^{-3}	2.4×10^{-3}	4.8×10^{-5}
0.7	6.4×10^{-3}	4.2×10^{-3}	1.5×10^{-3}	3.0×10^{-5}
0.8	3.9×10^{-3}	2.7×10^{-3}	9.1×10^{-4}	1.8×10^{-5}
0.9	2.4×10^{-3}	1.8×10^{-3}	5.5×10^{-4}	1.1×10^{-5}
1.0	1.4×10^{-3}	1.1×10^{-3}	3.4×10^{-4}	6.8×10^{-6}
1.1	8.7×10^{-4}	7.3×10^{-4}	2.1×10^{-4}	4.1×10^{-6}
1.2	5.3×10^{-4}	4.6×10^{-4}	1.2×10^{-4}	2.5×10^{-6}
1.3	3.2×10^{-4}	2.9×10^{-4}	7.5×10^{-5}	1.5×10^{-6}
1.4	1.9×10^{-4}	1.8×10^{-4}	4.6×10^{-5}	9.3×10^{-7}
1.5	1.2×10^{-4}	1.2×10^{-4}	2.8×10^{-5}	5.6×10^{-7}
1.6	7.2×10^{-5}	7.2×10^{-5}	1.7×10^{-5}	3.4×10^{-7}
1.7	4.3×10^{-5}	4.5×10^{-5}	1.0×10^{-5}	2.1×10^{-7}
1.8	2.6×10^{-5}	2.8×10^{-5}	6.2×10^{-6}	1.3×10^{-7}
1.9	1.6×10^{-5}	1.7×10^{-5}	3.8×10^{-6}	7.6×10^{-8}
2.0	9.7×10^{-6}	1.1×10^{-5}	2.3×10^{-6}	4.6×10^{-8}
l_2-error($E_{2,N}$)	0.14	0.037	0.022	0.00042

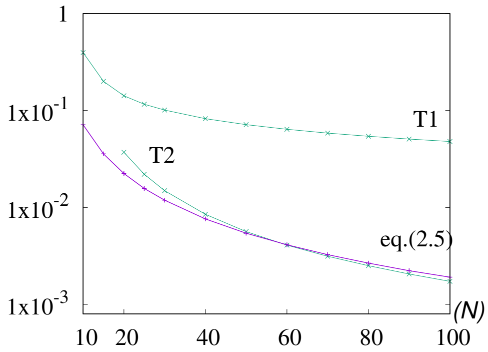
In addition, Figure 3 compares the l_2-norm error $E_{2,N}$ and the l_∞-norm error $E_{\infty,N}$ for the approximates $\{y_{i+1}\}_{i=0}^{N-1}$ obtained by the proposed methods (2.5), the original Euler method and the second-order Taylor method, with respect to the number of nodes $10 \le N \le 100$.

3. AN EXTENDED METHOD WITH WEIGHTED AVERAGES

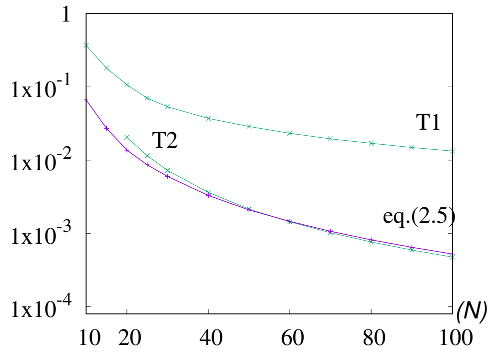
To extend the proposed method associated with the mean values $\overline{t}_i = (t_i + t_{i+1})/2$ and $\overline{w} = (w + x)/2$ in (2.3), for the approximations $w = y_i$ at t_i and $x = y_{i+1}$ at t_{i+1}, we set weighted averages as

$$t_i^{[\delta]} = \delta t_i + (1-\delta) t_{i+1}, \qquad w^{[\delta]} = \delta w + (1-\delta) x, \tag{3.1}$$

where $0 < \delta \le 1$.



(a) $E_{2,N}$



(b) $E_{\infty,N}$

FIGURE 3. Graphs of the l_2-norm errors, $E_{2,N}$ in (a) and the l_∞-norm errors, $E_{\infty,N}$ in (b), $10 \le N \le 100$, for Example 3.

Employing the Euler method and the weighted averages $t_i^{[\delta]}$ and $w^{[\delta]}$, with $t_{i+1} - t_i^{[\delta]} = \delta h$, we extend the implicit Eq.(2.4) to determine x as follows.

$$x = w^{[\delta]} + \delta h f(t_i^{[\delta]}, w^{[\delta]}).$$

That is, from the second equation in (3.1),

$$x = w + hf(t_i^{[\delta]}, w^{[\delta]}). \tag{3.2}$$

It is noted that the standard implicit Euler method (or backward Euler method),

$$x = w + hf(t_{i+1}, x)$$

is equivalent to (3.2) with $\delta = 0$, an exceptional case.

For solving the nonlinear equation numerically, if we define a function $F^{[\delta]}$ as

$$F^{[\delta]}(x) = x - w - hf(t_i^{[\delta]}, w^{[\delta]})$$

then the problem of finding x in the Eq.(3.2), based on the Newton method with an initial guess $x^{(0)} = w$, provides the following iterative method.

$$\begin{aligned} x^{(k)} &= x^{(k-1)} - \frac{F^{[\delta]}(x^{(k-1)})}{F^{[\delta]\prime}(x^{(k-1)})} \\ &= x^{(k-1)} - \frac{x^{(k-1)} - w - hf\left(t_i^{[\delta]}, \delta w + (1-\delta)x^{(k-1)}\right)}{1 - hf_x\left(t_i^{[\delta]}, \delta w + (1-\delta)x^{(k-1)}\right)}, \quad k = 1, 2, \ldots k_{\max}, \end{aligned} \tag{3.3}$$

where f_x is a partial derivative defined, for $\eta = w^{[\delta]}$, by

$$\begin{aligned} f_x\left(t_i^{[\delta]}, \eta\right) &:= \frac{\partial}{\partial x} f\left(t_i^{[\delta]}, w^{[\delta]}\right) \\ &= \frac{\partial}{\partial x} f\left(t_i^{[\delta]}, \delta w + (1-\delta)x\right) = (1-\delta)\left[\frac{\partial}{\partial \eta} f\left(t_i^{[\delta]}, \eta\right)\right]. \end{aligned}$$

When $\delta = 1$, we have $t_i^{[\delta]} = t_i$, $w^{[\delta]} = w$, and $f_x\left(t_i^{[\delta]}, \eta\right) \equiv 0$. Thus the formula (3.3) becomes

$$x^{(k)} = w + hf(t_i, w),$$

independently of $x^{(k-1)}$, which is equivalent to the original Euler method.

On the other hand, if we set $k_{\max} = 1$ with $x^{(0)} = w$ then from (3.3) we have a single-iteration

$$x = x^{(1)} = w + \frac{hf\left(t_i^{[\delta]}, w\right)}{1 - hf_x\left(t_i^{[\delta]}, w\right)}. \tag{3.4}$$

When $f(t, y)$ is a linear map of y, the iteration (3.3) is reduced to the single iteration formula (3.4).

The following theorem shows that convergence order of the proposed implicit method (3.2) becomes 2 as the parameter δ goes to $\frac{1}{2}$.

Theorem 3.1. *Suppose $f(t, y)$, in the initial value problem* (1.1)*, satisfies a Lipschitz condition in the variable y with a Lipschitz constant L on a set $D = \{(t, y) \mid a \le t \le b,\ -\infty < y < \infty\}$. Furthermore, let f be twice continuously differentiable in D and let the exact solution $y(t)$ satisfy*

$$|y''(t)| \le M, \qquad (t \in [a, b])$$

for some positive constant M. Then, for each $i = 0, 1, 2, \ldots, N-1$ the approximate solution $y_{i+1}\ (= x)$ to $y(t_{i+1})$ obtained by the formula (3.2) *with $0 < \delta \le 1$ satisfies*

$$|y(t_{i+1}) - y_{i+1}| \le \left(\frac{hM}{L}\left|\delta - \frac{1}{2}\right| + \frac{C}{L}h^2\right)\left\{e^{(i+1)hL} - 1\right\},$$

where C is a positive constant and the step size h is assumed to be small enough.

Proof. The exact solutions $y(t_i)$ and $y(t_{i+1})$ at the nodes $t = t_i$ and $t = t_{i+1}$, respectively, satisfy

$$y(t_{i+1}) = y(t_i) + hf(t_i, y(t_i)) + \frac{h^2}{2} f'(t_i, y(t_i)) + C_1 h^3 \tag{3.5}$$

for some constant C_1, where f' is a total derivative with respect to t such as

$$f'(t, y(t)) = f_t(t, y(t)) + f_y(t, y(t)) f(t, y(t)).$$

From the Eq.(3.2) for the approximates $y_i = w$ and $y_{i+1} = x$ to $y(t_i)$ and $y(t_{i+1})$, respectively, we have

$$y_{i+1} \;=\; y_i \;+\; hf\left(t_i^{[\delta]}, w^{[\delta]}\right). \tag{3.6}$$

Then, from (3.5) and (3.6),

$$y(t_{i+1}) - y_{i+1} = y\left(t_i\right) - y_i \;+\; h\left\{f\left(t_i, y(t_i)\right) - f\left(t_i^{[\delta]}, w^{[\delta]}\right)\right\} \;+\; \frac{h^2}{2} f'\left(t_i, y(t_i)\right) + C_1 h^3. \tag{3.7}$$

But, since

$$t_i^{[\delta]} = \delta t_i + (1-\delta) t_{i+1} = t_i + (1-\delta)h$$

and

$$w^{[\delta]} = \delta y_i + (1-\delta) y_{i+1} = y_i + (1-\delta) hf\left(t_i^{[\delta]}, w^{[\delta]}\right),$$

it follows that, by Taylor's theorem,

$$f\left(t_i^{[\delta]}, w^{[\delta]}\right) = f\left(t_i, y_i\right) + (1-\delta) h f_t(t_i, y_i) + (1-\delta) hf\left(t_i^{[\delta]}, w^{[\delta]}\right) f_y(t_i, y_i) + C_2 h^2.$$

for some constant C_2. That is,

$$f\left(t_i^{[\delta]}, w^{[\delta]}\right) = \frac{1}{1-(1-\delta) h f_y(t_i, y_i)} \Big\{ f\left(t_i, y_i\right) + (1-\delta) h f_t(t_i, y_i) \Big\} + C_2' h^2$$

for a constant $C_2' = C_2 / \left\{1 - (1-\delta) h f_y(t_i, y_i)\right\}$. Then we have

$$\begin{aligned}
& f\left(t_i, y(t_i)\right) - f\left(t_i^{[\delta]}, w^{[\delta]}\right) \\
&= \frac{1}{1-(1-\delta) h f_y(t_i, y_i)} \Big\{ \left[f\left(t_i, y(t_i)\right) - f\left(t_i, y_i\right)\right] \\
&\qquad\qquad - (1-\delta) h \left[f_y(t_i, y_i) f\left(t_i, y(t_i)\right) + f_t(t_i, y_i)\right] \Big\} - C_2' h^2 \\
&= \frac{1}{1-(1-\delta) h f_y(t_i, y_i)} \Big\{ \left[f\left(t_i, y(t_i)\right) - f\left(t_i, y_i\right)\right] - (1-\delta) h f'\left(t_i, y(t_i)\right) \Big\} - C_2'' h^2
\end{aligned}$$

for some constant C_2''. The last equality results from $f_y(t_i, y_i) f\left(t_i, y(t_i)\right) + f_t(t_i, y_i) = f_y(t_i, y(t_i)) f\left(t_i, y(t_i)\right) + f_t(t_i, y(t_i)) + O\left(h\right) = f'\left(t_i, y(t_i)\right) + O\left(h\right)$.

Therefore, from the Eq.(3.7)

$$\begin{aligned}
\left|y\left(t_{i+1}\right) - y_{i+1}\right| \leq \left|y\left(t_i\right) - y_i\right| \;+\; & \frac{h}{1-(1-\delta) h f_y(t_i, y_i)} \left|f\left(t_i, y(t_i)\right) - f\left(t_i, y_i\right)\right| \\
&+ \frac{h^2}{1-(1-\delta) h f_y(t_i, y_i)} \left| \left\{ \frac{1}{2} - \frac{(1-\delta)}{2} h f_y(t_i, y_i) - (1-\delta) \right\} f'\left(t_i, y(t_i)\right) \right| \\
&+ |C_1 - C_2''| h^3.
\end{aligned}$$

By the assumptions,

$$\left|f\left(t_i, y(t_i)\right) - f\left(t_i, y_i\right)\right| \leq L \left|y(t_i) - y_i\right|, \qquad \left|f'\left(t_i, y(t_i)\right)\right| = \left|y''\left(t_i\right)\right| \leq M$$

and the step size h is supposed to be small enough. Thus we have

$$\begin{aligned}|y(t_{i+1}) - y_{i+1}| &\leq (1+hL)\,|y(t_i) - y_i| \,+\, h^2 M \left|\delta - \frac{1}{2}\left\{1 + (1-\delta)hf_y(t_i, y_i)\right\}\right| + C_3 h^3 \\ &\leq (1+hL)\,|y(t_i) - y_i| \,+\, h^2 M \left|\delta - \frac{1}{2}\right| + Ch^3\end{aligned}$$

for some constants $C_3 > 0$ and $C = \frac{|1-\delta|}{2} M \|f_y\|_\infty + C_3$. Referring to Lemma 5.8 in [3], if we set $a_i = |y(t_i) - y_i|$ $(a_0 = 0)$ with $s = hL$ and $t = h^2 M \left|\delta - \frac{1}{2}\right| + Ch^3$, then

$$\frac{t}{s} = \frac{h}{L} M \left|\delta - \frac{1}{2}\right| \,+\, \frac{C}{L} h^2$$

and it follows that

$$|y(t_{i+1}) - y_{i+1}| \leq \frac{t}{s}\left\{e^{(i+1)hL} - 1\right\} = \left(\frac{hM}{L}\left|\delta - \frac{1}{2}\right| + \frac{C}{L}h^2\right)\left\{e^{(i+1)hL} - 1\right\}.$$

This completes the proof. □

In addition, to surmise whether a numerical method applied to the stiff differential equation can overcome the instability problem, we consider the *A-stability* introduced in the literature [10, 11]. The A-stability result of the proposed method is given in the following theorem.

Theorem 3.2. *The proposed method* (3.2) *is A-stable for every* $0 < \delta \leq \frac{1}{2}$.

Proof. According to the definition given in [10], the stability function ψ of the proposed method (3.2) is

$$\psi(z) = \frac{1 + \delta z}{1 - (1-\delta)z}\,, \qquad z \in \mathbb{C}.$$

Thus the region of absolute stability is

$$\begin{aligned}S &= \left\{ z \in \mathbb{C} \mid \; |\psi(z)| = \left|\frac{1+\delta z}{1-(1-\delta)z}\right| < 1 \right\} \\ &= \left\{ z \in \mathbb{C} \;\middle|\; \mathrm{Re}(z) < \left(\frac{1-2\delta}{2}\right)|z|^2 \right\}.\end{aligned}$$

For $\delta = \frac{1}{2}$, in particular,

$$S = \left\{ z \in \mathbb{C} \;\middle|\; \mathrm{Re}(z) < 0 \right\}.$$

When $\delta \neq \frac{1}{2}$, setting $z = x + yi$ $(x,\, y \in \mathbb{R})$ gives

$$S = \left\{ z \in \mathbb{C} \mid \; \left(x - \frac{1}{1-2\delta}\right)^2 + y^2 > \left(\frac{1}{1-2\delta}\right)^2 \right\} \tag{3.8}$$

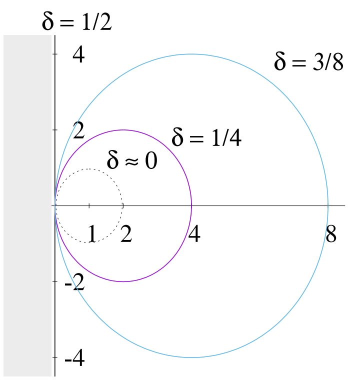
for $0 < \delta < \frac{1}{2}$, and

$$S = \left\{ z \in \mathbb{C} \mid \left(x - \frac{1}{1-2\delta} \right)^2 + y^2 < \left(\frac{1}{1-2\delta} \right)^2 \right\} \tag{3.9}$$

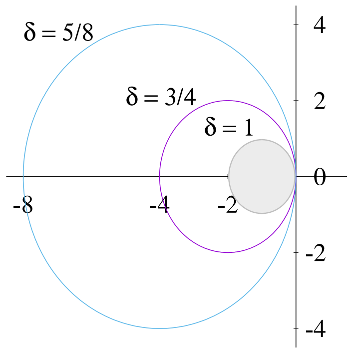
for $\frac{1}{2} < \delta < 1$.

Since for every $0 < \delta \leq \frac{1}{2}$ the stability region includes the left half complex plane, $\{z \in \mathbb{C} \mid \mathrm{Re}(z) < 0\}$ (See Figure 4(a)), the method (3.2) is A-stable. □

Figure 4, plotted by the formulas (3.8) and (3.9), illustrates the boundary curves of the stability regions of the proposed method (3.2) for some parameters $0 < \delta \leq \frac{1}{2}$ in (a) and $\frac{1}{2} < \delta \leq 1$ in (b). Each stability region means the outside of each corresponding curve in (a) and the inside of each corresponding curve in (b). The greyed out area indicates, for example, the case of $\delta = \frac{1}{2}$ in (a) and $\delta = 1$ in (b). In fact, the cases of $\delta = \frac{1}{2}$ and $\delta = 1$ correspond to the proposed method (2.4) and the standard Euler method, respectively.



(a) $0 < \delta \leq \frac{1}{2}$



(b) $\frac{1}{2} < \delta \leq 1$

FIGURE 4. Boundary curves of the stability regions of the method (3.2) with $0 < \delta \leq \frac{1}{2}$ in (a) and with $\frac{1}{2} < \delta \leq 1$ in (b).

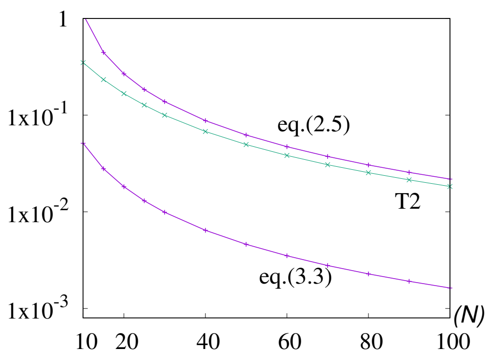
We recall the examples selected in the previous section(Example 1–Example 3). For $t_i^{[\delta]}$ and $w^{[\delta]} = \eta$ defined by (3.1) we have

$$f_x(t_i^{[\delta]}, \eta) = \begin{cases} 2(1-\delta)(1-t_i^{[\delta]})\eta, & \text{for Example 1} \\ 1-\delta, & \text{for Example 2} \\ -10(1-\delta)e^{5t_i^{[\delta]}}(t_i^{[\delta]}-\eta), & \text{for Example 3.} \end{cases}$$

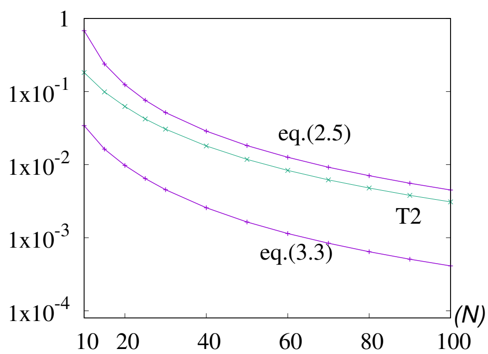
In numerical implementation of the proposed method we chose values of the parameter δ, for each $h = (b-a)/N$, by the formula below.

$$\delta^* = \begin{cases} \frac{1}{2} + \frac{h}{6}, & \text{for Example 1 and Example 2} \\ \frac{1}{2} - h, & \text{for Example 3} \end{cases}$$

which assures convergence order two, for h small enough, based on Theorem 3.1 and it also assures the A-stability for the stiff problem(Example 3) based on Theorem 3.2. Moreover, beyond expectations, the proposed method with $\delta = \delta^*$ results in the outstanding accuracy of the approximate solutions. In practice, last columns in Table 1–Table 4 include numerical errors of the proposed method (3.3) (or (3.4)) with $\delta = \delta^*$. In addition, Fig. 5–Fig. 7 show l_2–norm errors and l_∞–norm errors with respect to the number of nodes, $10 \le N \le 100$. It can be seen that the proposed method with $\delta = \delta^*$ gives a higher accuracy than the other compared second-order methods.



(a) $E_{2,N}$



(b) $E_{\infty,N}$

FIGURE 5. Graphs of the l_2-norm errors, $E_{2,N}$ in (a) and the l_∞-norm errors, $E_{\infty,N}$ in (b), $10 \le N \le 100$, for Example 1.

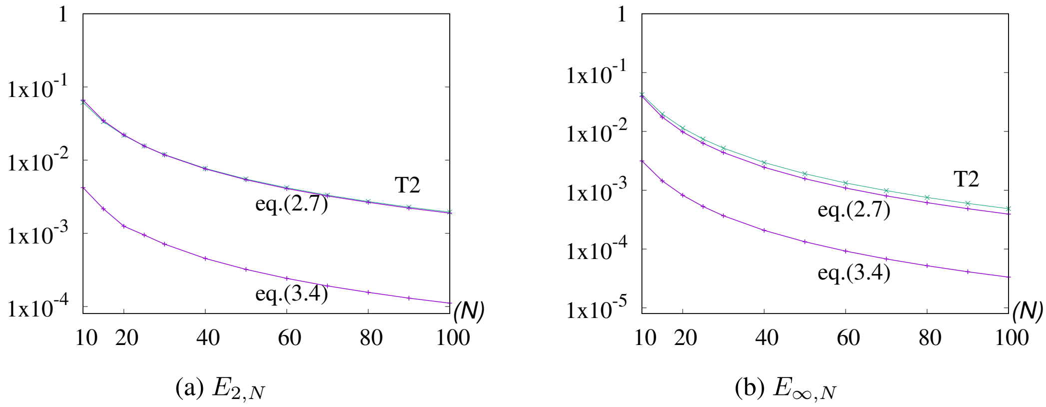


FIGURE 6. Graphs of the l_2-norm errors, $E_{2,N}$ in (a) and the l_∞-norm errors, $E_{\infty,N}$ in (b), $10 \leq N \leq 100$, for Example 2.

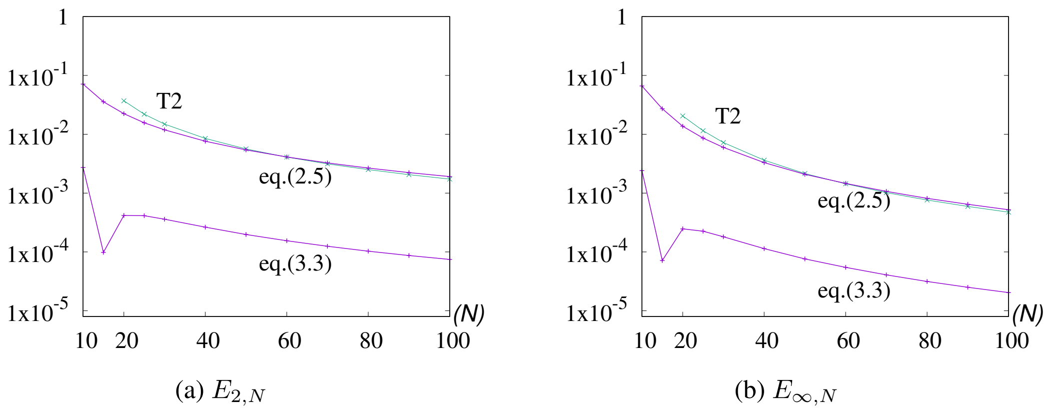


FIGURE 7. Graphs of the l_2-norm errors, $E_{2,N}$ in (a) and the l_∞-norm errors, $E_{\infty,N}$ in (b), $10 \leq N \leq 100$, for Example 3.

The following fixed-point iteration can be used instead of the iteration (3.3) associated with the Newton method to solve the proposed implicit Eq.(3.2).

$$x^{(k)} = w + hf\left(t_i^{[\delta]}, \delta w + (1-\delta)x^{(k-1)}\right), \qquad k = 1, 2, \dots k_{\max} \tag{3.10}$$

with $x^{(0)} = w$. Indeed, numerical experiments on Example 1 and Example 2 show that this iteration gives the same results as the iteration (3.3), whereas the former requires more repetitions than the latter to satisfy a given tolerance error. However, it can also be seen that, unlike (3.3), the iteration (3.10) is not successful in obtaining convergent solutions to the stiff problem like Example 3.

4. Conclusions

For numerical solutions of the initial value problem at equally spaced nodes, we proposed a single-step implicit method based on the Euler method. It is proved that the convergence order of the method is improved to 2. Numerical results for some selected examples illustrate that the proposed method has the similar error behavior to the second-order Taylor method. The proposed method is particularly effective for the stiff initial value problem even when the step size is not very small. Moreover, using weighted averages with a parameter $0 < \delta \le 1$, we extended the method. We have shown that for the parameter δ close enough to $\frac{1}{2}$ the extended method has convergence order two and it is A-stable for every $0 < \delta \le \frac{1}{2}$. It can be seen that the extended method with an appropriately chosen parameter δ near $\frac{1}{2}$ further improves approximation errors.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) [No. 2021R1F1A1047343].

References

[1] R. Kress, *Numerical Analysis*, Graduate Texts in Mathematics 181, Springer Verlag, New York, 1998.

[2] K.A. Atkinson, *An Introduction to Numerical Analysis, 2nd edition*, John Wiley & Sons, New York, 1989.

[3] R.L. Burden and J.D. Faires, *Numerical Analysis*, 6th Edition, Springer, New York, 1997.

[4] J.C. Butcher, *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons, New York, 2003.

[5] A.M. Wazwaz, *A new method for solving singular initial value problems in the second- order ordinary differential equations*, Applied Mathematics and Computations, **28** (2002), 45–57.

[6] K.H. Mohammad, S.A. Mohammad, A.H. Mohammad, A.S. Mohamma, and B.H. Mohammad, *An Implicit Method for Numerical Solution of Second Order Singular Initial Value Problems*, The Open Mathematics Journal, **7** (2014), 1–5.

[7] O. Koch, P. Kofler, and E.B. Weinmüller, *The implicit Euler method for the numerical solution of singular initial value problems*, Applied Numerical Mathematics, **34** (2000), 231—252.

[8] D.F. Griffiths and D.J. Higham, *Numerical methods for ordinary differential equations*, Springer, New York, 2010.

[9] A. Guzel and W. Layton, *Time filters increase accuracy of the fully implicit method*, BIT Numerical Mathematics, **58** (2018), 301—315.

[10] G.G. Dahlquist, *A special stability problem for linear multistep methods*, BIT Numerical Mathematics, **3** (1963), 27—43.
[11] G.G. Dahlquist, W. Liniger, and O. Nevanlinna, *Stability of two-step methods for variable integration steps*, SIAM Journal on Numerical Analysis, **20** (1983), 1071—1085.
[12] Wikipedia, *Stiff equation*, http://en.wikipedia.org/wiki/Stiff_equation# A-stability.