ROCKY MOUNTAIN
JOURNAL OF MATHEMATICS
Volume 38, Number 1, 2008

A FAMILY OF THE FUNCTIONAL EPSILON ALGORITHMS FOR ACCELERATING CONVERGENCE

R. THUKRAL

ABSTRACT. This paper introduces two new functional epsilon type algorithms, namely, the modified functional epsilon algorithm and the enhanced functional epsilon algorithm. The modified functional epsilon algorithm is based on the modified Padé approximant, and the enhanced functional epsilon algorithm is actually a modification of the improved functional epsilon algorithm. The use of the functional epsilon type algorithms for accelerating the convergence of sequence of functions is demonstrated. The estimates of the approximate solution derived using the modified functional epsilon algorithms are found to be more accurate than other similar algorithms.

1. Introduction. In this paper we introduce two new algorithms, namely, the modified functional epsilon algorithm and the enhanced functional epsilon algorithm for accelerating the convergence of sequence of functions. We examine the effectiveness of these new algorithms with the classical functional epsilon algorithm and the improved functional epsilon algorithm by showing the accuracy of the approximate solution. The modified functional epsilon algorithm is based on the modified Padé approximant. The formula for the modified functional epsilon algorithm is actually derived from the second row sequence of the modified Padé approximant.

The prime motive for the development of these new functional epsilon type algorithms was to establish that there exists a better accelerator than the classical functional epsilon algorithms and the improved functional epsilon algorithm. We begin by describing the fundamentals of each of the algorithms. In order to evaluate these functional epsilon type algorithms we use the following proposition.

2000 AMS *Mathematics subject classification.* Primary 65D15.
Keywords and phrases. Modified functional epsilon algorithm, enhanced functional epsilon algorithm, improved functional epsilon algorithm, classical functional epsilon algorithm, convergence acceleration, Neumann series.
Received by the editors on October 28, 2004, and in revised form on November 9, 2005.

DOI:10.1216/RMJ-2008-38-1-291

Proposition. *The functional sequence used in the algorithms is based on the generating function $f(x, \lambda)$, which is a series of functions expressed as*

$$f(x,\lambda) = \sum_{i=0}^{\infty} C_i(x)\lambda^i, \tag{1}$$

in which $C_i(x) \in L_2[a, b]$ are given and $[a, b]$ is the domain of definition of $C_i(x)$ in some natural sense. We also suppose that $f(x, \lambda)$ is holomorphic as a function of λ at the origin $\lambda = 0$. Then (1) converges for values of $|\lambda|$ which are sufficiently small. In this paper, we see how the modified functional epsilon algorithm and enhanced functional epsilon algorithm can be used to accelerate the convergence of a series having the form (1) for $\lambda = 1$.

1.1 The modified functional epsilon algorithm (MFEA). To show how we have designed the modified functional epsilon algorithm, we briefly review the essentials definitions of the modified Padé approximant.

A modified Padé approximant of type (n, k) for the given power series (1) is the rational function

$$r(x,\lambda) = A(x,\lambda) \div B(\lambda), \tag{2}$$

where $A(x, \lambda), B(\lambda)$ are polynomials in λ, $A(x, \lambda) \in L_2[a, b]$ as a function of x and

$$\begin{aligned} &\text{(i)} \quad \partial\{A\} \le n, \quad \partial\{B\} \le k, \\ &\text{(ii)} \quad B(0) \ne 0, \\ &\text{(iii)} \quad A(x,\lambda) - B(\lambda) f(x,\lambda) = 0(\lambda^{n+1}). \end{aligned} \tag{3}$$

The definition of the numerator polynomial of the modified Padé approximant of type (n, k) as

$$A(x,\lambda) = \begin{vmatrix} \int_a^b C_{n-k}(x)\,dx & \int_a^b C_{n-k+1}(x)\,dx & \cdots & \int_a^b C_n(x)\,dx \\ \int_a^b C_{n-k+1}(x)\,dx & \int_a^b C_{n-k+2}(x)\,dx & \cdots & \int_a^b C_{n+1}(x)\,dx \\ \vdots & \vdots & & \vdots \\ \sum_{i=0}^{n-k} C_i(x)\lambda^k & \sum_{i=0}^{n-k+1} C_i(x)\lambda^{k-1} & \cdots & \sum_{i=0}^{n} C_i(x)\lambda^i \end{vmatrix} \tag{4}$$

and appropriately the denominator polynomial of the modified Padé approximant of type (n, k) is given as

(5)

$$B(\lambda) = \begin{vmatrix} \int_a^b C_{n-k}(x)\,dx & \int_a^b C_{n-k+1}(x)\,dx & \cdots & \int_a^b C_n(x)\,dx \\ \int_a^b C_{n-k+1}(x)\,dx & \int_a^b C_{n-k+2}(x)\,dx & \cdots & \int_a^b C_{n+1}(x)\,dx \\ \vdots & \vdots & & \vdots \\ \lambda^k & \lambda^{k-1} & \cdots & 1 \end{vmatrix}$$

provided $B(0) \neq 0$ and $C_i(x)$ are the coefficients of (1).

For the purpose of this paper, we use the modified Padé approximant of type $(n, 2)$, which is given by

(6)

$$r(x,1) = \frac{A(x,1)}{B(1)} = \frac{\begin{vmatrix} \int_a^b C_{n-2}(x)\,dx & \int_a^b C_{n-1}(x)\,dx & \int_a^b C_n(x)\,dx \\ \int_a^b C_{n-1}(x)\,dx & \int_a^b C_n(x)\,dx & \int_a^b C_{n+1}(x)\,dx \\ \sum_{i=0}^{n-2} C_i(x) & \sum_{i=0}^{n-1} C_i(x) & \sum_{i=0}^{n} C_i(x) \end{vmatrix}}{\begin{vmatrix} \int_a^b C_{n-2}(x)\,dx & \int_a^b C_{n-1}(x)\,dx & \int_a^b C_n(x)\,dx \\ \int_a^b C_{n-1}(x)\,dx & \int_a^b C_n(x)\,dx & \int_a^b C_{n+1}(x)\,dx \\ 1 & 1 & 1 \end{vmatrix}}$$

and form a new epsilon type algorithm.

Expanding the determinants of (6), the numerator polynomial of the modified Padé approximant is given as

(7)

$$\begin{aligned} A(x,1) &= \sum_{i=0}^{n-2} C_i(x)\left[\int_a^b C_{n-1}(x)\,dx \int_a^b C_{n+1}(x)\,dx - \left(\int_a^b C_n(x)\,dx\right)^2\right] \\ &\quad - \sum_{i=0}^{n-1} C_i(x)\left[\int_a^b C_{n-2}(x)\,dx \int_a^b C_{n+1}(x)\,dx \right. \\ &\qquad\qquad \left. - \int_a^b C_{n-1}(x)\,dx \int_a^b C_n(x)\,dx\right] \\ &\quad + \sum_{i=0}^{n} C_i(x)\left[\int_a^b C_{n-2}(x)\,dx \int_a^b C_n(x)\,dx - \left(\int_a^b C_{n-1}(x)\,dx\right)^2\right] \end{aligned}$$

and the denominator polynomial as

$$
\begin{aligned}
(8)\quad & B(1)\\
&= \left[\int_a^b C_{n-1}(x)\,dx \int_a^b C_{n+1}(x)\,dx - \left(\int_a^b C_n(x)\,dx \right)^2 \right]\\
&\quad - \left[\int_a^b C_{n-2}(x)\,dx \int_a^b C_{n+1}(x)\,dx - \int_a^b C_{n-1}(x)\,dx \int_a^b C_n(x)\,dx \right]\\
&\quad + \left[\int_a^b C_{n-2}(x)\,dx \int_a^b C_n(x)\,dx - \left(\int_a^b C_{n-1}(x)\,dx \right)^2 \right].
\end{aligned}
$$

We first initialize the essentials for all the functional epsilon type algorithms

$$
(9)\quad \varepsilon_{-1}^n(x) = 0, \qquad \varepsilon_0^n(x) = \sum_{i=0}^n C_i(x), \qquad \Delta\varepsilon_k^n(x) = \varepsilon_k^{n+1}(x) - \varepsilon_k^n(x)
$$

for $n, k \in \mathbf{N}$, where Δ operates, now and in sequel, on the variable n.

We shall convert the numerator polynomial expression (7) into an appropriate functional epsilon algorithm; thus, (7) becomes

$$
\begin{aligned}
(10)\quad & A(x,1)\\
&= \varepsilon_0^{n-2}(x) \left[\int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \int_a^b \Delta\varepsilon_0^n(x)\,dx - \left(\int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx \right)^2 \right]\\
&\quad - \varepsilon_0^{n-1}(x) \left[\int_a^b \Delta\varepsilon_0^{n-3}(x)\,dx \int_a^b \Delta\varepsilon_0^n(x)\,dx \right.\\
&\qquad\qquad \left. - \int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx \right]\\
&\quad + \varepsilon_0^n(x) \left[\int_a^b \Delta\varepsilon_0^{n-3}(x)\,dx \int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx - \left(\int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \right)^2 \right]
\end{aligned}
$$

and similarly (8) becomes

$$
\begin{aligned}
(11)\quad & B(1) \\
&= \left[\int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \int_a^b \Delta\varepsilon_0^{n}(x)\,dx - \left(\int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx \right)^2 \right] \\
&\quad - \left[\int_a^b \Delta\varepsilon_0^{n-3}(x)\,dx \int_a^b \Delta\varepsilon_0^{n}(x)\,dx \right. \\
&\qquad \left. - \int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx \right] \\
&\quad + \left[\int_a^b \Delta\varepsilon_0^{n-3}(x)\,dx \int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx - \left(\int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \right)^2 \right].
\end{aligned}
$$

The subsequent expressions are simplified by using the following notation

$$E_n = \int_a^b \Delta\varepsilon_0^{n}(x)\,dx. \tag{12}$$

We begin with the denominator polynomial; the expression (11) is simplified further by pairing various terms. The first pair is given in the lefthand side and we replace it with the expression on the righthand side,

$$-[E_{n-1}^2 + E_{n-2}^2] = -[E_{n-1} - E_{n-2}]^2 - 2E_{n-1}E_{n-2}. \tag{13}$$

We actually consider the expression on the righthand side of (13) in two parts, thus

$$-[E_{n-1} - E_{n-2}]^2 = -[\Delta E_{n-2}]^2 \tag{14}$$

and

$$-2E_{n-1}E_{n-2}. \tag{15}$$

Furthermore, the expression (15) is added to the last term of the middle row of (11) and we are left with

$$-E_{n-1}E_{n-2}. \tag{16}$$

The next pair we consider of (11) is on the lefthand side and simplified on righthand side

$$E_{n-2}E_n - E_{n-3}E_n = E_n \Delta E_{n-3}. \tag{17}$$

The last pair is considered to be formed by the remaining term of (13) and is the resultant term (16)

$$-E_{n-2}E_{n-1} + E_{n-3}E_{n-1} = -E_{n-1}\Delta E_{n-3}. \tag{18}$$

Collecting (14), (17) and (18) and using the original notation the most simplified expression for the denominator polynomial of the modified Padé approximant is given as

$$\begin{aligned} B(1) = & \left[\int_a^b (\Delta^2 \varepsilon_0^{n-3}(x))\,dx\right]\left[\int_a^b \Delta^2 \varepsilon_0^{n-1}(x)\,dx\right] \\ & - \left[\int_a^b (\Delta^2 \varepsilon_0^{n-2}(x))\,dx\right]^2. \end{aligned} \tag{19}$$

We progress to convert the numerator polynomial of the modified Padé approximant (10). In order to obtain an appropriate expression for the numerator polynomial we must insert two components of

$$\varepsilon_0^{n-1}(x)[E_{n-2}E_n - E_{n-1}^2] \quad \text{and} \quad \varepsilon_0^{n-1}(x)[E_{n-3}E_{n-1} - E_{n-2}^2]. \tag{20}$$

Therefore the expression (12) becomes

$$\begin{aligned} A(x,1) = & \varepsilon_0^{n-2}(x)[E_{n-2}E_n - E_{n-1}^2] - \varepsilon_0^{n-1}(x)[E_{n-3}E_n - E_{n-2}E_{n-1}] \\ & + \varepsilon_0^{n}(x)[E_{n-3}E_{n-1} - E_{n-2}^2] + \varepsilon_0^{n-1}(x)[E_{n-2}E_n - E_{n-1}^2] \\ & - \varepsilon_0^{n-1}(x)[E_{n-2}E_n - E_{n-1}^2] + \varepsilon_0^{n-1}(x)[E_{n-3}E_{n-1} - E_{n-2}^2] \\ & - \varepsilon_0^{n-1}(x)[E_{n-3}E_{n-1} - E_{n-2}^2]. \end{aligned} \tag{21}$$

Collecting and simplifying appropriate terms of (19) and (21), we obtain the expression for the modified functional epsilon algorithm of type $(n, 1)$ as

(22)

$$\frac{A(x,1)}{B(1)} = \varepsilon_0^{n-1}(x)$$
$$+ \frac{\Delta\varepsilon_0^{n-1}(x)\left[\int_a^b \Delta\varepsilon_0^{n-3}(x)\,dx \int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx - \left(\int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx\right)^2\right]}{\left[\int_a^b (\Delta^2\varepsilon_0^{n-3}(x))\,dx\right]\left[\int_a^b \Delta^2\varepsilon_0^{n-1}(x)\,dx\right] - \left[\int_a^b (\Delta^2\varepsilon_0^{n-2}(x))\,dx\right]}$$
$$\frac{\Delta\varepsilon_0^{n-2}(x)\left[\int_a^b \Delta\varepsilon_0^{n-2}(x)\,dx \int_a^b \Delta\varepsilon_0^{n}(x)\,dx - \left(\int_a^b \Delta\varepsilon_0^{n-1}(x)\,dx\right)^2\right]}{\left[\int_a^b (\Delta^2\varepsilon_0^{n-3}(x))\,dx\right]\left[\int_a^b \Delta^2\varepsilon_0^{n-1}(x)\,dx\right] - \left[\int_a^b (\Delta^2\varepsilon_0^{n-2}(x))\,dx\right]^2}$$

for $n, k \in \mathbf{N}$.

The iterative formula for the modified functional epsilon algorithm of type (n, k) is expressed as

(23)

$$\varepsilon_k^n(x) = \varepsilon_{k-1}^{n-1}(x)$$
$$+ \frac{\Delta\varepsilon_{k-1}^{n-1}(x)\left[\int_a^b \Delta\varepsilon_{k-1}^{n-3}(x)\,dx \int_a^b \Delta\varepsilon_{k-1}^{n-1}(x)\,dx - \left(\int_a^b \Delta\varepsilon_{k-1}^{n-2}(x)\,dx\right)^2\right]}{\left[\int_a^b (\Delta^2\varepsilon_{k-1}^{n-3}(x))\,dx\right]\left[\int_a^b \Delta^2\varepsilon_{k-1}^{n-1}(x)\,dx\right] - \left[\int_a^b (\Delta^2\varepsilon_{k-1}^{n-2}(x))\,dx\right]^2}$$
$$- \frac{\Delta\varepsilon_{k-1}^{n-2}(x)\left[\int_a^b \Delta\varepsilon_{k-1}^{n-2}(x)\,dx \int_a^b \Delta\varepsilon_{k-1}^{n}(x)\,dx - \left(\int_a^b \Delta\varepsilon_{k-1}^{n-1}(x)\,dx\right)^2\right]}{\left[\int_a^b (\Delta^2\varepsilon_{k-1}^{n-3}(x))\,dx\right]\left[\int_a^b \Delta^2\varepsilon_{k-1}^{n-1}(x)\,dx\right] - \left[\int_a^b (\Delta^2\varepsilon_{k-1}^{n-2}(x))\,dx\right]^2}$$

for $n, k \in \mathbf{N}$.

We shall demonstrate the performance of the modified functional epsilon algorithm given by (23) in Section 4.

1.2 The enhanced functional epsilon algorithm (EFEA). The enhanced functional epsilon algorithm is actually a modified version of the improved functional epsilon algorithm. In order to calculate the new estimates we use the initial conditions given in (9), and the formula of the enhanced functional epsilon algorithm is

(24)

$$\varepsilon_k^n(x) = \varepsilon_{k-1}^n(x) - \frac{\Delta\varepsilon_{k-1}^n(x)\int_a^b \Delta\varepsilon_{k-1}^n(x)\Delta^2\varepsilon_{k-1}^{n+1}(x)\,dx}{\int_a^b [\Delta\varepsilon_{k-1}^{n+2}(x)\Delta^2\varepsilon_{k-1}^n(x) - \Delta\varepsilon_{k-1}^n(x)\Delta^2\varepsilon_{k-1}^{n+1}(x)]\,dx}$$

for $n, k \in \mathbf{N}$.

The entries of $\varepsilon_k^n(x)$ are displayed in the following functional epsilon table, Figure 1.

$$
\begin{array}{cccccc}
\varepsilon_{-1}^{-1}(x)=0 & & & & & \\
 & \varepsilon_0^{-1}(x)=0 & & & & \\
\varepsilon_{-1}^{0}(x)=0 & & \varepsilon_1^{-1}(x)=0 & & & \\
 & \varepsilon_0^{0}(x) & & \varepsilon_2^{-1}(x)=0 & & \\
\varepsilon_{-1}^{1}(x)=0 & & \varepsilon_1^{0}(x) & & \varepsilon_3^{-1}(x)=0 & \\
 & \varepsilon_0^{1}(x) & & \varepsilon_2^{0}(x) & & \ddots \\
\varepsilon_{-1}^{2}(x)=0 & & \varepsilon_1^{1}(x) & & \varepsilon_3^{0}(x) & \\
 & \varepsilon_0^{2}(x) & & \varepsilon_2^{1}(x) & \vdots & \ddots \\
\varepsilon_{-1}^{3}(x)=0 & & \varepsilon_1^{2}(x) & \vdots & & \\
\vdots & \varepsilon_0^{3}(x) & \vdots & & & \\
 & \vdots & & & &
\end{array}
$$

FIGURE 1. The modified functional epsilon table.

The structure of this paper is as follows. In Sections 2 and 3 we review two particular functional epsilon type algorithms, namely, the classical functional epsilon algorithm and the improved functional epsilon algorithm, respectively. In Section 4 we demonstrate the performance of the modified functional epsilon algorithm and enhanced functional epsilon algorithm with the corresponding classical functional epsilon algorithm and improved functional epsilon algorithm for two types of row sequence. The effectiveness of these new algorithms for accelerating the convergence of a sequence of functions has been investigated in the context of the Neumann series of a linear Fredholm integral equation. In our findings the modified functional epsilon algorithm is proved to be effective as an algorithm.

2. The classical functional epsilon algorithm (CFEA). The classical functional epsilon algorithm is well established [**1, 2, 8**]; thus, we shall state the essential formula used in calculating the approximate solution.

We use the initialized estimates given by (9), then we usually calculate the new estimates by

$$\varepsilon_k^n(x) = \varepsilon_{k-2}^n(x) - \frac{[\varepsilon_{k-1}^{n+1}(x) - \varepsilon_{k-1}^n(x)]}{\int_a^b [\varepsilon_{k-1}^{n+1}(x) - \varepsilon_{k-1}^n(x)]^2\,dx} \tag{25}$$

for $n, k \in \mathbf{N}$ and the initial estimates are given by (9).

In difference operator form

$$\varepsilon_k^n(x) = \varepsilon_{k-2}^n(x) - \frac{\Delta\varepsilon_{k-1}^n(x)}{\int_a^b [\Delta\varepsilon_{k-1}^n(x)]^2\,dx}. \tag{26}$$

3. The improved functional epsilon algorithm (IFEA). The improved functional epsilon algorithm is actually based on the integral Padé approximant [**5**–**10**]. The improved functional epsilon algorithm is very efficient as a method and has many advantages compared to the classical functional epsilon algorithm [**8**]. We shall state the essential formula used in calculating the approximate solution

$$\begin{aligned}\varepsilon_k^n(x) &= \varepsilon_{k-1}^{n+1}(x)\\ &\quad - \left[\frac{(\varepsilon_{k-1}^{n+1}(x) - \varepsilon_{k-1}^n(x)) \int_a^b [(\varepsilon_{k-1}^{n+1}(x) - \varepsilon_{k-1}^n(x))(\varepsilon_{k-1}^n(x) - \varepsilon_{k-1}^{n-1}(x))]\,dx}{\int_a^b [(\varepsilon_{k-1}^{n+1}(x) - \varepsilon_{k-1}^n(x))\{(\varepsilon_{k-1}^{n+1}(x) - \varepsilon_{k-1}^n(x)) - (\varepsilon_{k-1}^n(x) - \varepsilon_{k-1}^{n-1}(x))\}]\,dx}\right]\end{aligned} \tag{27}$$

for $n, k \in \mathbf{N}$ and in difference operator form

$$\varepsilon_k^n(x) = \varepsilon_{k-1}^{n+1}(x) - \left[\frac{\Delta\varepsilon_{k-1}^n(x) \int_a^b \Delta\varepsilon_{k-1}^n(x)\Delta\varepsilon_{k-1}^{n-1}(x)\,ds}{\int_a^b \Delta\varepsilon_{k-1}^n(x)\Delta^2\varepsilon_{k-1}^{n-1}(x)\,dx}\right]. \tag{28}$$

4. Applications of the epsilon type algorithms. To demonstrate the performance of each of the algorithms, we take two familiar linear Fredholm integral equations of the second kind. We determine the consistency and stability of the results by examining the convergence of each of the algorithms for two particular type of row sequence. The findings are generalized by illustrating the effectiveness of these algorithms for determining the approximate solution of two linear integral equations. Consequently, we shall demonstrate the efficiency of the modified functional epsilon algorithm and the enhanced functional epsilon algorithm by showing the error obtained by each of the algorithms. We illustrate the convergence of the methods described by making two distinct comparisons of the estimates based on two particular types of row sequence. In each case, the comparisons with other algorithms were made using a similar amount of data, that is, using the same number of terms of the Neumann series.

4.1 Numerical example 1. We investigate the convergence of functional sequences of the Neumann series solution of the linear integral equation. We shall consider the linear Fredholm integral equation of the form

$$f(x,\lambda) = g(x) + \lambda \int_0^1 k(x,y) f(y,\lambda)\, dy, \tag{29}$$

where

$$g(x) = x \quad \text{and} \quad k(x,y) = \begin{cases} (y-3)(1+x) & 0 \le y \le x \le 1, \\ (x-3)(1+y) & 0 \le x \le y \le 1. \end{cases}$$

This integral equation is a linear inhomogeneous Fredholm of the second kind with a nondegenerate kernel. The analytic solution of (29) is given by

$$f(x,\lambda) = \frac{3[\sinh(\omega x) + \omega\cosh(\omega x)] + \sinh(\omega w - \omega) - 2\omega\cosh(\omega x - \omega)}{(1+2\omega^2)\sinh(\omega) - 3\omega\cosh(\omega)} \tag{30}$$

where $\omega = 2\sqrt{\lambda}$. For a particular value $\lambda = 1$ the analytic solution (30) in the power series is

$$\begin{aligned} f(x,1) = {} & -0.229569 + 0.770431x - 0.459138x^2 + 0.513621x^3 \\ & - 0.153046x^4 \cdots, \end{aligned} \tag{31}$$

and the Neumann series solution of (29) is

$$\begin{aligned} f(x,\lambda) = \sum_{i=0}^{\infty} C_i(x)\lambda^i = {} & x + \left(\frac{2}{3}x^3 - \frac{7}{6}x - \frac{7}{6}\right)\lambda \\ & + \left(\frac{2}{15}x^5 - \frac{7}{9}x^2 + \frac{176}{45}x + \frac{176}{45}\right)\lambda^2 + \cdots. \end{aligned} \tag{32}$$

In Table 1 we show the errors incurred by the modified functional epsilon algorithm of type $(2, 1)$, the enhanced functional epsilon algorithm of type $(1, 1)$, the improved functional epsilon algorithm of type $(2, 1)$, the functional epsilon algorithm of type $(1, 2)$ for $x = 0(0.25)1$. We see a remarkable precision of the modified functional epsilon algorithm when compared to the other algorithms.

TABLE 1. Errors occurring in the solution of (29) using the modified functional epsilon algorithm, the enhanced functional epsilon algorithm, the classical functional epsilon algorithm and the improved functional epsilon algorithm.

x	MFEA1	EFEA1	IFEA1	CFEA1
0.0	0.441(-4)	0.547(-2)	0.547(-2)	0.117(-1)
0.25	0.317(-4)	0.465(-2)	0.465(-2)	0.991(-2)
0.50	0.258(-4)	0.397(-3)	0.403(-3)	0.901(-3)
0.75	0.290(-4)	0.416(-2)	0.416(-2)	0.891(-2)
1.0	0.946(-4)	0.565(-2)	0.564(-2)	0.123(-1)

The approximate solutions used in calculating the errors displayed in Table 1 are

(33)
$$MFEA1 = -0.22953 + 0.77047x - 0.45976x^2 + 0.51341x^3 - 0.14373x^4 + 0.08013x^5 + 0.00235x^7$$

(34)
$$EFEA1 = -0.23504 + 0.76496x - 0.42465x^2 + 0.52512x^3 - 0.18978x^4 + 0.09538x^5 + 0.00310x^7$$

(35)
$$IFEA1 = -0.23504 + 0.76496x - 0.42464x^2 + 0.52512x^3 - 0.18978x^4 + 0.09538x^5 + 0.00310x^7$$

(36)
$$CFEA1 = -0.21788 + 0.78212x - 0.53404x^2 + 0.48865x^3 - 0.4631x^4 + 0.047868x^5 + 0.00076x^7.$$

In Table 2 we show the errors incurred by the modified functional epsilon algorithm of type $(3, 2)$, the enhanced functional epsilon algorithm of type $(1,2)$, the improved functional epsilon algorithm of type $(3,2)$, the functional epsilon algorithm of type $(1,4)$ for $x = 0(0.25)1$. We see a remarkable precision of the modified functional epsilon algorithm when compared to the other algorithms.

The approximate solutions used in calculating the errors displayed in Table 2 are

(37)
$$\begin{aligned} MFEA2 = & -0.22957 + 0.77043x - 0.45914x^2 + 0.51362x^3 - 0.15303x^4 \\ & + 0.10265x^5 - 0.02021x^6 + 0.00951x^7 - 0.00121x^8 \\ & + 0.389(-3)x^9 + 0.418(-5)x^{11} \end{aligned}$$

(38)
$$\begin{aligned} EFEA2 = & -0.22957 + 0.77043x - 0.45914x^2 + 0.51362x^3 - 0.15305x^4 \\ & + 0.10272x^5 - 0.02042x^6 + 0.00978x^7 - 0.00137x^8 \\ & + 0.425(-3)x^9 + 0.476(-5)x^{11} \end{aligned}$$

(39)
$$\begin{aligned} IFEA2 = & -0.22957 + 0.77043x - 0.45914x^2 + 0.51362x^3 - 0.15305x^4 \\ & + 0.10272x^5 - 0.02042x^6 + 0.00978x^7 - 0.00137x^8 \\ & + 0.425(-3)x^9 + 0.476(-5)x^{11} \end{aligned}$$

(40)
$$\begin{aligned} CFEA2 = & -0.22957 + 0.77043x - 0.4592x^2 + 0.5136x^3 - 0.15149x^4 \\ & + 0.09863x^5 - 0.01595x^6 + 0.00708x^7 - 0.255(-3)x^8 \\ & + 0.161(-3)x^9 + 0.881(-6)x^{11}. \end{aligned}$$

TABLE 2. Errors occurring in the solution of (29) using the modified functional epsilon algorithm, the enhanced functional epsilon algorithm, the classical functional epsilon algorithm and the improved functional epsilon algorithm.

x	MFEA2	EFEA2	IFEA2	CFEA2
0.0	0.213(-8)	0.163(-6)	0.136(-6)	0.451(-6)
0.25	0.184(-8)	0.229(-7)	0.101(-7)	0.178(-5)
0.50	0.203(-8)	0.222(-6)	0.258(-6)	0.272(-5)
0.75	0.953(-9)	0.336(-7)	0.695(-7)	0.174(-5)
1.0	0.424(-8)	0.219(-6)	0.186(-6)	0.550(-5)

4.2 Numerical example 2. We investigate the convergence of functional sequences of the Neumann series solution of the linear integral equation. We shall consider the linear Fredholm integral equation of the form

$$f(x,\lambda) = g(x) + \lambda \int_0^1 k(x,y) f(y,\lambda)\, dy, \tag{41}$$

where

$$g(x) = 1 \quad \text{and} \quad k(x,y) = \begin{cases} 1+x-y & 0 \le y \le x \le 1, \\ 1+y-x & 0 \le x \le y \le 1. \end{cases}$$

This integral equation is also a linear inhomogeneous Fredholm of the second kind with a nondegenerate kernel, and the explicit solution of (41) is given by

$$f(x,\lambda) = \frac{2\cosh(\omega x - 2^{-1}\omega)}{2\cosh(2^{-1}\omega) - 3\omega \sinh(2^{-1}\omega)} \tag{42}$$

where $\omega = \sqrt{2\lambda}$. For a particular value $\lambda = 1$ the analytic solution (42) in power series is

$$\begin{aligned} f(x,\lambda) = &-3.429523 + 2.953015x - 3.429523x^2 + 0.984338x^3 \\ &- 0.571587x^4 + \cdots . \end{aligned} \tag{43}$$

It is familiar that the Neumann series of (41) converges [**3, 4**] and the first few terms of this series are

$$\begin{aligned} f(x,\lambda) = \sum_{i=0}^{\infty} C_i(x)\lambda^i &= 1 + \left(\frac{3}{2} - x + x^2\right)\lambda \\ &+ \left(2 - \frac{4}{3}x + \frac{3}{2}x^2 - \frac{1}{3}x^3 + \frac{1}{6}x^4\right)\lambda^2 + \cdots . \end{aligned} \tag{44}$$

In Table 3 we show the errors incurred by the modified functional epsilon algorithm of type $(3,1)$, the enhanced functional epsilon algorithm of type $(2,1)$, the improved functional epsilon algorithm of type $(3,1)$, the functional epsilon algorithm of type $(2,2)$ for $x = 0(0.25)1$. We see a remarkable precision of the modified functional epsilon algorithm when compared to the other algorithms.

The approximate solutions used in calculating the errors displayed in Table 3 are

(45)
$$MFEA1 = -3.4295 + 2.9530x - 3.4295x^2 + 0.9843x^3 - 0.5716x^4 + 0.09844x^5 - 0.03802x^6 + 0.004463x^7 - 0.001116x^8$$

(46)
$$EFEA1 = -3.4295 + 2.9530x - 3.4296x^2 + 0.9844x^3 - 0.5714x^4 + 0.09841x^5 - 0.03829x^6 + 0.004705x^7 - 0.001176x^8$$

(47)
$$IFEA1 = -3.4295 + 2.9530x - 3.4296x^2 + 0.9844x^3 - 0.5714x^4 + 0.09841x^5 - 0.03829x^6 + 0.004705x^7 - 0.001176x^8$$

(48)
$$CFEA1 = -3.4297 + 2.9531x - 3.4259x^2 + 0.9835x^3 - 0.5832x^4 + 0.1x^5 - 0.01706x^6 + 0.01395x^7 - 0.003487x^8.$$

TABLE 3. Errors occurring in the solution of (41) using the modified functional epsilon algorithm, the enhanced functional epsilon algorithm, the classical functional epsilon algorithm and the improved functional epsilon algorithm.

x	MFEA1	EFEA1	IFEA1	CFEA1
0.0	0.955(-9)	0.254554(-5)	0.254563(-5)	0.194(-3)
0.25	0.492(-8)	0.141971(-6)	0.141892(-6)	0.112(-4)
0.50	0.116(-7)	0.255722(-5)	0.255715(-5)	0.194(-3)
0.75	0.492(-8)	0.141971(-6)	0.141892(-6)	0.112(-4)
1.0	0.955(-9)	0.254554(-5)	0.254563(-5)	0.194(-3)

In Table 4 we show the errors incurred by the modified functional epsilon algorithm of type $(4, 2)$, the enhanced functional epsilon algorithm of type $(2, 2)$, the improved functional epsilon algorithm of type $(4, 2)$, the functional epsilon algorithm of type $(2, 4)$ for $x = 0(0.25)1$. We see a remarkable precision of the modified functional epsilon algorithm when compared to the other algorithms.

The approximate solutions used in calculating the errors displayed in the Table 4 are

(49)
$$\begin{aligned} MFEA2 = {} & -3.4295 + 2.9530x - 3.4295x^2 + 0.9843x^3 - 0.5716x^4 \\ & + 0.09843x^5 - 0.03811x^6 + 0.004687x^7 - 0.001361x^8 \\ & + 0.1301(-3)x^9 - 0.3008(-4)x^{10} + 0.2214(-5)x^{11} \\ & - 0.369(-6)x^{12} \end{aligned}$$

(50)
$$\begin{aligned} EFEA2 = {} & -3.4295 + 2.9530x - 3.4295x^2 + 0.9843x^3 - 0.5716x^4 \\ & + 0.09843x^5 - 0.03811x^6 + 0.004687x^7 - 0.001361x^8 \\ & + 0.1302(-3)x^9 - 0.3017(-4)x^{10} + 0.2254(-5)x^{11} \\ & - 0.3756(-6)x^{12} \end{aligned}$$

(51)
$$\begin{aligned} IFEA2 = {} & -3.4295 + 2.9530x - 3.4295x^2 + 0.9843x^3 - 0.5716x^4 \\ & + 0.09843x^5 - 0.03811x^6 + 0.004687x^7 - 0.001361x^8 \\ & + 0.1302(-3)x^9 - 0.3017(-4)x^{10} + 0.2254(-5)x^{11} \\ & - 0.3756(-6)x^{12} \end{aligned}$$

(52)
$$\begin{aligned} CFEA2 = {} & -3.4295 + 2.9530x - 3.4295x^2 + 0.9843x^3 - 0.5716x^4 \\ & + 0.09843x^5 - 0.0381x^6 + 0.004642x^7 - 0.001299x^8 \\ & + 0.8106(-4)x^9 - 0.4595(-5)x^{10} + 0.6337(-5)x^{11} \\ & - 0.1056(-5)x^{12}. \end{aligned}$$

TABLE 4. Errors occurring in the solution of (41) using the modified functional epsilon algorithm, the enhanced functional epsilon algorithm, the classical functional epsilon algorithm and the improved functional epsilon algorithm.

x	MFEA2	EFEA2	IFEA2	CFEA2
0.0	0.936(-18)	0.970(-12)	0.960(-12)	0.961(-9)
0.25	0.297(-15)	0.947(-12)	0.957(-12)	0.901(-9)
0.50	0.235(-14)	0.955(-12)	0.946(-12)	0.862(-9)
0.75	0.297(-15)	0.947(-12)	0.957(-12)	0.901(-9)
1.0	0.936(-18)	0.970(-12)	0.960(-12)	0.961(-9)

5. Remarks and conclusion. In this paper, we have shown two new functional epsilon type algorithms, namely the modified functional epsilon algorithm and the enhanced functional epsilon algorithm. These algorithms are essentially for accelerating the convergence of a sequence of functions. Moreover, the performance of these new algorithms has been demonstrated and compared with the classical functional epsilon algorithm and the improved functional epsilon algorithm. The formula of the modified functional epsilon algorithm is based on the second row sequence of the modified Padé approximant and then designed as an iterative process. The prime motive of the development of the modified functional epsilon algorithm and the enhanced functional epsilon algorithm was to increase the accuracy of the established algorithms, particularly the improved functional epsilon algorithm and the classical functional epsilon algorithm.

We have demonstrated the epsilon type algorithms for two types of row sequence purely to illustrate the accuracy of the approximate solution, the stability of the convergence, the consistency of the results and to determine the efficiency of each of the methods. In all the numerical examples performed, we have found that the modified functional epsilon algorithm produces better estimates than the other similar functional epsilon type algorithms. Furthermore, the estimates of the enhanced functional epsilon algorithm are not good as the modified functional epsilon algorithm and are very similar to the improved functional epsilon algorithm. The positive feature of the enhanced functional epsilon algorithm is that it produces better estimates than the classical functional epsilon algorithm. Consequently, it should be noted that, like all other acceleration methods, these new methods each have their own domain of validity and in certain circumstances should not be used.

Acknowledgments. I am grateful to the anonymous referee for his helpful comments on this paper.

REFERENCES

1. G.A. Baker and P.R. Graves-Morris, *Padé approximants*, 2nd ed., Cambridge University Press, Cambridge, 1996.

2. I.D. Coope and P.R. Graves-Morris, *The rise and fall of the vector epsilon algorithm*, Numerical Algorithms **5** (1993), 275–286.

3. B.L. Moiseiwitsch, *Integral equations*, Longman, New York, 1977.

4. F. Smithies, *Integral equations*, Cambridge University Press, Cambridge, 1962.

5. R. Thukral, *Solution of integral equations using integral Padé approximants* I, J. National Academy Math. **12** (1998), 1–14.

6. ———, *A family of Pad'e type approximants for accelerating the convergence of sequences*, J. Comput. Appl. Math. **102** (1999), 287–302.

7. ———, *Solution of integral equations using Padé type approximants*, J. Integral Equations Appl. **13** (2001), 1–26.

8. ———, *Introduction to the improved functional epsilon algorithm*, J. Comput. Appl. Math. **147** (2002), 9–25.

9. ———, *Similarities of the integral Padé approximant*, J. Appl. Math. Comput. **135** (2003), 129–145.

10. ———, *Similarities of the integral Padé approximant* II, J. Appl. Math. Comput. **158** (2004), 869–885.

Padé Research Centre, 39 Deanswood Hill, Leeds, West Yorkshire, LS17 5JS, England
Email address: r.thukral@ntlworld.com