Numerical Solutions of Volterra Integral Equations with a Solution Dependent Delay

BARUCH CAHLON AND LOUIS J. NACHMAN

Department of Mathematical Sciences, Oakland University, Rochester, Michigan 48063

Submitted by C. L. Dolph

INTRODUCTION

Volterra integral equations arise in a wide variety of mathematical, scientific, and engineering problems. One such problem is the solution of parabolic differential equations with initial boundary conditions [1, p. 68]. Another application deals with the temperature in nuclear reactors [1, Chap. IV, Sect. 8] where the delayed neutron is ignored. For more physical applications of Volterra integral equations see the references in [2]. Many physical problems are better represented by a Volterra functional integral equation rather than a Volterra integral equation; that is, the problem has a delay built in which cannot be ignored [3–5].

A considerable amount of work has been done on functional differential equations ([6] and references contained there); that is, ordinary differential equations which have terms which depend on the history of the solution, including optimization problems [7]. Numerical solutions of functional differential equations are derived in [8, 9] and in other papers. There is also a large body of work dealing with numerical solutions of Volterra integral equations without delay (see, e.g., [10, 11]). However, very little work has been done with numerical solutions of Volterra integral equations with a delay. In [12] a numerical solution of Volterra integral equations with a constant delay was considered while in [13] a Volterra integral equation with a nonconstant delay was considered.

In this paper we consider the numerical solution of nonlinear Volterra integral equations with a solution dependent delay. In Section 1 we prove existence and uniqueness of solutions for the equation considered as well as a priori boundedness of the solution. In Section 2 we define a numerical solution and prove that the numerical solution converges to the solution of the integral equation. In Section 3 we discuss several practical methods for computing the numerical solution of Section 2. In Section 4 we supply some numerical examples.
1. Existence and Uniqueness

In this section we give existence and uniqueness theorems for solutions of equations of the form

\begin{align}
Y(x) &= f(x) + \int_0^x H(x, t, y(t), y(t - \tau(y(t)))) \, dt, \quad 0 \leq x \leq a. \quad (1.1a) \\
Y(x) &= g(x), \quad -\tau_0 \leq x < 0. \quad (1.1b)
\end{align}

where \( \tau_0 \) and \( a \) are positive constants.

Equation (1.1a) is a nonlinear Volterra integral equation with a delay which depends on the solution.

The space in which we will prove existence and uniqueness is \( C^\alpha[-\tau_0, a] \), the space of all \( \alpha \) Hölder continuous functions, \( 0 \leq \alpha \leq 1 \), on \([-\tau_0, a] \). This space is a Banach space with norm

\begin{equation}
\|Y\|_{C^\alpha} = \max_{-\tau_0 \leq x \leq a} |Y(x)| + \sup_{x_1, x_2 \in [-\tau_0, a]} \frac{|Y(x_1) - Y(x_2)|}{|x_1 - x_2|^\alpha} \quad (1.2)
\end{equation}

where \( Y \in C^\alpha[-\tau_0, a] \).

We will use the following assumptions on \( g, \tau, \) and \( H \).

(A1) \( g(0) = f(0) \) and \( g \in C^\alpha[-\tau_0, 0], f \in C^\alpha[0, a] \).

(A2) \( g'(0) = f'(0) + H(0, 0, f(0), g(-\tau(f(0)))) \)

(A3) \( -\tau_0 \leq x - \tau(u) < x \) for all \( x \geq 0 \), and all \( u \in \mathbb{R} \),

where \( \mathbb{R} \) is the reals.

(A4) \( \tau \) is continuous on \( \mathbb{R} \)

(A5) The function \( H(x, t, u, v) \) is integrable with respect to \( t \) and Lipschitz with respect to \( x \). That is, for every real \( u \) and \( v \) there is a \( L(u, v) \), a nonnegative and continuous function, such that

\[ |H(x_1, t, u, v) - H(x_2, t, u, v)| \leq L(u, v)|x_1 - x_2|. \]

(A6) The function \( H(x, t, u, v) \) satisfies a Lipschitz condition with respect to \( u \) and \( v \). That is, there are constants \( L_1 \) and \( L_2 \) such that

\[ |H(x, t, u_1, v_1) - H(x, t, u_2, v_2)| \leq L_1 |u_1 - u_2| + L_2 |v_1 - v_2|. \]

We will also need the following sets

\[ D = \{ s \in C^\alpha[-\tau_0, a] | s(x) = g(x) \text{ for } -\tau_0 \leq x \leq 0 \}, \quad (1.3) \]
and for each \( r > 0 \),
\[
B_r = \{ s \in C^\alpha[-\tau_0, a] \mid \| s \|_{C^\alpha} \leq r \}. \tag{1.4}
\]
\[
E_r = B_r \cap D. \tag{1.5}
\]
For the set \( D \) it is easy to show

**Remark 1.** The set \( D \) is closed and convex in the space \( C^\alpha[-\tau_0, a] \).

Using Remark 1, it is then easy to see that, for \( r \) sufficiently large, \( E_r \) is not empty and \( E_r \) is convex, bounded, and closed in \( C^\alpha[-\tau_0, a] \).

We define the operator \( T \) on \( E_r \) as follows:

Let \( y \in E_r \) and
\[
(Ty)y = f(x) + \int_0^x H(x, t, y(t), y(t - \tau(y(t)))) \, dt, \quad 0 \leq x \leq a, \tag{1.6a}
\]
\[
(Ty)y = g(x), \quad -\tau_0 \leq x \leq 0. \tag{1.6b}
\]
We first show existence and uniqueness of the solution to (1.1) in \( C^\alpha[-\tau_0, a] \) for the case \( \alpha = 0 \). We will need the following lemmas.

**Lemma 1.1.** Suppose (A1), (A3)-(A6) hold and that \( \alpha = 0 \). Then the range of the operator \( T \) is a subset of \( E_r \) for \( r \) sufficiently large and sufficiently small.

**Proof.** Since we consider \( \alpha = 0 \) it is clear from assumptions (A1), (A3)-(A5) that \( Ty \in D \). We will show that \( Ty \in B_r \).

Consider
\[
\| Ty \|_{C^0} \leq \max \left\{ \sup_{0 \leq x \leq a} \left\| f(x) + \int_0^x H(x, t, y(t), y(t - \tau(y(t)))) \, dt \right\|, \right. \\
\left. \sup_{-\tau_0 \leq x \leq 0} \| g(x) \| \right\}
\]
\[
\leq \max \left\{ \sup_{0 \leq x \leq a} \| f(x) \| \\
+ \sup_{0 \leq x \leq a} \left\| \int_0^x H(x, t, 0, 0) - H(x, t, y(t), y(t - \tau(y(t)))) \, dt \right\|, \\
+ \sup_{0 \leq x \leq a} \left\| \int_0^x H(x, t, 0, 0) \, dt \right\|, \sup_{-\tau_0 \leq x \leq 0} \| g(x) \| \right\}. \tag{1.7}
\]
From assumptions (A1), (A3)-(A6), there is an \( m \) such that
\[
m = \max_{0 \leq x \leq a} \left\| \int_0^x H(x, t, 0, 0) \, dt \right\| < \infty \tag{1.8}
\]
and
\[ \| Ty \|_{\mathcal{C}^0} \leq \max \{ \| f \|_{\mathcal{C}^0} + a(L_1 + aL_2 \| y \|_{\mathcal{C}^0} + m, \| g \|_{\mathcal{C}^0} \} \]
\[ \leq \max \{ \| f \|_{\mathcal{C}^0} + aL_1r + aL_2r + m, \| g \|_{\mathcal{C}^0} \}. \quad (1.9) \]

Let \( r \) be large enough so that
\[ \| g \|_{\mathcal{C}^0} < r \quad (1.10a) \]
and
\[ \| f \|_{\mathcal{C}^0} + aL_1r + aL_2r + m < r, \quad (1.10b) \]
or
\[ r > \max \left\{ \| g \|_{\mathcal{C}^0}, \frac{\| f \|_{\mathcal{C}^0} + m}{1 - a(L_1 + L_2)} \right\} \quad (1.11) \]
and let \( a \) be such that
\[ 0 < a < \frac{1}{L_1 + L_2}. \quad (1.12) \]

From (1.10a)–(1.12), we can conclude that \( Ty \in E_r \), completing the proof.

**Lemma 1.2.** Under the assumptions of Lemma 1.1, the operator \( T \) is continuous and compact.

**Proof.** We first show that \( T \) is compact. From Lemma 1.1 \( T \) is bounded. Let \( y \in E_r \) and let \( x_1 \geq x_2 \geq 0 \). Consider
\[ |(Ty)_{x_1} - (Ty)_{x_2}| \leq |f(x_1) - f(x_2)| \]
\[ + \int_{x_1}^{x_2} |H(x_2, t, y(t), y(t - \tau(y(t))))| \, dt \]
\[ + \int_{0}^{\tau_{y_1}} |H(x_1, t, y(t), y(t - \tau(y(t))))| \, dt \]
\[ - H(x_2, t, y(t), y(t - \tau(y(t)))) \, dt. \quad (1.13) \]
The fact that \( y \in E_r \), assumptions (A1), (A3)–(A5), and the fact that \( f \) is a continuous function on a closed interval imply that there are numbers \( \delta_1, \delta_2, \) and \( \delta_3 > 0 \) such that for \( |x_1 - x_2| < \delta_1 \), the first term of (1.13) is less than \( \varepsilon/3 \), for \( |x_1 - x_2| < \delta_2 \), the second term of (1.13) is less than \( \varepsilon/3 \), and for \( |x_1 - x_2| < \delta_3 \), the third term of (1.13) is less than \( \varepsilon/3 \). Hence, for \( |x_1 - x_2| < \delta = \min(\delta_1, \delta_2, \delta_3) \)
\[ |(Ty)_{x_1} - (Ty)_{x_2}| < \varepsilon. \quad (1.14) \]
For \( x_1 < x_2 < 0 \) the proof is obvious and for \( x_1 < 0 < x_2 \) we can follow the proof of the first case. Hence, \( T \) is a compact operator.

We now show that \( T \) is continuous. Let \( y_k \in E \), and \( y_k \to y \). Since \( TE \), is compact, there exists a subsequence \( T y_{k_j} \), which converges to \( y \) as \( j \to \infty \). That is,

\[
\gamma(x) = \lim_{j \to \infty} \left[ f(x) + \int_0^x H(x, t, y_{k_j}(t), y_{k_j}(t - \tau(y_{k_j}(t)))) \, dt \right]. \tag{1.15}
\]

Using the Lebesgue dominated convergence theorem and the fact that \( y_{k_j} \to y \) and \( \tau(y_{k_j}) \to \tau(y) \), we have

\[
\gamma(x) = f(x) + \int_0^x H(x, t, y(t), y(t - \tau(y(t)))) \, dt = (Ty)x \tag{1.16}
\]

for \( a \geq x \geq 0 \). Equation (1.16) is also clearly true for \( -\tau_0 \leq x \leq 0 \).

Since any subsequence of \( \{Ty_{k_j}\} \) converges to the same limit \( \gamma \), we conclude that \( \{Ty_{k_j}\} \) converges to \( (Ty) \) and \( T \) is continuous.

As a result of Lemmas 1.1, 1.2, and a Schauder fixed point theorem, we have

**Theorem 1.1.** Suppose that (A1), (A3)-(A6) are satisfied. Then the integral equation (1.1) has a solution \( y \) on \([0, a]\) where \( a \) satisfies (1.12) and \( y \in E \), where \( r \) satisfies (1.11).

From Theorem 1.1 the existence of a continuous solution is guaranteed while conditions (A1) and (A5) imply that a solution of (1.1) will satisfy a Lipschitz condition. That is, \( y \in C^\alpha[-\tau_0, a] \) for \( \alpha = 1 \).

Uniqueness is discussed in

**Theorem 1.2.** Suppose the assumptions of Theorem 1.1 hold. Suppose also

(A7) \( \tau \) satisfies a uniform Lipschitz condition with Lipschitz constant \( L_\tau \).

Then the integral equation (1.1) has a unique solution \( y \) for \(-\tau_0 \leq x \leq a\) where \( a \) satisfies (1.12) and

\[
(L_1 + L_2)^2 a^2 - a((\| f \| + m) L_2 L_\tau + 2(L_1 + L_2)) + 1 > 0 \tag{1.17}
\]

and \( y \in E \), for \( r \) satisfying (1.11) and

\[
r < \frac{1 - a(L_1 + L_2)}{aL_2 L_\tau}. \tag{1.18}
\]
If in addition to the above, \( \| g \| > (\| f \| + m)/(1 - \alpha a) \), then there exists a unique solution to (1.1) on the interval \([-\tau_0, a]\) where \( a \) satisfies (1.12), (1.17), and also

\[
0 < a < \frac{1}{L_1 + L_2 + \| g \| L_2 L_2}
\]

(1.19)

and \( y \in E_r \) for \( r \) satisfying (1.11) and (1.18).

The proof can be obtained easily by using the Lipschitz conditions on \( H \) and \( \tau \) and from the fact that any two solutions are identical in \([-\tau_0, 0]\). The existence and uniqueness can be extended beyond \([-\tau_0, a]\) if \( H \) and \( \tau \) are uniformly Lipschitz, by an extension process.

In many applications the function \( H(x, t, y(t), y(t - \tau(y(t)))) \) has the form

\[
H(x, t, y(t), y(t - \tau(y(t)))) = k(x, t) h(x, t, y(t), y(t - \tau(y(t))))
\]

so that equations (1.1a) and (1.1b) take the form

\[
y(x) = f(x) + \int_0^x k(x, t) h(x, t, y(t), y(t - \tau(y(t)))) \, dt, \quad 0 \leq x \leq a \quad (1.20a)
\]

\[
y(x) = g(x), \quad -\tau_0 \leq x < 0. \quad (1.20b)
\]

Assume (A1), (A3), (A7), and \( h \) satisfies (A6) and (A5) (we can assume that \( h(x, t, \cdot, \cdot) \) is continuous with respect to \( t \)), and that \( k(x, t) \) satisfies the conditions:

(A8) for fixed \( x, k(x, t) \in L^1[0, a] \),

(A9) \( \int_0^a k(x, t) \, dt \) is Lipschitz with respect to \( x \) for \( 0 \leq x \leq a \).

Then from the results of this section, it is easily seen that (1.20a), (1.20b) has a unique solution satisfying a Lipschitz condition for \( a \) satisfying (1.12) and (1.17).

2. Convergence

In this section we consider a numerical scheme for approximating solutions to Eq. (1.20). We prove that solutions to the numerical scheme converge to the solution of the integral equation.

To approximate the solution of Eqs. (1.20a) and (1.20b) we consider uniform partitions \( \{t_{j,n}\} \) for the interval \([0, a]\) and the approximation

\[
\int_0^x k^\pm(x, t) y(t) \, dt \sim \sum_{j=0}^n w_{j,n}^\pm(x) y(t_{j,n}), \quad (2.1)
\]
where $k^{+}$ and $k^{-}$ are, respectively, the positive and negative parts of $k$.

The functions $w_{j,t}^{+}$ and $w_{j,t}^{-}$ are $2n + 2$ numerical weight functions satisfying

(A10) $w_{j,n}^{\pm}(x) \in R[0, a]$.

(A11) $w_{j,n}^{+}(x) \geq 0$, $w_{j,n}^{+}(t_0, n) = 0$ and $w_{j,n}^{+}(x) = 0$ for $x > t_{j,n}$.

(A12) $\lim_{n \to \infty} \left\| \int_0^x k^{+}(x, t) y(t) \, dt - \sum_{j=0}^n w_{j,n}^{+}(x) y(t_{j,n}) \right\| = 0$ for $y \in C[0, a]$.

Here $R[0, a]$ is the space of Reimann integrable functions defined on $[0, a]$ and $\| \cdot \|$ denotes the supremum norm on $C[0, a]$.

We now let

$$w_{j,n}(x) = w_{j,n}^{+}(x) - w_{j,n}^{-}(x)$$

so that $\lim_{n \to \infty} \left\| \int_0^x k(x, t) y(t) \, dt - \sum_{j=0}^n w_{j,n}(x) y(t_{j,n}) \right\| = 0$ for $y \in C[0, a]$.

We also assume

(A13) $\sum_{j=0}^n w_{j,n}(x) \left( h(x, t_{j,n}, s(t_{j,n}), s(t_{j,n} - \tau(s(t_{j,n})))) - \int_0^x k(x, t) h(x, t, s(t), s(t - \tau(s(t)))) \, dt \right)$ as $n \to \infty$ and for $0 < x \leq a$, where $s(x)$ is the solution of (1.20a) and (1.20b).

To find an approximate solution for Eqs. (1.20a) and (1.20b) we replace them with the equations

$$s_{n}(x) = f(x) + \sum_{j=0}^n w_{j,n}(x) h(x, t_{j,n}, s_{n}(t_{j,n}))$$

and

$$s_{n}(x) = g(x) \quad \text{for} \quad -\tau_0 \leq x \leq 0. \quad (2.3b)$$

We will say that the solution $s_{n}(x)$ of (2.3a) and (2.3b) converges to $s(x)$ on $[0, a]$ if

$$\lim_{n \to \infty} \sup_{0 \leq x \leq a} |s_{n}(x) - s(x)| = 0.$$ 

For convenience, we will drop the $n$ subscript from $t$ and $w$ throughout the remainder of this section.

The following lemma follows easily from its hypotheses.

**Lemma 2.1.** Suppose that $k(x, t)$ satisfies (A8) and (A9). Then for $\varepsilon > 0$ there exists a $\delta = \delta(\varepsilon)$ such that for all $0 \leq x_1 < x_2 < a$, with $0 < x_2 - x_1 \leq \delta$ we have

$$\int_{x_1}^{x_2} |k(x, t)| \, dt < \varepsilon.$$

The numerical weight functions $w_{j}(x)$ satisfy an analog of Lemma 2.1.
For a partition \( \{ t_j \}_{j=1}^n \) we will use the notation \([c, d]_n\) or \([c, d]_n\) for the set of integers \( j \) such that \( t_j \in [c, d] \) or \( t_j \in [c, d] \), respectively.

Using Lemma 2.1 of [14] one can show

**Lemma 2.2.** Suppose (A8)-(A12) are satisfied. Then for each \( \varepsilon > 0 \), there exist points \( 0 = p_1 < \cdots < p_\rho(x) = a \) and an \( N = N(\varepsilon) \) such that for all \( n \geq N, \)

\[
\sum_{j \in [p_\rho, p_{\rho+1}]} |w_j(x)| < \varepsilon \quad \text{for } i = 0, 1, \ldots, R(\varepsilon) - 1
\]

uniformly for \( x \in [0, a] \).

**Corollary 2.3.** Suppose (A8)-(A12) are satisfied. Then for each \( \varepsilon > 0 \) there is an \( N \) such that for all \( n \geq N, \; x \in [0, a], \) and \( 0 < j \leq n, \) we have \( |w_j(x)| < \varepsilon \).

**Theorem 2.4.** Assume (A1), (A3), and (A5)-(A12) are satisfied. Then

\[
|s_n(x) - s(x)| \leq QM_R,
\]

where \( M_1 = 4 \) and \( M_r \) is defined by the recurrence relation

\[
M_r = \frac{4}{3} + \frac{(2r - 1)}{6} M_{r-1} \quad \text{for } r = 2, 3, \ldots, R
\]

and

\[
Q = \sup_{0 \leq s \leq a} \left| \int_0^s k(x, t) h(x, t, s(t), s(t - \tau(s(t)))) dt - \sum_{j=0}^{n-1} w_j(x) h(x, t_j, s(t_j), s(t_j - \tau(s(t_j)))) \right|.
\]

**Proof.** We prove this result on the intervals \([p_{j-1}, p_j], j = 1, 2, \ldots, R\), by induction. We first show (2.5) for \( j = 1 \). Let \( \varepsilon(x) = s_n(x) - s(x) \) and \( t \in [p_0, p_1]_n \). Then

\[
\varepsilon(x) = \sum_{j=0}^{n-1} w_j(x) h(x, t_j, s_n(t_j), s_n(t_j - \tau(s_n(t_j))))
\]

\[
- \int_0^s k(x, t) h(x, t, s(t), s(t - \tau(s(t)))) dt
\]
or

\[ e(x) = \sum_{j=0}^{n} w_j(x) \left[ h\left(x, t, s_n(t_j), s_n(t_j - \tau(s_n(t_j)))\right) - h(x, t_j, s(t_j), s(t_j - \tau(s(t_j)))) \right] \]

\[ - \left[ \int_{0}^{t} k(x, t) h(x, t, s(t), s(t - \tau(s(t)))) \, dt \right. \]

\[ - \sum_{j=0}^{i} w_j(x) h(x, t_j, s(t_j), s(t_j - \tau(s(t_j)))) \right]. \quad (2.9) \]

From assumption (A6),

\[ |e(x)| \leq L_1 \sum_{j=0}^{i} |w_j(x)| |s_n(t_j) - s(t_j)| \]

\[ + L_2 \sum_{j=0}^{i} |w_j(x)| |s_n(t_j - \tau(s_n(t_j))) - s(t_j - \tau(s(t_j)))| \]

\[ + Q(x), \quad (2.10) \]

where \( Q(x) \) is the absolute value of the second term. Now consider

\[ L_2 \sum_{j=0}^{i} |w_j(x)| |s_n(t_j - \tau(s_n(t_j))) - s(t_j - \tau(s(t_j)))| \]

\[ \leq L_2 \sum_{j=0}^{i} |w_j(x)| |e(t_j - \tau(s_n(t_j)))| \]

\[ + L_2 \sum_{j=0}^{i} |w_j(x)| |s(t_j - \tau(s_n(t_j))) - s(t_j - \tau(s(t_j)))|. \quad (2.11) \]

From Theorem 1.2, there is a unique solution to the integral equation and (A1) and (A5) imply that the solution satisfies a Lipschitz condition. Moreover, from condition (A7), \( \tau \) also satisfies a Lipschitz condition and therefore, from (2.11),

\[ L_2 \sum_{j=0}^{i} |w_j(x)| |s_n(t_j - \tau(s_n(t_j))) - s(t_j - \tau(s(t_j)))| \]

\[ \leq L_2 L_s L_\tau \sum_{j=0}^{i} |w_j(x)| |e(t_j)| \]

\[ + L_2 \sum_{j=0}^{i} |w_j(x)| |e(t_j - \tau(s_n(t_j)))|, \quad (2.12) \]
where \( L_r \) and \( L_s \) are the Lipschitz constants for \( \tau \) and \( s \), respectively. Therefore (2.10) and (2.12) imply

\[
|\varepsilon(x)| \leq L_1 \sum_{j=0}^i |w_j(x)||s_n(t_j) - s(t_j)| + L_2L_sL_r \sum_{j=0}^i |w_j(x)||\varepsilon(t_j)| + L_2 \sum_{j=0}^i |w_j(x)||\varepsilon(t_j - \tau(s_n(t_j)))| + Q(x). \tag{2.13}
\]

We define

\[
\|\varepsilon\|_i = \sup \{|\varepsilon(x)| : x \in [t_{j-1}, t_j]\}, \tag{2.14a}
\]

\[
\|\varepsilon\|_i = \max \{\|\varepsilon\|_j : 0 \leq j \leq i\}. \tag{2.14b}
\]

It is clear from \( A3 \) that for \( n \) sufficiently large,

\[
t_j - \tau(u) < t_{j-1} \quad \text{for all } u \in \mathbb{R}. \tag{2.15}
\]

From (2.14) and (2.15) and (2.13) we obtain

\[
|\varepsilon(x)| \leq L_1 \sum_{j=0}^i |w_j(x)||\varepsilon\|_j + L_2L_sL_r \sum_{j=0}^i |w_j(x)||\varepsilon\|_j + L_2 \sum_{j=0}^{i-1} |w_j(x)||\varepsilon\|_j + Q(x) \tag{2.16}
\]

or

\[
|\varepsilon(x)| \leq (L_1 + L_2L_sL_r \sup_{x \in [p_0, p_1]} \sum_{j=0}^i |w_j(x)||\varepsilon\|_j) + L_2 \sup_{x \in [p_0, p_1]} \sum_{j=0}^{i-1} |w_j(x)||\varepsilon\|_j + \sup_{x \in [p_0, p_1]} Q(x). \tag{2.17}
\]

From Corollary 2.3 we can choose \( n \) sufficiently large to insure that

\[
\sup_{x \in [p_0, p_1]} (1 - (L_1 + L_2L_sL_r)|w_n(x)|)^{-1} < 2. \tag{2.18}
\]

From (2.17), (2.18) and the definitions (2.14a) and (2.14b) we have

\[
\|\varepsilon\|_i \leq \sum_{j=0}^{i-1} \frac{q_j}{2} \|\varepsilon\|_j + 2Q, \tag{2.19a}
\]

where

\[
Q = \sup_{x \in [0, a]} Q(x). \tag{2.19b}
\]
and
\[
\sum_{j=0}^{i-1} \frac{a_j}{2} = (L_1 + L_2 L_r L_t) \sup_{x \in [p_0, p_1]} \left( \sum_{j=0}^{i-1} |w_j(x)| \right) + L_2 \sup_{x \in [p_0, p_1]} \left( \sum_{j=1}^{i-1} |w_j(x)| \right).
\] (2.20)

Using Lemma 2.2 and the Linz lemma [15] we obtain \( \xi_i = \alpha^i \xi \) and
\[
\| \bar{\xi} \| \leq \frac{2Q + 0 \cdot (1/2)}{1 - (1/2)} = 4Q = M_1 Q
\] (2.21)
completing the first part of the proof.

Now assume that
\[
\| \bar{\xi} \| \leq M_r Q
\] (2.22)
for \( i \in [p_{r-1}, p_r] \) and \( l = 1, 2, \ldots, r - 1 \). The \( M_r \) are defined recursively in (2.6). We will show that (2.22) holds for \( l = r \). Arguing as in (2.13) we obtain the following inequality for \( i \in [p_{r-1}, p_r] \)
\[
| \varepsilon(x) | \leq (L_1 + L_2 L_r L_t) \left( \sum_{j=1}^{r-1} \sum_{l \in [p_{j-1}, p_j]} |w_j(x)| \| \varepsilon \|_l + \sum_{l \in [p_{r-1}, p_r]} |w_r(x)| \| \varepsilon \|_l \right)
+ L_2 \left[ \sum_{j=1}^{r} \sum_{l \in S_{ij}} |w(x)| | \varepsilon (t_l - \tau(s_n(t_l))) | + \sum_{l \in S_2} |w_r(x)| | \varepsilon (t_l - \tau(s_n(t_l))) | \right] + \bar{Q}(x),
\] (2.23)
where \( S_{ij} = \{ l \in [p_{j-1}, p_j] : t_l - \tau(s_n(t_l)) \leq p_{j-1}, l \leq i \} \) and \( S_2 = \{ l \leq i : t_l - \tau(s_n(t_l)) > p_{r-1} \} \). From Lemma 2.2, for \( n \) sufficiently large
\[
\sup_{x} \sum_{l \in [p_{r-1}, p_r]} |w_r(x)| \leq \frac{1}{8(L_1 + L_2 L_r L_t)} < \frac{1}{8V},
\] (2.24)
where \( V = \min(L_1, L_2, L_r, L_t) \). Using the notation of (2.14a), (2.14b), and (2.19a) we obtain the following
\[
| \varepsilon(x) | \leq (L_1 + L_2 L_r L_t) \left( \sum_{j=1}^{r-1} \sup_{x \in [p_{j-1}, p_j]} \sum_{l \in [p_{j-1}, p_j]} |w_j(x)| \| \varepsilon \|_l \right)
+ \sup_{l \in [p_{r-1}, p_r]} |w_r(x)| \| \varepsilon \|_l \)
Using (2.14a) and (2.14b) again, and (2.22) we have

\[
\| \tilde{e} \|_i \leq (L_1 + L_2 L_s L_r) \left[ \sum_{j=1}^{r-1} \sup x \sum_{\ell \in [p_{j-1}, p_j]} |w_j(x)| \| M_j \|_i \right.
\]
\[
+ \sup x \sum_{\ell \in [p_{j-1}, p_j]} |w_j(x)| \| \tilde{e} \|_{i-1}
\]
\[
+ L_2 \sum_{j=1}^r \sup x \sum_{t_{j-1} < p_{j-1}} |w_j(x)| \| \tilde{e} \|_{i-1}
\]
\[
+ L_2 \sup x \sum_{t_{j-1} > p_{j-1}} |w_j(x)| \| \tilde{e} \|_{i} + Q.
\]  

We can justify inequality (2.26) as follows. The first and second terms come from (2.22) and the notation of (2.14a) and (2.14b). In the third term the sum could actually be on fewer points, such as \( t_j - \tau(s_n(t_i)) < t_{j-1} < p_{j-1} \). In this case this point will appear in the fourth sum. Since \( \| \tilde{e} \|_j \) is monotonically increasing with respect to \( j \) we can therefore replace this value by a greater value in the fourth sum, thus showing the validity of the right-hand side. Since the right-hand side is independent of \( x \) we therefore have the validity of inequality (2.26).

It is clear from (2.6) that \( M_j \geq 0 \) for all \( j \geq 1 \) and monotonically increasing. Therefore,

\[
\| \tilde{e} \|_i \leq (L_1 + L_2 L_s L_r) \left[ M_{r-1} Q \sum_{j=1}^r \sup x \sum_{\ell \in [p_{j-1}, p_j]} |w_j(x)| \right.
\]
\[
+ \sup x \sum_{\ell \in [p_{j-1}, p_j]} |w_j(x)| \| \tilde{e} \|_{j-1}
\]
\[
+ L_2 M_{r-1} Q \sum_{j=1}^r \sup x \sum_{t_j < p_{j-1}} |w_j(x)|
\]
\[
+ L_2 \sup x \sum_{t_j > p_{j-1}} |w_j(x)| \| \tilde{e} \|_{j} + Q.
\]  

(2.27)
From (2.24)
\[ \| \tilde{e} \|_1 \leq (r - 1) M_{r - 1}^Q + \frac{rQM_{r - 1}}{8} + \sum_{l \in \{p_{r - 1}, p_r\}_n} \alpha_{ll} \| \tilde{e} \|_1, \]  
(2.28)

where
\[ \sum_{x \|} \leq (L_1 + L_2 L_3 L_4) \sup_{x \in \{p_{r - 1}, p_r\}_n} \sum_{l \leq i} |w_i(x)| + L_2 \sup_{x \|} \sum_{l > p_{r - 1}, i} |w_i(x)| < \frac{1}{\alpha}. \]  
(2.29)

Now using the Linz lemma [15], we obtain
\[ \| \tilde{e} \|_1 \leq \frac{\alpha}{3} T, \]  
(2.30)

where \( T = Q + (r - 1) M_{r - 1} Q + (rQM_{r - 1}/8) \). Therefore,
\[ \| \tilde{e} \|_1 \leq \frac{4}{3} Q \left[ 1 + \frac{r - 1}{8} M_{r - 1} + \frac{r}{8} M_{r - 1} \right] = QM, \]  
(2.31)

and the theorem is proved.

From Theorem 2.4 we conclude immediately

**Theorem 2.5.** Assume \( (A1), (A3), (A5)-(A12) \) are satisfied. Then
\[ \lim_{n \to \infty} \sup_{0 \leq x \leq a} (|s_n(x) - s(x)|) = 0. \]

Moreover, from the proof of Theorem 2.4 we have

**Corollary 2.6.** Assume that \( (A1), (A3), \) and \( (A5)-(A12) \) are satisfied. Then the rate of convergence is \( O(Q) \), that is,
\[ \sup_{x} (|s_n(x) - s(x)|) = O(Q). \]

**3. Numerical Methods**

To solve Eq. (1.20) we replace it with the system of equations
\[ s_n(x) = f(x) + \sum_{j=0}^{n} w_j(x) h(x, t_j, s_n(t_j), s_n(t_1 - \tau(s_n(t_1)))) \]  
(3.1)
If hypotheses (A1), (A3), and (A5)-(A12) are satisfied, then
\[ \sup_x \sup_{0 \leq x \leq \alpha} (|s_n(x) - s(x)|) = 0 \] by Theorem 2.5.

The system above is highly nonlinear and it is not possible to find a reasonable finite system of equations which could be solved to produce a solution. Instead, we consider an iteration scheme to solve for \[ \{s_n(t_j)\}_{j=0}^\infty. \]

We first choose nodes \( \{t_j\}_{j=0}^n \) uniformly on \([0, \alpha]\) with \( n \) large enough to ensure
\[ t_j - \tau(u) < t_{j-1} \quad \text{for all } u \in \mathbb{R}. \] (3.2)

We set \( s_n(t_0) = f(t_0) = g(t_0) \) and proceed to compute \( s_n(t_1) \) as follows: From (3.1) we have
\[
\begin{align*}
    s_n(t_1) &= f(t_1) + w_0(t_1) h(t_1, t_0, s_n(t_0), s_n(t_0 - \tau(s_n(t_0)))) \\
    &\quad + w_1(t_1) h(t_1, t_1, s_n(t_1), s_n(t_1 - \tau(s_n(t_1))))
\end{align*}
\]

Let \( s_0^0(t_1) = f(t_1) + w_0(t_1) h(t_1, t_0, s_n(t_0), s_n(t_0 - \tau(s_n(t_0)))) \). Since \( s_n(t_0) \) is known and \( t_0 - \tau(s_n(t_0)) < 0 \) we can use \( s_n(t_0 - \tau(s_n(t_0)))) = g(t_0 - \tau(s_n(t_0))) \) to compute \( s_0^0(t_1) \).

Assuming we have computed \( s_n(t_k) \), let \( s_n^k(t_1) = s_n^k(t_0) + w_i(t_1) h(t_1, t_1, s_n(t_1), s_n(t_1 - \tau(s_n(t_1)))) \). Again since \( t_1 - \tau(s_n(t_1)) < t_0 = 0 \), by (3.2), we can use \( g(t_1 - \tau(s_n(t_1))) \) for \( s_n^k(t_1 - \tau(s_n(t_1))) \) and \( s_n^{k+1}(t_1) \) can be computed.

We let \( s_n(t_1) = \lim_{k \to \infty} s_n^k(t_1) \).

Let us assume that \( \{s_n(t_j)\}_{j=0}^\infty \) have been computed. To extend the solution to \( t_{j_0} \) we again consider the iteration scheme
\[
\begin{align*}
    s_{n+1}^j(t_{j_0}) &= s_n^j(t_{j_0}) + w_j(t_{j_0}) \\
    &\quad \times h(t_{j_0}, t_{j_0}, s_n^j(t_{j_0}), s_n^j(t_{j_0} - \tau(s_n^j(t_{j_0})))), \quad (3.3)
\end{align*}
\]

where
\[
\begin{align*}
    s_n^0(t_{j_0}) &= f(t_{j_0}) + \sum_{j=0}^{j_0-1} w_j(t_{j_0}) \\
    &\quad \times h(t_{j_0}, t_j, s_n(t_j), s_n(t_j - \tau(s_n(t_j))))). \quad (3.4)
\end{align*}
\]

The problem with this scheme is that, while \( \{s_n(t_j)\}_{j=0}^\infty \) are assumed known, \( t_j - \tau(s_n(t_j)) \) is generally not a node and so \( s_n(t_j - \tau(s_n(t_j))) \) must be approximated using the known node values \( \{s_n(t_j)\}_{j=0}^\infty \). Further, if \( \{s_n(t_j)\}_{j=0}^\infty \) are known, we must still find some way to approximate \( s_n(t_{j_0} - \tau(s_n^0(t_{j_0}))) \).

We solve the first of these problems by changing our induction hypothesis, assuming that not only \( \{s_n(t_j)\}_{j=0}^\infty \) are known, but also
{s_n(t_j - \tau(s_n(t_j)))}_{j<j_0} \text{ are known. From (3.2) and the discussion of the computation of } s_n(t_1), \text{ we have no problem with } s_n(t_j - \tau(s_n(t_j))).

To approximate \( s'_n(t_{j_0} - \tau(s'_n(t_{j_0}))) \) we note that \( t_{j_0} - \tau(s'_n(t_{j_0})) \in [0, t_{j_0} - 1] \) by (3.2). We can therefore use the known values \{s_n(t_j)\}_{j<j_0} \text{ and } \{s_n(t_j - \tau(s_n(t_j)))\}_{j<j_0} \text{ to approximate } s'_n(t_{j_0} - \tau(s'_n(t_{j_0}))). \text{ In the numerical examples in §4 we used the following two methods for approximating } s'_n(t_{j_0} - \tau(s'_n(t_{j_0}))), \text{ one based on the iteration scheme of (3.3) and (3.4) and the other based on cubic spline interpolating functions.}

**Method 1.** If \{s_n(t_j)\}_{j<j_0} \text{ and } \{s_n(t_j - \tau(s_n(t_j)))\}_{j<j_0} \text{ are known, then Eq. (3.1) gives us a formula for } s_n(x) \text{ for any } x \leq t_{j_0} - 1. \text{ Since } t_{j_0} - \tau(s'_n(t_{j_0})) < t_{j_0} - 1 \text{ by (3.2), we can use (3.1) to approximate } s'_n(t_{j_0} - \tau(s'_n(t_{j_0}))) \text{ with } s_n(t_{j_0} - \tau(s'_n(t_{j_0}))).

Method 1 has the advantage of being relatively quick, and we know that it will converge to the solution of the original equation from the theorems of Section 2. It does, however, tend to compound round-off error and so we considered another interpolation scheme which should be more accurate.

**Method 2.** We introduce mesh points \{t_j\}_{j=-n}^0 \text{ on } [-\tau_0, 0] \text{ and use a cubic spline to interpolate the values } \{s_n(t_j)\}_{j=0}^{j_0-1}. \text{ As each } s'_n(t_{j_0}) \text{ is computed, a new spline, interpolating } \{s_n(t_j)\}_{j=0}^{j_0-1} \text{ and } s'_n(t_{j_0}) \text{ is computed and its values at } x = t_{j_0} - \tau(s'_n(t_{j_0})) \text{ is used to approximate } s'_n(t_{j_0} - \tau(s'_n(t_{j_0}))).

Whichever method is used, we take } s_n(t_{j_0}) = \lim_{t \to -\tau_0} s'_n(t_{j_0}) \text{ and then approximate } s_n(t_{j_0} - \tau(s_n(t_{j_0}))) \text{ using the chosen interpolation scheme.}

To completely specify a numerical scheme we must, in addition to specifying the method of computing } t_{j_0} - \tau(s'_n(t_{j_0})), \text{ also specify the numerical weight functions } \{w_i(x)\}. \text{ The weight functions chosen will depend, more or less, on the nature of the function } H(x, t, u, v) \text{ and its decomposition}

\[ H(x, t, u, v) = k(x, t) h(x, t, u, v). \]

In the following section we consider two different examples whose decompositions necessitate different kinds of weight functions.

4. **Numerical Examples**

We consider two examples.

**Example 1.**

\[
y(x) = \begin{cases} f(x) + \int_0^x 3\tau(y(t) + y(t - \tau(y(t)))) \, dt & \text{for } x \geq 0 \\ 0 & \text{for } x < 0. \end{cases}
\]
where
\[ \tau(u) = e^{-u} + 0.01 \quad \text{for} \quad u \in \mathbb{R} \]
and
\[ f(x) = x - \int_0^x 3t(t + \rho(t)) \, dt, \]
where
\[ \rho(t) = \begin{cases} 0 & \text{if} \quad t - \tau(t) < 0, \\ t - \tau(t) & \text{if} \quad t - \tau(t) \geq 0. \end{cases} \]
This problem has \( s(t) \) as solution where
\[ s(t) = \begin{cases} 0 & \text{if} \quad t < 0, \\ t & \text{if} \quad t \geq 0. \end{cases} \]
Let \( k(x, t) = 3t \) and \( h(x, t, u, v) = u + v \). Then
\[ H(x, t, u, v) = k(x, t) h(x, t, u, v). \]
In an example of this type, where both \( k \) and \( h \) are defined and integrable throughout the interval \([0, 1]\), we can use any simple integration technique to approximate the integral
\[ \int_0^x k(x, t) h(x, t, u, v) \, dt \quad (4.1) \]
as long as the weight functions implicitly defined by the integration technique satisfy the hypotheses (A10)–(A13). For this example we used Simpson's Rule with a corresponding linear piece on \([0, t]\) when necessary. We also used Simpson's three-eights rule with appropriate quadratic or pair of quadratic approximations on the left-hand end of the partition \( \{t_j\}_{j=1}^n \) when necessary. The results on the interval \([0, 1]\) for \( n = 10 \) and \( n = 100 \) are given in Tables I and II. Iteration was halted when
\[ |s_{n}^{i+1}(t_j) - s_{n}^{i}(t_j)| < 10^{-9}. \]
All computations were performed in double precision on Oakland University's Honeywell Level 68DPS1 under the Multics operating system. Answers are truncated to 9 digits.
For our second example we consider the heat equation
\[ u_t(x, t) = u_{xx}(x, t) \quad \text{for} \quad x > 0, \quad t > 0 \quad (4.2) \]
with boundary conditions

\[ u(x, 0) = f(x) \]  
\[ -u_t(0, t) = G(u(x_0, t - \tau), \varphi(t)), \]

where \( G(u, v) \) is continuous in both variables and denotes the rate of flow of heat from \( x_0 \) at temperature \( u \) to the end of the rod at temperature \( v \). Note the delay in (4.3b); the rate of flow at time \( t \) depends on the temperature at \( x \) at some earlier time \( t - \tau \). Such a situation might occur if the reaction of a thermostat was delayed. Standard techniques applied to find a solution of the above problem for \( x \) fixed at \( x_0 \), that is, to solve for \( y(t) = \)
TABLE II
Solutions to Example 1 using Method 2

<table>
<thead>
<tr>
<th>t</th>
<th>y(t)</th>
<th>No. of iterations</th>
<th>t</th>
<th>y(t)</th>
<th>No. of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>n = 10</td>
<td></td>
<td></td>
<td>n = 100</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Simpson's rule</td>
<td></td>
<td></td>
<td>Three-eighths</td>
<td></td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000000000</td>
<td>0</td>
<td>0.0000</td>
<td>0.0000000000</td>
<td>0</td>
</tr>
<tr>
<td>0.1000</td>
<td>0.100507614</td>
<td>6</td>
<td>0.1000</td>
<td>0.1000000005</td>
<td>4</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.200020718</td>
<td>6</td>
<td>0.2000</td>
<td>0.200000020</td>
<td>4</td>
</tr>
<tr>
<td>0.3000</td>
<td>0.300545906</td>
<td>7</td>
<td>0.3000</td>
<td>0.300000048</td>
<td>4</td>
</tr>
<tr>
<td>0.4000</td>
<td>0.400090252</td>
<td>7</td>
<td>0.4000</td>
<td>0.400000090</td>
<td>4</td>
</tr>
<tr>
<td>0.5000</td>
<td>0.500643964</td>
<td>9</td>
<td>0.5000</td>
<td>0.500000151</td>
<td>5</td>
</tr>
<tr>
<td>0.6000</td>
<td>0.601361876</td>
<td>9</td>
<td>0.6000</td>
<td>0.600001259</td>
<td>5</td>
</tr>
<tr>
<td>0.7000</td>
<td>0.702392784</td>
<td>11</td>
<td>0.7000</td>
<td>0.700002023</td>
<td>5</td>
</tr>
<tr>
<td>0.8000</td>
<td>0.802387872</td>
<td>10</td>
<td>0.8000</td>
<td>0.800003199</td>
<td>5</td>
</tr>
<tr>
<td>0.9000</td>
<td>0.905015244</td>
<td>12</td>
<td>0.9000</td>
<td>0.900005039</td>
<td>5</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.006487016</td>
<td>11</td>
<td>1.0000</td>
<td>1.000008198</td>
<td>5</td>
</tr>
</tbody>
</table>


\[ y(t) = \begin{cases} 
    f(t) + \int_0^t k(t-\sigma) G(y(\sigma-\tau), \varphi(\sigma)) \, d\sigma & \text{for } t > 0 \\
    g(t) & \text{for } t < 0,
\end{cases} \tag{4.4} \]

where the kernel \( k(t-\sigma) \) is of the form

\[ k(t-\sigma) = \frac{1}{\sqrt{\pi(t-\sigma)}}. \]

We therefore considered an example of this form where the answer was known.
Example 2.

\[
y(x) = \begin{cases} 
  f(x) + \int_0^x \frac{G(y(t-\tau(y(t))), \varphi(t))}{\sqrt{\pi(x-t)}} dt, & x \geq 0 \\
  0, & x < 0,
\end{cases}
\]

where \(G(u, v) = u - v, \varphi(t) \equiv 1, \tau(u)\) is as in Example 1,

\[
f(x) = x^2 - \int_0^x \frac{\rho(t) - 1}{\sqrt{\pi(x-t)}} dt
\]

and

\[
\rho(t) = \begin{cases} 
  0 & \text{for } t - \tau(t^2) < 0 \\
  \left[ t - \tau(t^2) \right]^2 & \text{for } t - \tau(t^2) \geq 0.
\end{cases}
\]

### TABLE III

Solutions to Example 2 using Method 1

<table>
<thead>
<tr>
<th>(n = 10)</th>
<th>(n = 100)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(t)</td>
<td>(y(t))</td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.0400000000</td>
</tr>
<tr>
<td>0.4000</td>
<td>0.1600000000</td>
</tr>
<tr>
<td>0.6000</td>
<td>0.3600000000</td>
</tr>
<tr>
<td>0.8000</td>
<td>0.644399435</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.011256792</td>
</tr>
<tr>
<td>1.2000</td>
<td>1.446823980</td>
</tr>
<tr>
<td>1.4000</td>
<td>1.974380612</td>
</tr>
<tr>
<td>1.6000</td>
<td>2.573330831</td>
</tr>
<tr>
<td>1.8000</td>
<td>3.259040943</td>
</tr>
<tr>
<td>2.0000</td>
<td>4.021918185</td>
</tr>
<tr>
<td>(q = 3)</td>
<td>(q = 3)</td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.0400000000</td>
</tr>
<tr>
<td>0.4000</td>
<td>0.1600000000</td>
</tr>
<tr>
<td>0.6000</td>
<td>0.3600000000</td>
</tr>
<tr>
<td>0.8000</td>
<td>0.644399435</td>
</tr>
<tr>
<td>1.0000</td>
<td>0.998994447</td>
</tr>
<tr>
<td>1.2000</td>
<td>1.437950810</td>
</tr>
<tr>
<td>1.4000</td>
<td>1.957875710</td>
</tr>
<tr>
<td>1.6000</td>
<td>2.557277035</td>
</tr>
<tr>
<td>1.8000</td>
<td>3.237215583</td>
</tr>
<tr>
<td>2.0000</td>
<td>3.996483416</td>
</tr>
</tbody>
</table>
The solution is the function $s$ where

$$s(x) = \begin{cases} x^2 & \text{for } x \geq 0 \\ 0 & \text{for } x < 0. \end{cases}$$

To solve this equation we use Atkinson's technique [16] and replace $G(y(t - \tau(y(t))), q(t))$ with the piecewise Lagrange polynomial of degree $q$ or less on the interval in question. We then evaluate the integral $\int_0^t \sqrt{\pi(x-t)} \, dt$ exactly. Explicit formulas for the weight functions thus generated can be found in [17]. Tables III and IV give results of solving this equation using methods 1 and 2 with values of 2 and 3 for $q$. In both of these examples we used the delay function

$$\tau(t) = e^{-t} + 0.01.$$  

### Table IV

Solutions to Example 2 using Method 2

<table>
<thead>
<tr>
<th>$n = 10$</th>
<th>$n = 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>$y(t)$</td>
</tr>
<tr>
<td>$q = 2$</td>
<td></td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.0400000000</td>
</tr>
<tr>
<td>0.4000</td>
<td>0.1600000000</td>
</tr>
<tr>
<td>0.6000</td>
<td>0.3600000000</td>
</tr>
<tr>
<td>0.8000</td>
<td>0.644302496</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.011883947</td>
</tr>
<tr>
<td>1.2000</td>
<td>1.446623741</td>
</tr>
<tr>
<td>1.4000</td>
<td>1.971946504</td>
</tr>
<tr>
<td>1.6000</td>
<td>2.572476221</td>
</tr>
<tr>
<td>1.8000</td>
<td>3.258742832</td>
</tr>
<tr>
<td>2.0000</td>
<td>4.020172390</td>
</tr>
<tr>
<td>$q = 3$</td>
<td></td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>0.2000</td>
<td>0.0400000000</td>
</tr>
<tr>
<td>0.4000</td>
<td>0.1600000000</td>
</tr>
<tr>
<td>0.6000</td>
<td>0.3600000000</td>
</tr>
<tr>
<td>0.8000</td>
<td>0.644302496</td>
</tr>
<tr>
<td>1.0000</td>
<td>0.999487795</td>
</tr>
<tr>
<td>1.2000</td>
<td>1.439439543</td>
</tr>
<tr>
<td>1.4000</td>
<td>1.959310875</td>
</tr>
<tr>
<td>1.6000</td>
<td>2.558831504</td>
</tr>
<tr>
<td>1.8000</td>
<td>3.239054747</td>
</tr>
<tr>
<td>2.0000</td>
<td>3.998334158</td>
</tr>
</tbody>
</table>
The function \( t - \tau(t) \) has one root, between 0.5 and 0.6 and the effect of the delay on the numerical solutions can be observed in Tables I and II. Similarly \( t - \tau(t^2) = 0 \) for a value of \( t \) between 0.6 and 0.7 and the effects of the delay can be seen in Tables III and IV.

For Example 1, method 2, using cubic spline interpolation is clearly more accurate. For Example 2, method 1 seems to be more accurate but only marginally so. In terms of actual CPU time, method 2 was faster than method 1 and seemed to be more stable (the cubic version of method 1 blew up on \([0, 33\) with \(n = 100\)). In general then, we recommend method 2.

For problems similar to Example 1, the quadratic integration techniques (Simpson's Rule) appears more accurate than the cubic (three-eights rule) which might be expected considering the error terms for these two integration techniques. For problems similar to Example 2, the cubic integration technique is more accurate. We recommend that the choice of integration technique be based on the type of problem considered.

REFERENCES

