Thai Journal of Mathematics
Volume 21 Number 2 (2023)
Pages 299–314

http://thaijmath.in.cmu.ac.th

ISSN 1686-0209

Trigonometric Spline Method for the Numerical Solutions of Singularly Perturbed Volterra Integro-Differential Equations

Sara Ivany and Reza Jalilian*

Department of Mathematics, Razi University, Tagh Bostan, Kermanshah, P.O. Box 6714967346, Iran
e-mail : s.eivany@yahoo.com (S. Ivany);
rezajalilian72@gmail.com, r.jalilian@razi.ac.ir (R. Jalilian)

Abstract In this study, the trigonometric spline function is presented for solving a singularly perturbed initial value problem for a linear first-order Volterra integro-differential equation. By this idea integro-differential equation will be changed into the system of algebraic equation. The relations and the error analysis of trigonometric spline function are derived. The convergence analysis of the method discussed. Some examples are included to demonstrate the validity and applicability of the technique and to compare the computed results with the existing methods.

MSC: 41A15; 65RXX; 65D07; 34K05; 34K30
Keywords: trigonometric spline; singularly perturbed; Volterra integro-differential equation; convergence

Submission date: 20.05.2021 / Acceptance date: 21.10.2021

1. Introduction

In this study, we apply trigonometric spline function to develop numerical solution of the singularly perturbed Volterra integro-differential equations (SPVIDEs) :

$$\begin{cases} \epsilon y'(x) + p(x)y(x) + \displaystyle\int_0^x k(x,t)y(t)\,dt = f(x), \; x \in I := [0,\Omega], \\ y(0) = y_0, \end{cases} \tag{1.1}$$

where ϵ, $(\epsilon \in (0,1])$ is the perturbation parameter. Here $f(x)$ and $k(x,t)$ are sufficiently smooth functions with $(x,t) \in (I \times I)$, $p(x) \geq \alpha > 0$ for some constant α, and $y(x)$ is the unknown function. By choosing $\epsilon = 0$ in (1.1), we obtain the reduced equation:

$$p(x)y(x) + \int_0^x k(x,t)y(t)\,dt = f(x), \tag{1.2}$$

*Corresponding author.

Published by The Mathematical Association of Thailand.

which is a Volterra integral equation of the second kind. SPVIDEs occur in many chemicals, physical and biological problems. Some of its applications are in diffusion-dissipation processes, epidemic dynamics, synchronous control systems, renewal processes and filament stretching, (see, [1–6]).

A survey of the literature on singularly perturbed Volterra integral and integro-differential equations is given in [7].
In general, singularly perturbed differential equations are usually identified by a small parameter ϵ multiplying some or all of the highest-order terms in the differential equations. In such cases, when the perturbed parameter ϵ tends to zero ($\epsilon \longrightarrow 0$), width length of boundary layer becomes thinner. That's why the behavior of unknown function $y(x)$ in the boundary layer is difficult to simulate numerically, i.e., the solution of (1.1) in the presence of the singular perturbation parameter ϵ varies very rapidly in a thin layer near $x = 0$, compared to the solution of (1.2), (see, also [1, 8]). In recent years, a considerable amount of numerical methods such as exponential techniques [8], finite difference method [9, 10], the Petrov-Galerkin method [11], the spectral method [12], and so on. The results show that classical numerical methods, such as finite volume, finite element, and finite difference do not work well for these problems because they often produce oscillatory solutions which are inaccurate when the perturbed parameter ϵ is small. Also, other numerical methods have been studied to solve this type of problem. For example, a special class of singularly perturbed integro-differential-algebraic equations and singularly perturbed integro-differential systems have been solved by implicit Runge-Kutta methods in [13], and the convergence of the extended implicit Pouzet-Volterra-Runge-Kutta methods applied to singularly perturbed systems of Volterra integro-differential equations analyzed in [14]. The exponential scheme and stability analysis of this scheme is discussed in [15]. The numerical discretization of SPVIDEs and Volterra integral equations by tension spline collocation methods in certain tension spline spaces are considered in [16]. In [17], Bijura analyzed the existence of the initial layers whose thickness is not of the order of magnitude $O(\epsilon)$, $\epsilon \longrightarrow 0$, and constructed approximate solutions using the initial layer. The convergence properties of a difference scheme for nonlinear SPVIDEs on a graded mesh studied in [18]. Studied the convergence properties of a finite difference scheme on a Shishkin mesh for problem (1.1), using the midpoint difference operator and trapezoidal integration in [19], and proved it to be almost second-order accurate, in [1], analyzed and exponentially fitted difference scheme on a uniform mesh for the same problem. Recently, there has been a growing interest in the numerical solution of SPVIDEs. For example, the coupled method for SPVIDEs is discussed by Tao and Zhang in [20], and the uniformly convergent numerical method is used for them by Iragi and Munyakazi in [21].
In this work, we pursue two important aims. The first aim is to explore non-polynomial spline interpolation with multiple parameters and to produce the error of approximate trigonometric spline. The second aim is to introduce a new approximate technique to find solutions of SPVIDEs, and we demonstrate the convergence analysis for this technique. The main advantage of our algorithm is that it can be used directly without using assumption or transformation formulae.

The structure of this paper is organized as follows. In section 2, the trigonometric spline formulation to solve singularly perturbed Volterra integro-differential equation is derived. In section 3, the convergence of the method is investigated. Some numerical results are given to clarify the method in section 4, and the computed results are compared with other known methods in [1], [21] and [20]. Finally, we have a conclusion of our study.

2. Trigonometric Spline Method

Here we study interpolation based on the trigonometric spline. For the existence and uniqueness of trigonometric interpolation see [22] and [23] also for error analysis in trigonometric interpolations see [24]. In [25] developed a scheme based on exponential quartic spline for solving third-order self adjoint singularly perturbed boundary value problems and the parametric cubic spline method was presented for solving Bratus problem in [26]. In [24] trigonometric interpolation introduced of the following form:

$$Q_n(x) = \sum_{i=0}^{n} a_i \phi_i(x), \tag{2.1}$$

where $\{\phi_0(x), \phi_1(x), \phi_2(x), ..., \phi_n(x)\}$ is a set of linearly independent real-valued continuous functions on $[a, b]$, and coefficients $a_0, a_1, a_2, ..., a_n$ are determined by the interpolation conditions. The spline functions presented in this paper, similarly articles [27], [28], [29], [30], [31], [32], [33], and [34], have the following form,

$$Q_n(x) \in \text{span}\{\sin(\omega x), \cos(\omega x), \sin(2\omega x), \cos(2\omega x)\}.$$

We consider a uniform mesh Ω with nodal points x_i on $[a, b]$ such that

$$\Omega : a = x_0 < x_1 < ... < x_n = b,$$

where $h = \frac{(b-a)}{n}$. Let $S_i(x, \omega)$ be the interpolating spline function which interpolate $y(x)$ at x_k, defined on the interval $[x_i, x_{i+1}]$, $i = 0, 1, ..., n-1$, as

$$\begin{cases} S_\Omega(x,\omega) = & a_i \sin\big(\omega\,(x - x_i)\big) + b_i \cos\big(\omega(x - x_i)\big) \\ & +c_i \sin\big(2\omega\,(x - x_i)\big) + d_i \cos\big(2\omega\,(x - x_i)\big), \end{cases} \tag{2.2}$$

where a_i, b_i, c_i and d_i are real and ω is an arbitrary parameter. To derive expression for the coefficients in equations of (2.2) in terms of y_i, y_{i+1}, M_i and M_{i+1}, we first denote

$$\begin{cases} S_\Omega(x_i,\omega) = y_i, & S_\Omega(x_{i+1},\omega) = y_{i+1}, \\ S''_\Omega(x_i,\omega) = M_i, & S''_\Omega(x_{i+1},\omega) = M_{i+1}, \end{cases} \tag{2.3}$$

$$\begin{cases} S_\Omega(x_i,\omega) = y_i, & S_\Omega(x_{i+1},\omega) = y_{i+1}, \\ S'_\Omega(x_i,\omega) = m_i, & S'_\Omega(x_{i+1},\omega) = m_{i+1}. \end{cases} \tag{2.4}$$

By using the equations (2.2) and (2.3), we get the following expressions:

$$\begin{cases} a_i = -\frac{\csc(\theta)}{3\omega^2}\Big((\cos(\theta)M_i - M_{i+1}) + 4\omega^2\,(\cos(\theta)y_i - y_{i+1})\Big), \\ b_i = \frac{1}{3\omega^2}\Big(M_i + 4\omega^2 y_i\Big), \\ c_i = \frac{\csc(2\theta)}{3\omega^2}\Big((\cos(2\theta)M_i - M_{i+1}) + \omega^2\,(\cos(2\theta)y_i - y_{i+1})\Big), \\ d_i = -\frac{1}{3\omega^2}\Big(M_i + \omega^2 y_i\Big). \end{cases} \tag{2.5}$$

From the continuity of the first derivatives of spline functions $S_{i-1}(x, \omega)$ and $S_i(x, \omega)$ at $x = x_i$, we get the following consistency relations:

$$\Big(\alpha_1 M_{i-1} + 2\alpha_2 M_i + \alpha_1 M_{i+1}\Big) = \frac{1}{h^2}\Big(y_{i-1} - 2\alpha_3 y_i + y_{i+1}\Big), i = 1, 2, ..., n-1, \tag{2.6}$$

where

$$\alpha_1 = \frac{1-\sec(\theta)}{\theta^2(-4+\sec(\theta))}, \quad \alpha_2 = \frac{-\sin(\theta)\tan(\theta)}{\theta^2\,(-4+sec(\theta))}, \quad \alpha_3 = \frac{2+\cos(2\theta)}{-1+4\cos(\theta)},$$

and $\theta = \omega h$. If $\theta \longrightarrow 0$, then $(\alpha_1, \alpha_2, \alpha_3) \longrightarrow (\frac{1}{6},\frac{1}{3},1)$. Moreover, assuming $(\alpha_1, \alpha_2, \alpha_3) \longrightarrow (\frac{1}{12}, \frac{5}{12}, 1)$, we get the following relation:

$$\Big(M_{i-1} + 10M_i + M_{i+1}\Big) = \frac{12}{h^2}\Big(y_{i-1} - 2y_i + y_{i+1}\Big), i = 1, 2, ..., n-1. \tag{2.7}$$

By expanding (2.6) in Taylor series about x_i, we obtain the following local truncation error:

$$\begin{cases} T_i = & (2\alpha_3 - 2)y_i + (2\alpha_1 + 2\alpha_2 - 1)\,h^2 y_i'' + \frac{1}{12}(12\alpha_1 - 1)\,h^4 y_i^{(4)} \\ & +\frac{1}{360}(30\alpha_1 - 1)\,h^6 y_i^{(6)} + \frac{1}{20160}(56\alpha_1 - 1)\,h^8 y_i^{(8)} + O(h^{10}). \end{cases} \tag{2.8}$$

Similarly, by using (2.2) and (2.4), we have

$$\Big(\beta_1 m_{i-1} + 2\beta_2 m_i + \beta_1 m_{i+1}\Big) = \frac{\beta_3}{h}\Big(y_{i+1} - y_{i-1}\Big), \; i = 1, 2, ..., n-1, \tag{2.9}$$

where

$$\beta_1 = \frac{3\theta\,\csc^2(\frac{\theta}{2})\sin(\theta)}{(2+\cos(\theta))}, \quad \beta_2 = \frac{3\theta\,\csc^2(\frac{\theta}{2})(2\sin(\theta)+\sin(2\theta))}{2(2+\cos(\theta))}, \quad \beta_3 = \frac{3\theta^2\,\csc^2(\frac{\theta}{2})(1+2\cos(\theta))}{(2+\cos(\theta))}.$$

Now in relation (2.9), if $\theta \longrightarrow 0$, that $(\beta_1, \beta_2, \beta_3) \longrightarrow (\frac{1}{3}, \frac{2}{3}, 1)$ and if let $(\beta_1, \beta_2, \beta_3) \longrightarrow (\frac{1}{6}, \frac{5}{6}, 1)$ we have the following system:

$$\Big(m_{i-1} + 10m_i + m_{i+1}\Big) = \frac{1}{2h}\Big(y_{i+1} - y_{i+1}\Big), \; i = 1, 2, ..., n-1. \tag{2.10}$$

Similarly, by expanding (2.9) in Taylor series about x_i, we obtain the following local truncation error:

$$\begin{cases} T_i = & (2\beta_1 + 2\beta_2 - 2\beta_3)\,hy_i' + \frac{1}{3}(3\beta_1 - \beta_3)\,h^3 y_i^{(3)} + \frac{1}{60}(5\beta_1 - \beta_3)\,h^5 y_i^{(5)} \\ & +\frac{1}{2520}(7\beta_1 - \beta_3)\,h^7 y_i^{(7)} + O(h^9). \end{cases} \tag{2.11}$$

Now, the values M_i and m_i are determined as the unique solutions of linear systems (2.7) and (2.10). We approximated $m_0 = \frac{-3y_0+4y_1-y_2}{2h}$ and $m_n = \frac{y_{n-2}-4y_{n-1}+3y_n}{2h}$, and also, using boundary conditions, we approximate M_i for $i = 0, n$. So the values M_i for $i = 0, 1, 2, ..., n$ are determined as the solutions of the following linear system:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & 10 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 10 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 10 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 10 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ \vdots \\ \vdots \\ M_{n-2} \\ M_{n-1} \\ M_n \end{bmatrix} = \frac{12}{h^2} \begin{bmatrix} y_0 \\ y_0 - 2y_1 + y_2 \\ y_1 - 2y_2 + y_3 \\ \vdots \\ \vdots \\ y_{n-3} - 2y_{n-2} + y_{n-1} \\ y_{n-2} - 2y_{n-1} + y_n \\ y_n \end{bmatrix}. \tag{2.12}$$

And the values m_i for $i = 0, 1, 2, ..., n$ are determined as follows:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ 1 & 10 & 1 & 0 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 10 & 1 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \cdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cdots & 1 & 10 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 1 & 10 & 1 \\ 0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ \vdots \\ \vdots \\ m_{n-2} \\ m_{n-1} \\ m_n \end{bmatrix} = \frac{6}{h} \begin{bmatrix} \frac{-3y_0+4y_1-y_2}{12} \\ y_2 - y_0 \\ y_3 - y_1 \\ \vdots \\ \vdots \\ y_{n-1} - y_{n-3} \\ y_n - y_{n-2} \\ \frac{y_{n-2}-4y_{n-1}+3y_n}{12} \end{bmatrix}. \quad (2.13)$$

Briefly, relations (2.12) and (2.13) can be written as follows, in which the matrix W is strictly diagonally-dominant matrix, then the matrix W is invertible. Hence,

$$\begin{cases} h^2 WM = RY \quad \Longrightarrow \quad M = \dfrac{1}{h^2} W^{-1} RY, \\ hWm = SY \quad \Longrightarrow \quad m = \dfrac{1}{h} W^{-1} SY. \end{cases} \quad (2.14)$$

From (2.2) and (2.3) we have

$$\begin{aligned} &S_i(x, \omega) = \\ &\frac{\sin\big(\omega(x - x_i)\big)}{3\omega^2}\Big(- \cot(\theta)M_i + \csc(\theta)M_{i+1} - 4\omega^2 \cot(\theta)y_i + 4\omega^2 \csc(\theta)y_{i+1}\Big) \\ &+ \frac{\cos\big(\omega(x - x_i)\big)}{3\omega^2}\Big(M_i + 4\omega^2 y_i\Big) \\ &+ \frac{\sin\big(2\omega(x - x_i)\big)}{3\omega^2}\Big(\cot(2\theta)M_i - \csc(2\theta)M_{i+1} + \omega^2 \cot(2\theta)y_i - \omega^2 \csc(2\theta)y_{i+1}\Big) \\ &+ \frac{\cos\big(2\omega(x - x_i)\big)}{3\omega^2}\Big(- M_i - \omega^2 y_i\Big). \end{aligned}$$

The discretization of Equations (1.1), with the help of equation (2.2), is as follows

$$\begin{aligned} \epsilon y_i' + a_i y_i = f_i + \int_0^x k(x_i, t)y(t)\,dt \\ &= f_i + \sum_{j=0}^{n-1} \int_{t_j}^{t_{j+1}} k(x_i, t)y(t)\,dt \\ &\cong f_i + \sum_{j=0}^{n-1} \int_{t_j}^{t_{j+1}} k(x_i, t)S_j(t)\,dt \\ &= f_i - \sum_{j=0}^{n-1} \frac{\cot(\theta)M_j + 4\omega^2 \cot(\theta)y_j}{3\omega^2} \int_{t_j}^{t_{j+1}} \sin\big(\omega(t - t_j)\big)k(x_i, t)\,dt \end{aligned}$$

$$
\begin{aligned}
&+\sum_{j=0}^{n-1}\frac{csc(\theta)M_{j+1}+4\omega^2\csc(\theta)y_{j+1}}{3\omega^2}\int_{t_j}^{t_{j+1}}\sin\big(\omega(t-t_j)\big)k(x_i,t)dt\\
&+\sum_{j=0}^{n-1}\frac{M_j+4\omega^2y_j}{3\omega^2}\int_{t_j}^{t_{j+1}}\cos\big(\omega(t-t_j)\big)k(x_i,t)dt\\
&+\sum_{j=0}^{n-1}\frac{\cot(2\theta)M_j+\omega^2\cot(2\theta)y_j}{3\omega^2}\int_{t_j}^{t_{j+1}}\sin\big(2\omega(t-t_j)\big)k(x_i,t)dt\\
&-\sum_{j=0}^{n-1}\frac{\csc(2\theta)M_{j+1}+\omega^2\csc(2\theta)y_{j+1}}{3\omega^2}\int_{t_j}^{t_{j+1}}\sin\big(2\omega(t-t_j)\big)k(x_i,t)dt\\
&-\sum_{j=0}^{n-1}\frac{M_j+\omega^2y_j}{3\omega^2}\int_{t_j}^{t_{j+1}}\cos\big(2\omega(t-t_j)\big)k(x_i,t)dt,
\end{aligned}
$$

assuming

$$
\begin{cases}
a(i,j)=\displaystyle\int_{t_j}^{t_{j+1}}\sin\big(\omega(t-t_j)\big)k(x_i,t)dt,\\
b(i,j+1)=\displaystyle\int_{t_j}^{t_{j+1}}\sin\big(\omega(t-t_j)\big)k(x_i,t)dt,\\
c(i,j)=\displaystyle\int_{t_j}^{t_{j+1}}\cos\big(\omega(t-t_j)\big)k(x_i,t)dt,\\
d(i,j)=\displaystyle\int_{t_j}^{t_{j+1}}\sin\big(2\omega(t-t_j)\big)k(x_i,t)dt,\\
e(i,j+1)=\displaystyle\int_{t_j}^{t_{j+1}}\sin\big(2\omega(t-t_j)\big)k(x_i,t)dt,\\
f(i,j)=\displaystyle\int_{t_j}^{t_{j+1}}\cos\big(2\omega(t-t_j)\big)k(x_i,t)dt.
\end{cases}
$$

and introduce the following relations:

$$
\begin{cases}
a(i,n)=0, & b(i,0)=0, \quad c(i,n)=0,\\
d(i,n)=0, & e(i,0)=0, \quad f(i,n)=0.
\end{cases}
$$

The above notations can be written in the matrix-vector form: $A=(a_{i,j})$, $B=(b_{i,j})$, $C=(c_{i,j})$, $D=(d_{i,j})$, $E=(e_{i,j})$, $F=(f_{i,j})$, $P=(p_{i,j})$ also if suppose

$$
\begin{gathered}
M\approx\hat{M}=(\hat{M_0},\hat{M_1},...,\hat{M_{n-1}},\hat{M_n})^T,\\
Y\approx\hat{Y}=(\hat{Y_0},\hat{Y_1},...,\hat{Y_{n-1}},\hat{y_n})^T,\\
m\approx\hat{m}=(\hat{m_0},\hat{m_1},...,\hat{m_{n-1}},\hat{m_n})^T,\\
\hat{F}=(\hat{f_0},\hat{f_1},...,\hat{f_{n-1}},\hat{f_n})^T.
\end{gathered}
$$

After substitution, we get

$$\begin{aligned}\epsilon\hat{m}+P\hat{Y}=\hat{F}&+\frac{-\cot(\theta)}{3\omega^2}A\hat{M}-\frac{-4\cot(\theta)}{3}A\hat{Y}+\frac{\csc(\theta)}{3\omega^2}B\hat{M}+\frac{4csc(\theta)}{3}B\hat{Y}\\&+\frac{1}{3\omega^2}C\hat{M}+\frac{4}{3}C\hat{Y}+\frac{\cot(2\theta)}{3\omega^2}D\hat{M}+\frac{\omega^2\cot(2\theta)}{3\omega^2}D\hat{Y}-\frac{csc(2\theta)}{3\omega^2}E\hat{M}\\&-\frac{\csc(2\theta)}{3}E\hat{Y}-\frac{1}{3\omega^2}F\hat{M}-\frac{1}{3}F\hat{Y}.\end{aligned}\tag{2.15}$$

By solving the above system, an approximation solution of the equation (1.1) can be obtained. Now, the y_i function can be approximated using the non-polynomial spline $\hat{S}_i(x,\omega)$, where

$$\begin{aligned}&\hat{S}_i(x,\omega)=\\&\frac{\sin\big(\omega(x-x_i)\big)}{3\omega^2}\Big(-\cot(\theta)\hat{M}_i+\csc(\theta)\hat{M_{i+1}}-4\omega^2\cot(\theta)\hat{y_i}+4\omega^2\csc(\theta)\hat{y_{i+1}}\Big)\\&+\frac{\cos\big(\omega(x-x_i)\big)}{3\omega^2}\Big(\hat{M}_i+4\omega^2\hat{y_i}\Big)\\&+\frac{\sin\big(2\omega(x-x_i)\big)}{3\omega^2}\Big(\cot(2\theta)\hat{M}_i-\csc(2\theta)\hat{M_{i+1}}+\omega^2\cot(2\theta)\hat{y_i}-\omega^2\csc(2\theta)\hat{y_{i+1}}\Big)\\&+\frac{\cos\big(2\omega(x-x_i)\big)}{3\omega^2}\Big(-\hat{M}_i-\omega^2\hat{y_i}\Big)\\&+O(h^4).\end{aligned}\tag{2.16}$$

3. Convergence of the Method

We investigate the convergence analysis for the developed method. The equation (2.15) can be written in the matrix-vector form:

$$\begin{aligned}&\epsilon\hat{m}+P\hat{Y}+\Big(\frac{\cot(\theta)}{3\omega^2}A-\frac{csc(\theta)}{3\omega^2}B-\frac{1}{3\omega^2}C-\frac{\cot(2\theta)}{3\omega^2}D+\frac{\csc(2\theta)}{3\omega^2}E+\frac{1}{3\omega^2}F\Big)\hat{M}\\&+\Big(-\frac{4\cot(\theta)}{3}A-\frac{4\csc(\theta)}{3}B-\frac{4}{3}C-\frac{\omega^2\cot(2\theta)}{3\omega^2}D+\frac{\csc(2\theta)}{3}E+\frac{1}{3}F\Big)\hat{Y}\\&=\hat{F}.\end{aligned}\tag{3.1}$$

Assuming that

$$\begin{aligned}H_1&=\Big(\frac{\cot(\theta)}{3\omega^2}A-\frac{\csc(\theta)}{3\omega^2}B-\frac{1}{3\omega^2}C-\frac{\cot(2\theta)}{3\omega^2}D+\frac{\csc(2\theta)}{3\omega^2}E+\frac{1}{3\omega^2}F\Big),\\H_2&=\Big(-\frac{4\cot(\theta)}{3}A-\frac{4\csc(\theta)}{3}B-\frac{4}{3}C-\frac{\omega^2\cot(2\theta)}{3\omega^2}D+\frac{\csc(2\theta)}{3}E+\frac{1}{3}F\Big).\end{aligned}$$

So using the equation (3.1) we have the following expressions

$$\begin{aligned}&\epsilon\hat{m}+P\hat{Y}+H_1\hat{M}+H_2\hat{Y}=\hat{F},\\&\Big(\epsilon(\frac{1}{h}W^{-1}S)+P+H_1(\frac{1}{h^2}W^{-1}R)+H_2\Big)\hat{Y}=\hat{F},\end{aligned}\tag{3.2}$$

Now, considering the above system with the exact solution $\bar{Y} = y(x_i), i = 0, 1, ..., n$, we get:

$$\Big(\frac{1}{h}\epsilon W^{-1}S + P + \frac{1}{h^2}H_1W^{-1}R + H_2\Big)\bar{Y} = \hat{F} + \bar{T}, \tag{3.3}$$

Where $\bar{T} = [t_0, t_1, ..., t_n]^T$ is the local truncation error vector. By applying (3.2) and (3.3), gives

$$P\Big(I + P^{-1}(\frac{1}{h}\epsilon W^{-1}S + \frac{1}{h^2}H_1W^{-1}R + H_2)\Big)\tilde{E} = \bar{T}. \tag{3.4}$$

,

Lemma 3.1. *Let N is a $(n \times n)$ matrix with $\|N\|_\infty < 1$, so the matrix $(I - N)$ is invertible. Moreover $\|(I - N)^{-1}\|_\infty < \frac{1}{1-\|N\|_\infty}$.*

Lemma 3.2. *The matrix W is invertible.*

Proof. See [35]. ■

Note that if P be a diagonal matrix with the inverse P^{-1} then $\|P^{-1}\|_\infty < \frac{1}{min|p_{i,i}|} = \frac{1}{\lambda}$, for $i = 0, 1, ..., n$. Now we need to show that inverse of $\Big[I + P^{-1}(\frac{\epsilon}{h}W^{-1}S + \frac{1}{h^2}H_1W^{-1}R + H_2)\Big]$ exists.

Lemma 3.3. *The matrix $\Big[I + P^{-1}(\frac{\epsilon}{h}W^{-1}S + \frac{1}{h^2}H_1W^{-1}R + H_2)\Big]$ is nonsingular. Provided*

$$\frac{1}{\lambda}\Big(\frac{12\epsilon}{h} + \|k\|\,(b-a)\,\eta_1 + \|k\|\,(b-a)\,\frac{\eta_2}{12}\Big) < 1,$$

where

$$\eta_1 = \frac{2\cos(3\theta) - 2\cos(5\theta) - 8\theta\cos(2\theta) + 8\theta\cos(4\theta) + \theta^2\cos(2\theta) - \theta^2\cos(6\theta)}{2\theta^3\,sin(2\theta)\cos(2\theta)},$$

$$\eta_2 = \frac{-8\theta + 8\cos(3\theta) - 8\cos(5\theta) - 32\theta\cos(2\theta) + 40\theta\cos(2\theta)^2 + \theta^2\cos(2\theta) - 2\theta^2 cos(3\theta)^2 + \theta^2}{2\theta\, sin(2\theta) cos(2\theta)}.$$

Proof. It can be verified as :

$$\|A\| = \|B\| \le \|k\| n \int_{t_j}^{t_{j+1}} \sin\big(\omega(t - t_j)\big)\,dt = \|k\|\,|\frac{n}{\omega}|\Big[\cos\big(\omega(t - t_j)\big)\Big]_{t_j}^{t_{j+1}} =$$

$$\|k\|\,|\frac{n}{\omega}|\Big(\cos(\omega h) - \cos(0)\Big) = \|k\|\frac{n\Big(\cos(\theta) - 1\Big)}{\omega} = \|k\|\frac{nh\Big(\cos(\theta) - 1\Big)}{\theta},$$

$$\|C\| \le \|k\| n \int_{t_j}^{t_{j+1}} \cos\big(\omega(t - t_j)\big)\,dt = \|k\|\frac{n}{\omega}\Big[\sin\big(\omega(t - t_j)\big)\Big]_{t_j}^{t_{j+1}} =$$

$$\|k\|\frac{n}{\omega}\Big(\sin(\omega h) - \sin(0)\Big) = \|k\|\frac{n\sin(\theta)}{\omega} = \|k\|\frac{nh\sin(\theta)}{\theta},$$

$$\|D\| = \|E\| \le \|k\| n \int_{t_j}^{t_{j+1}} sin\big(2\omega(t - t_j)\big)\,dt = \|k\|\,|\frac{n}{2\omega}|\Big[\cos\big(2\omega(t - t_j)\big)\Big]_{t_j}^{t_{j+1}} =$$

$$\|k\|\,|\frac{n}{2\omega}|\Big(\cos(2\omega h) - cos(0)\Big) = \|k\|\frac{n\Big(\cos(2\theta) - 1\Big)}{2\omega} = \|k\|\frac{nh\Big(\cos(2\theta) - 1\Big)}{2\theta},$$

$$\|F\| \le \|k\|\, n \int_{t_j}^{t_{j+1}} cos\big(2\omega(t-t_j)\big)\, dt = \|k\| \frac{n}{2\omega} \Big[sin\big(2\omega(t-t_j)\big)\Big]_{t_j}^{t_{j+1}} =$$

$$\|k\| \frac{n}{2\omega} \Big(\sin(2\omega h) - \sin(0)\Big) = \|k\| \frac{n\sin(\theta)}{2\omega} = \|k\| \frac{nh\,\sin(2\theta)}{2\theta},$$

as a result, we can show that

$\|A\| = \|B\| \le \|k\| \big(b-a\big) \frac{\big(\cos(\theta)-1\big)}{\theta}$, $\|C\| \le \|k\| \big(b-a\big) \frac{\big(\sin(\theta)\big)}{\theta}$ and $\|D\| = \|E\| \le \|k\| \big(b-a\big) \frac{\big(\cos(2\theta)-1\big)}{2\theta}$ and also $\|F\| \le \|k\| \big(b-a\big) \frac{\big(\sin(2\theta)\big)}{2\theta}$,

hence

$$\begin{aligned}\|H_1\| \le\ & |\tfrac{\cot(\theta)}{3\omega^2}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(\theta)-1\big)}{\theta}\Big) + |\tfrac{\csc(\theta)}{3\omega^2}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(\theta)-1\big)}{\theta}\Big)\\ & + |\tfrac{1}{3\omega^2}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\sin(\theta)\big)}{\theta}\Big) + |\tfrac{\cot(2\theta)}{3\omega^2}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(2\theta)-1\big)}{2\theta}\Big)\\ & + |\tfrac{\csc(2\theta)}{3\omega^2}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(2\theta)-1\big)}{2\theta}\Big) + |\tfrac{1}{3\omega^2}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\sin(2\theta)\big)}{2\theta}\Big),\end{aligned}$$

and

$$\begin{aligned}\|H_2\| \le\ & |\tfrac{4\cot(\theta)}{3}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(\theta)-1\big)}{\theta}\Big) + |\tfrac{4\csc(\theta)}{3}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(\theta)-1\big)}{\theta}\Big)\\ & + |\tfrac{4}{3}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\sin(\theta)\big)}{\theta}\Big) + |\tfrac{\cot(2\theta)}{3}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(2\theta)-1\big)}{2\theta}\Big)\\ & + |\tfrac{\csc(2\theta)}{3}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\cos(2\theta)-1\big)}{2\theta}\Big) + |\tfrac{1}{3}| \Big(\|k\| \big(b-a\big) \tfrac{\big(\sin(2\theta)\big)}{2\theta}\Big),\end{aligned}$$

after factoring and simplifying, we have

$$\begin{aligned}\|H_1\| \le\ & \Big(\|k\| \tfrac{\big(b-a\big)}{3\omega^2}\Big) \Big[cot(\theta)\Big(\tfrac{cos(\theta)-1}{\theta}\Big) + csc(\theta)\Big(\tfrac{cos(\theta)-1}{\theta}\Big) + \Big(\tfrac{sin(\theta)}{\theta}\Big)\\ & + cot(2\theta)\Big(\tfrac{cos(2\theta)-1}{2\theta}\Big) + \csc(2\theta)\Big(\tfrac{\cos(2\theta)-1}{2\theta}\Big) + \Big(\tfrac{\sin(2\theta)}{2\theta}\Big)\Big],\end{aligned}$$

$$\|H_1\| \le \|k\| \frac{\big(b-a\big)h^2}{12} \Big[\frac{2\cos(3\theta)-2\cos(5\theta)-8\theta\cos(2\theta)+8\theta\cos(4\theta)+\theta^2\cos(2\theta)-\theta^2\cos(6\theta)}{2\theta^3\sin(2\theta)\cos(2\theta)}\Big],$$

and

$$\begin{aligned}\|H_2\| \le\ & \Big(\|k\| \tfrac{\big(b-a\big)}{3}\Big) \Big[4cot(\theta)\Big(\tfrac{cos(\theta)-1}{\theta}\Big) + 4csc(\theta)\Big(\tfrac{cos(\theta)-1}{\theta}\Big) + 4\Big(\tfrac{sin(\theta)}{\theta}\Big)\\ & + cot(2\theta)\Big(\tfrac{cos(2\theta)-1}{2\theta}\Big) + \csc(2\theta)\Big(\tfrac{\cos(2\theta)-1}{2\theta}\Big) + \Big(\tfrac{\sin(2\theta)}{2\theta}\Big)\Big],\end{aligned}$$

$$\|H_2\| \le \|k\| \frac{\big(b-a\big)}{12} \Big[\frac{-8\theta+8\cos(3\theta)-8\cos(5\theta)-32\theta\cos(2\theta)+40\theta\cos(2\theta)^2+\theta^2\cos(2\theta)-2\theta^2\cos(3\theta)^2+\theta^2}{2\theta\sin(2\theta)\cos(2\theta)}\Big],$$

that by definition

$$\eta_1 = \frac{2\cos(3\theta)-2\cos(5\theta)-8\theta\cos(2\theta)+8\theta\cos(4\theta)+\theta^2\cos(2\theta)-\theta^2\ cos(6\theta)}{2\theta^3\sin(2\theta)\cos(2\theta)},$$

$$\eta_2 = \frac{-8\theta+8\cos(3\theta)-8\cos(5\theta)-32\theta cos(2\theta)+40\theta cos(2\theta)^2+\theta^2\cos(2\theta)-2\theta^2\cos(3\theta)^2+\theta^2}{2\theta\sin(2\theta)\cos(2\theta)},$$

we have

$$\|H_1\| \leq \|k\| \frac{(b-a)h^2}{12}\eta_1, \|H_2\| \leq \|k\| \frac{(b-a)}{12}\eta_2. \qquad \blacksquare$$

Note that $\|W^{-1}\| \leq 1$, $\|S\| \leq 12$ and $\|R\| \leq 48$. By using the lemma 3.1, the matrix $\Big[I + P^{-1}(\frac{1}{h}\epsilon W^{-1}S + \frac{1}{h^2}H_1W^{-1}R + H_2)\Big]$ is nonsingular, provided that

$$\|P^{-1}(\frac{1}{h}\epsilon W^{-1}S + \frac{1}{h^2}H_1W^{-1}R + H_2)\Big\| < 1,$$
$$\frac{1}{\lambda}\Big(\frac{12\epsilon}{h} + 4\|k\|(b-a)\eta_1 + \|k\|(b-a)\frac{\eta_2}{12}\Big) < 1.$$

Theorem 3.4. *Let $f(x) \in C^4(I)$, and $k(x,t) \in C^4(I \times I)$, such that*

$$\frac{1}{\lambda}\Big(\frac{12\epsilon}{h} + 4\|k\|(b-a)\eta_1 + \|k\|(b-a)\frac{\eta_2}{12}\Big) < 1,$$

then the error $\tilde{E}$ satisfies

$$\|\tilde{E}\|_{\infty,\Delta} \leq \xi h^2, \forall \Delta \in I,$$

where ξ and $I := [a,b]$, are constants.

Proof. Using (3.4) and Lemma 3.1 give the following relations:

$$\|\tilde{E}\| \leq \frac{\|P^{-1}\|\,\|\bar{T}\|}{1 - \|P^{-1}\Big(\frac{1}{h}\epsilon W^{-1}S + \frac{1}{h^2}H_1W^{-1}R + H_2\Big)\|}, \tag{3.5}$$

such that

$$\frac{1}{\lambda}\Big(\frac{12\epsilon}{h} + 4\|k\|(b-a)\eta_1 + \|k\|(b-a)\frac{\eta_2}{12}\Big) < 1,$$

by substituting $\|P^{-1}\|$, $\|T\|$, $\|H_1\|$, $\|H_2\|$, $\|W^{-1}\|$, $\|S\|$, and $\|R\|$, in (3.5) we get

$$\|\tilde{E}\| \leq \frac{\|P^{-1}\|\,\|\bar{T}\|}{1 - \|P^{-1}\|\Big(|\frac{\epsilon}{h}|\,\|W^{-1}\|\,\|S\| + |\frac{1}{h^2}|\,\|H_1\|\,\|W^{-1}\|\,\|R\| + \|H_2\|\Big)}$$

$$= \frac{\frac{1}{\lambda}\frac{h^2}{12}y^{(2)}}{1 - \frac{1}{\lambda}\Big(\frac{12\epsilon}{h} + 4\|k\|(b-a)\eta_1 + \|k\|(b-a)\frac{\eta_2}{12}\Big)} \equiv O(h^2). \qquad \blacksquare$$

4. Numerical Experiments

In this section, we demonstrate the order of convergence of the trigonometric spline method for the singularly perturbed Volterra integro differential equations (1.1) by two numerical examples. To this end, we implement the numerical scheme (2.15) to solve these examples, with different values of α_i, β_i and $h = \frac{1}{n}$, $n = 2^k$, $k = 4,5,6,7$. Also we consider $\epsilon = 10^p$ for different values of p. The maximum absolute errors and absolute errors in solutions of the method are tabulated in tables and the convergence curves of the numerical solutions in the $L_2([0,1])$ norm are drawn in figures. We compare our method with methods in [1], [21] and [20]. The absolute error, maximum absolute error and L_2 norm at all mesh points are calculated by using the following formulas, respectively.

$$e(h) = \big|Y - \hat{Y}\big|,$$
$$E(h) = \|Y - \hat{Y}\|_{L_\infty[0,1]} = \max_{0\le i\le n} \big|y(x_i) - \hat{y}(x_i)\big|,$$
$$\|Y - \hat{Y}\|_{L_2[0,1]} = \sqrt{\sum_{i=0}^{n} \Big(y(x_i) - \hat{y}(x_i)\Big)^2}.$$

The convergence ratio (C.R.) is obtained as follows:

$$C.R. = log_2\Big(\frac{E(h)}{E(\frac{h}{2})}\Big).$$

All the numerical solutions are obtained by using **MATLAB R2013a** software.

Example 4.1. Consider the following SPVIDE in [1] and [21]:

$$\epsilon y'(x) + y(x) + \int_0^x k(x,t)y(t)dt = f(x),\ x \in [0,1],$$

where $k(x,t) = t$, $f(x) = (2+9\epsilon+\epsilon x+11x+x^2)\exp(-x) - 10(\epsilon x+\epsilon^2)\exp(\frac{-x}{\epsilon}) + (5x^2 + 10\epsilon^2 - 2)$, and $y(0) = 10$ is the initial condition. The exact solution is:

$$y(x) = 10 - (10 + x)\exp(-x) + 10\exp(\frac{-x}{\epsilon}).$$

Values of $E(h)$, $C.R.$ and $e(h)$ for this problem are presented in **Table 1.** and **Table 2.** Here we compare our method with the method in [1] and [21] in **Table 3.** Also **Fig.1** demonstrate the convergence order curve of the numerical solution.

Table 1. Maximum errors and rates of convergence for **Example 1.**

	n = 16	n = 32	n = 64	n = 128	n = 256
$\epsilon = 10^{-4}$	4.4012e-03	1.2500e-03	3.2125e-04	7.6938e-05	1.7584e-05
C.R.		1.80	1.96	2.06	2.13
$\epsilon = 10^{-5}$	4.3000e-03	1.1000e-03	2.7107e-04	6.2922e-05	1.4178e-05
C.R.		1.97	2.02	2.10	2.15
$\epsilon = 10^{-6}$	4.1000e-03	1.0000e-03	2.2796e-04	4.6830e-05	1.0058e-05
C.R.		2.01	2.15	2.28	2.21
$\epsilon = 10^{-7}$	4.1000e-03	1.0000e-03	2.2796e-04	4.8757e-05	9.0290e-06
C.R.		2.01	2.15	2.22	2.43
$\epsilon = 10^{-8}$	4.1000e-03	1.0000e-03	2.5422e-04	5.8483e-05	8.9707e-06
C.R.		2.01	2.01	2.12	2.70

TABLE 2. The numerical results of **Example 1.** for $n = 256$, $\epsilon = 10^{-8}$

x	Exact Solution	Approximating Solution	Obtained absolute error
0.125	1.064719	1.06471	0.000008774156
0.250	2.017292	2.01728	0.000008462125
0.375	2.869374	2.86937	0.000008380148
0.500	3.631428	3.63142	0.000007607779
0.625	4.312847	4.31284	0.000006837392
0.750	4.922060	4.92205	0.000007012775
0.875	5.466626	5.46662	0.000006401281

TABLE 3. Comparison of the maximum errors obtained by our methods and those in [1] and [21], for **Example 1.**

		n = 16	n = 32	n = 64	n = 128
$\epsilon = 10^{-4}$	Results in [1]	0.269083	0.136862	0.068693	0.034077
	Results in [21]	4.14e-03	2.07e-03	1.03e-03	6.5e-04
	Results using our method	4.4012e-03	1.250e-03	3.2125e-04	7.69380e-05
$\epsilon = 10^{-5}$	Results in [1]	0.269870	0.137660	0.069498	0.034884
	Results in [21]	4.14e-03	2.07e-03	1.03e-03	6.5e-04
	Results using our method	4.300e-03	1.100e-03	2.7107e-04	6.2922e-05
$\epsilon = 10^{-6}$	Results in [1]	0.269948	0.137740	0.069578	0.034965
	Results in [21]	4.14e-03	2.07e-03	1.03e-03	6.5e-04
	Results using our method	4.100e-03	1.000e-03	2.2796e-04	4.6830e-05
$\epsilon = 10^{-7}$	Results in [1]	0.269956	0.137748	0.069586	0.034973
	Results in [21]	4.14e-03	2.07e-03	1.03e-03	6.5e-04
	Results using our method	4.100e-03	1.000e-03	2.2796e-04	4.8757e-05
$\epsilon = 10^{-8}$	Results in [1]	0.269957	0.137749	0.069587	0.034974
	Results in [21]	4.14e-03	2.07e-03	1.03e-03	6.5e-04
	Results using our method	4.100e-03	1.000e-03	2.5422e-04	5.8483e-05

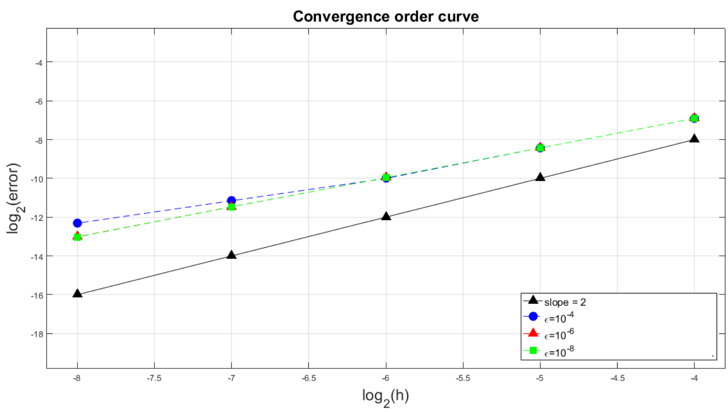

FIGURE 1. Convergence order curve curve for **Example 1.** with $\epsilon = 10^{-6}, 10^{-8}$ respectively.

Example 4.2. Consider the following SPVIDE in [20]:

$$\epsilon y'(x) + y(x) + \int_0^x k(x,t)y(t)dt = f(x), x \in [0,1],$$

Where $k(x,t) = \exp(t)$, $f(x) = (\epsilon+1)\exp(x-1) - \epsilon\exp(\frac{-(1+\epsilon)x}{\epsilon}) - \epsilon\exp(\frac{-x}{\epsilon}) + \frac{\exp(2x-1)}{2} + \epsilon - \frac{1}{2e}$, and $y(0) = 1+\exp(-1)$ is the initial condition. The exact solution to this problem is given by:

$$y(x) = \exp(x-1) + \exp(\frac{-(1+\epsilon)x}{\epsilon}).$$

Values of $E(h)$, $C.R.$ and $e(h)$ for this problem are presented in **Table 4.** and **Table 5.** Here we compare our method with the method in [20] in **Table 6.** Also, **Fig.2** demonstrate the convergence order curve of the numerical solution.

TABLE 4. Maximum errors and rates of convergence for **Example 2.**

	n = 16	n = 32	n = 64	n = 128	n = 256
$\epsilon = 10^{-4}$	2.4100e-02	6.8300e-03	1.8900e-03	4.8769e-04	1.3346e-04
C.R.		1.82	1.85	1.95	1.87
$\epsilon = 10^{-5}$	2.2990e-02	6.300e-03	1.7096e-03	4.4959e-04	1.1584e-04
C.R.		1.86	1.89	1.92	1.96
$\epsilon = 10^{-6}$	2.3000e-02	5.7914e-03	1.5270e-03	3.9111e-04	9.7050e-05
C.R.		1.98	1.92	1.96	2.01
$\epsilon = 10^{-7}$	2.3000e-02	5.7914e-03	1.5000e-03	3.8647e-04	9.6617e-05
C.R.		1.98	1.95	1.96	2.00
$\epsilon = 10^{-8}$	2.3000e-02	5.7914e-03	1.3102e-03	3.3268e-04	8.3170e-05
C.R.		1.98	2.15	1.96	2.00

TABLE 5. Numerical results for **Example 2.** for $n = 256$, $\epsilon = 10^{-8}$

x	Exact Solution	Approximating Solution	Obtained absolute error
0.125	0.4168620	0.4167849	0.000077095
0.25	0.4723666	0.4722850	0.000081563
0.375	0.5352614	0.5351911	0.000070348
0.5	0.6065307	0.6064539	0.000076810
0.625	0.6872893	0.6872209	0.000068438
0.75	0.7788008	0.7787350	0.000065765
0.875	0.8824969	0.8824318	0.000065139

TABLE 6. Comparison of the maximum errors obtained by our methods and those in [20] for **Example 2.**

		n = 32	n = 64	n = 128	n = 256
$\epsilon = 10^{-6}$	[20] Shishkin mesh	1.32e-02	4.71e-03	1.59e-03	5.19e-04
	[20] Shishkin mesh	1.05e-01	2.48e-02	5.78e-03	1.43e-03
	Results using our method	5.7914e-03	1.5270e-03	3.9111e-04	9.7050e-05
$\epsilon = 10^{-8}$	[20] Improved graded mesh	3.73e-03	1.11e-03	3.21e-04	9.17e-05
	[20] Improved graded mesh	1.32e-02	3.22e-03	8.13e-04	2.02e-04
	Results using our method	5.7914e-03	1.3102e-03	3.3268e-04	8.3170e-05

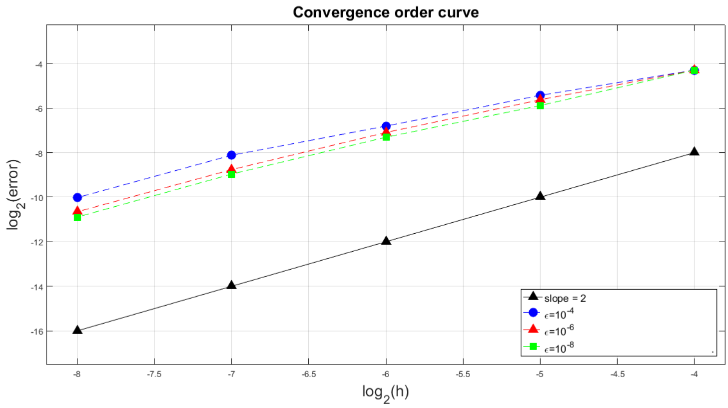

FIGURE 2. Convergence order curve for **Example 2.** with $\epsilon = 10^{-6}, 10^{-8}$ respectively.

5. Conclusion

This paper proposes a smooth method for approximating the solutions of SPVIDEs. SPVIDEs discretized using the non-polynomial spline function. The second-order convergence of the proposed method has been derived and the computational outcomes are conformable with theoretical expectations. We showed that the proposed scheme is second-order uniformly convergent with respect to the perturbation and the mesh parameters. Tables and figures show the feasibility and applicability of our method. The proposed method is more accurate than the methods presented in [1], [21] and [20].

Acknowledgements

We would like to thank the referees for their comments and suggestions on the manuscript.

References

[1] G.M. Amiraliyev, S. Şevgin, Uniform difference method for singularly perturbed Volterra integro-differential equations, Appl. Math. Comput. 179 (2) (2006) 731–741.

[2] J.S. Angell, W.E. Olmstead, Singularly perturbed Volterra integral equations II, Siam J. Appl. Math. 47 (6) (1987) 1150–1162.

[3] A.M. Bijura, Asymptotics of integro-differential models with integrable kernels, Int. J. Math. Math. Sci. 25 (2003) 1577–1598.

[4] G.S. Jordan, A nonlinear singularly perturbed Volterra integro-differential equation of non-convolution type, Proceedings of the Royal Society of Edinburgh: Section A Mathematics 80 (3-4) (1987) 235–247.

[5] G.S. Jordan, Some nonlinear singularly perturbed Volterra integro-differential equations, Volterra Equations, In: Londen SO., Staffans O.J. (eds) Volterra Equations, Lecture Notes in Mathematics, vol. 737, Springer, Berlin, Heidelberg.

[6] A.S. Lodge, J.B. McLeod, J.A. Nohel, A nonlinear singularly perturbed Volterra integro-differential equation occurring in polymer rheology, P ROY SOC EDINB A, 80 (1-2) (1978) 99–137.

[7] J.P. Kauthen, A survey of singularly perturbed Volterra equations, Appl. Numer. Math. 24 (2) (1997) 95–114.

[8] J.I. Ramos, Exponential techniques and implicit Runge-Kutta methods fo rsingularly-perturbed Volterra integro-differential equations, Neural Parallel Sci. Comput. 16 (2008) 387–404.

[9] M.K. Kadalbajoo, K.C. Patidar, ϵ-Uniformly convergent fitted mesh infinite difference methods for general singular perturbation problems, Appl. Math. Comput. 179 (1) (2006) 248–266.

[10] M. Kudu, I. Amirali, G.M. Amiraliyev, A finite-difference method for a singularly perturbed delay integro-differential equation, J. Compu.t Appl. Math. 308 (2016) 379–390.

[11] T. Lin, Y. Lin, M. Rao, S. Zhang, Petrovf Galerkin methods for linear Volterra integro-differential equations, SIAM J. Numer. Anal. 38 (2000) 937–963.

[12] S.E. El-Gendi, M.A.K. Ibrahim, A.B. Shamardan, The numerical treatment of a singular integro-differential equation arising from the diffusion of discrete particles in a turbulent fluid, J. Nat. Sci. Math. 28 (1988) 95–106.

[13] J.P. Kauthen, Implicit Runge-Kutta methods for some integro-differential-algebraic equations, Appl. Numer. Math. 13 (1-3) (1993) 125–134.

[14] J.P. Kauthen, Implicit Runge-Kutta methods for singularly perturbed integro-differential systems, Appl. Numer. Math. 18 (1) (1995) 201–210.

[15] A.A. Salama, D.J. Evans, Fourth order scheme of exponential type for singularly perturbed volterra integro-differential equations, Inter. J. Comput. Math. 77 (1) (2001) 153–164.

[16] V. Horvat, M. Rogina,Tension spline collocation methods for singularly perturbed Volterra integro-differential and Volterra integral equations, J. Comput. Appl. Math. 140 (2002) 381–402.

[17] A.M. Bijura, Initial-layer theory and model equations of Volterra type, Ima. J. Appl. Math. 71 (3) (2006) 315–331.

[18] S. Şevgin, Numerical solution of a singularly perturbed Volterra integro-differential equation, Adv. Differ. Equ. 2014 (1) (2014) 1–15.

[19] C. Zhongdi, X. Lifeng, A parameter robust numerical method for a singularly perturbed Volterra equation in security technologies, Matrix 1 (2006) 20–22.

[20] X. Tao, Y. Zhang, The coupled method for singularly perturbed Volterra integro-differential equations, Adv. Differ. Equ. 2019 (1) (2019).

[21] B. Iragi, J.B. Munyakazi, A uniformly convergent numerical method for a singularly perturbed Volterra integro-differential equation, Int. J. Comput. Math. 97 (4) (2019) 759–771.

[22] I.H. Sloan, Nonpolynomial interpolation, J. Approx. Theory, 39 (1983) 97–117.

[23] J. Stoer, R. Bulirsch , Introduction to Numerical Analysis, Second Edition, Springer Verlag, 1992.

[24] P. Singh, S.D. Joshi, R.K. Patney, K. Saha, Some studies on non-polynomial interpolation and error analysis, Appl. Math. Comput. 244 (2014) 809–821.

[25] A. Khan, P. Khandelwal, Numerical solution of third order singularly perturbed boundary value problems using exponential quartic spline, Thai J. Math. 17 (3) (2019) 663–672.

[26] M. Zarebnia, Z. Sarvari, New approach for numerical solution of the one-dimensional Bratu equation, Thai J. Math. 11 (3) (2013) 611–621.

[27] Q. Ding, P.J.Y. Wong, Mid-knot cubic non-polynomial spline for a system of second-order boundary value problems, Bound. Value Probl. 2018 (2018) Article no. 156.

[28] Q. Ding, P.J.Y. Wong, Quintic non-polynomial spline for time-fractional nonlinear Schr¨dinger equation, Adv. Differ. Equ. 2020 (2020) Article no. 577.

[29] R.K. Mohanty, S. Nayak, A. Khan, Non-polynomial cubic spline discretization for system of non-linear singular boundary value problems using variable mesh, Adv. Differ. Equ. 2017 (2017) Article no. 327.

[30] R. Jalilian, T. Tahernezhad, Exponential spline method for approximation solution of Fredholm integro–differential equation, Int. J. Comput. Math. 97 (3) (2019) 1–17.

[31] J. Rashidinia, R. Jalilian, V. Kazemi, Spline methods for the solutions of hyperbolic equations, Appl. Math. Comput. 190 (2007) 882–886.

[32] J. Rashidinia, R. Mohammadi, R. Jalilian, Spline methods for the solution of hyperbolic equation with variable coefficients, Numer. Methods Partial Differ. Equ. 23 (2007) 1411–1419.

[33] J. Rashidinia, R. Mohammadi, R. Jalilian, Cubic spline method for two-point boundary value problems, Int. J. Eng. Sci. 19 (2008) 39–43.

[34] T. Sultana, A. Khan, P. Khandelwal, A new non-polynomial spline method for solution of linear and non-linear third order dispersive equations, Adv. Differ. Equ. 2018 (2018) Article no. 316.

[35] T. Tahernezhad, R. Jalilian, Exponential spline for the numerical solutions of linear Fredholm integro-differential equations, Adv. Differ. Equ. 2020 (2020) Article no. 141.