



## An efficient Sinc polynomial collocation approach for solving m-dimensional stochastic Volterra integral equations

Faezeh Bahmani and Ali Eftekhari\*

Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Kashan, Kashan 87317-53153, Iran.

### Abstract

This paper introduces a polynomial sinc-based collocation method, combined with Gauss-Legendre and Newton-Cotes quadrature rules to solve stochastic Volterra integral equations (SVIEs) with a m-dimensional Brownian motion process. The proposed technique employs Lagrange polynomial interpolation at sinc-type collocation nodes to approximate the solution, thereby reducing the SVIE to a system of algebraic equations that can be solved at low to moderate computational cost. A rigorous convergence analysis of the scheme is presented, and several numerical experiments are carried out to illustrate its accuracy, efficiency, and reliability.

**Keywords.** Stochastic Volterra integral equations, Poly-sinc collocation method, Itô integral, m-dimensional Brownian motion process, Gauss-Legendre quadrature, Composite Newton-Cotes quadrature, Error analysis.

**2010 Mathematics Subject Classification.** 65C30, 60H35, 60H20, 41A55.

### 1. INTRODUCTION

Over the past few decades, advancements in computational tools have led to a shift from deterministic to stochastic models in the modeling of numerous phenomena. These techniques provide more realistic predictions by incorporating the inherent randomness in systems leading to reduced errors and a deeper understanding of system behavior. Integral equation theory traces its origins to concepts introduced by Fourier, while Volterra made significant contributions to their advancement and practical applications. The year 1895 marked a significant milestone in this field. In the past 15–20 years, there has been renewed interest in Volterra integral equations, driven by their expanding application domains and an improved understanding of their properties. Stochastic Volterra Integral Equations (SVIEs) represent a significant class of stochastic equations that have extensive applications across various scientific disciplines, including biology, physics, finance, and engineering. In practical applications, widely studied SVIEs include the Black-Scholes model, the Ornstein-Uhlenbeck Volterra model, and the Heston-Volterra model. This research focuses on the stochastic Volterra integral equation with a m-dimensional Brownian motion process, represented as follows:

$$y(t) = y_0(t) + \int_0^t \mathcal{R}_0(t, s, y(s))ds + \sum_{j=1}^m \int_0^t \mathcal{R}_j(t, s, y(s))dW_j(s), \quad 0 \leq t \leq 1, \quad (1.1)$$

Here,  $y(t)$  is the unknown stochastic process to be determined, while  $y_0(t)$  is a known function. The kernels  $\mathcal{R}_0(t, s, y(s))$  and  $\mathcal{R}_j(t, s, y(s))$ ,  $j = 1, \dots, m$ , are known stochastic process defined on the probability space  $(\Omega, \mathcal{F}, \mathbb{P})$ .

Additionally,  $W(t) = (W_1(t), W_2(t), \dots, W_m(t))$  represents a m-dimensional Brownian motion process, and

$$\int_0^t \mathcal{R}_j(t, s, y(s))dW_j(s), \quad j = 1, \dots, m,$$

are Itô integrals.

Received: 11 September 2025; Accepted: 07 April 2026.

\* Corresponding author. Email: [eftekhari@kashanu.ac.ir](mailto:eftekhari@kashanu.ac.ir).

Due to the involvement of stochastic processes, obtaining analytical solutions for many stochastic equations is often difficult or impossible. This presents fundamental challenges in applying and analyzing these equations across various scientific fields. To address these challenges, researchers have developed various numerical techniques to approximate solutions to SVIEs. Researchers have employed a wide range of polynomials and basis functions to tackle SVIEs, including Genocchi polynomials [29], second-kind Chebyshev wavelets with parallel computing process [1], modified hat functions [22], block pulse functions [16], Euler polynomials [17], shifted Jacobi polynomials [24], Lagrange polynomials [6], and Haar wavelets [21]. A key advantage of these methods is their ability to reduce the problem into a system of algebraic equations. Other notable approaches include moving least squares (MLS) combining with the spectral collocation method [18], Least squares method based on Haar wavelet [10], enhanced moving least squares [30], Least squares method based on block pulse function [11], least squares method based on generalized hat function [42], triangular functions with operational matrix of integration [14], Picard iteration with hat basis functions [26], numerical iterative method [27], Floater-Hormann interpolation method [19], and quintic B-spline collocation method [20]. Among approximation techniques, sinc methods are distinguished for their accuracy and reliability. Pioneered by Frank [31, 32], and further developed in [33], sinc methods have found wide application in mathematics, engineering, and many applied sciences [2, 5, 15, 25, 37].

This paper employs a poly-sinc approximation to solve the aforementioned SVIE. The polynomial-sinc collocation method was first introduced by Frank Stenger in [34] and has since been significantly advanced by subsequent research, notably by Stenger et al. [36] and Youssef et al. [39, 40]. The approach effectively handles a wide range of problems, including ordinary differential equations with endpoint singularities and fractional or stochastic differential and integral equations [4, 9, 12, 13, 41]. This method employs a collocation technique based on Lagrange interpolation at non-equidistant sinc points which are generated through conformal mapping. Compared to equidistant interpolation nodes, Chebyshev points, or even improved Chebyshev points, sinc points significantly enhance the approximation quality. Moreover, selecting these points as interpolation nodes preserves the distinctive feature of classical sinc approximation: an exponential convergence rate [34]. The estimate of the Lebesgue constant for this approximation exhibits a logarithmic asymptotic pattern [40]. This approach reduces the problem to solving a system of linear or nonlinear algebraic equations, enabling direct computations. Stenger introduced poly-sinc approximation to improve the accuracy associated with classical sinc differentiation [34], which has led to extensive discussions in the field between Stenger and other researchers [36]. Another advantage of this approximation is the ease of performing algebraic computations on sinc polynomials. By employing a minimal number of basis functions and sinc points as collocation nodes, a desirable solution approximation is achieved. Our paper is organized as follows: The preliminaries of stochastic calculus and poly-sinc interpolation are presented in section 2. Section 3 outlines a spectral collocation method based on poly-sinc approximation for solving stochastic Volterra integral equation with a  $m$ -dimensional Brownian motion process in the form of (1.1). The analysis of the convergence and error of the proposed method is performed in section 4. Section 5 contains numerical experiments that demonstrate the effectiveness of our approach. Finally, we conclude our paper with a brief summary in the last section.

## 2. MATHEMATICAL BACKGROUND

This section introduces the core mathematical frameworks essential for the subsequent analysis, encompassing stochastic calculus, poly-sinc interpolation, and Gauss-Legendre quadrature.

### 2.1. Stochastic calculus.

**Definition 2.1.** Let  $\Omega$  be a countable sample space. A probability space is defined as the triple  $(\Omega, \mathcal{F}, \mathbb{P})$ , where  $\mathcal{F}$  is a  $\sigma$ -field of subsets of  $\Omega$ , and  $\mathbb{P} : \mathcal{F} \rightarrow [0, 1]$  is a probability function such that:

- (i)  $0 \leq \mathbb{P}(A) \leq 1$  for all  $A \in \mathcal{F}$ ;
- (ii)  $\mathbb{P}(\Omega) = 1$ ;
- (iii) For any sequence of mutually disjoint events  $A_1, A_2, \dots \in \mathcal{F}$ ,

$$\mathbb{P}(\cup_{n=1}^{\infty} A_n) = \sum_{n=1}^{\infty} \mathbb{P}(A_n).$$



The concept of Brownian motion, originating from Robert Brown’s 1827 observations of pollen grains exhibiting random motion in water, is key to continuous-time stochastic processes.

**Definition 2.2.** A standard Brownian motion, denoted by  $W(t)$ ,  $t \geq 0$ , is a continuous-time stochastic process starting at the origin ( $W(0) = 0$ , with probability 1), that satisfies the following properties:

- (i) The increments of  $W(t)$  are independent;
- (ii)  $W(t)$  has continuous paths;
- (iii) The increments  $W(t) - W(s)$  are normally distributed with mean 0 and variance  $|t - s|$ , or equivalently,  $W(t) - W(s) \sim \mathcal{N}(0, |t - s|)$ .

**Definition 2.3.** Consider  $\mathcal{W}(S, R)$  as the set of functions  $f : [0, \infty) \times \Omega \rightarrow \mathbb{R}$ , satisfying the following three conditions

- (i) The mapping  $(t, \omega) \rightarrow f(t, \omega)$  is  $\mathcal{BF}$  measurable, where  $\mathcal{B}$  denotes the Borel  $\sigma$ -algebra on  $[0, \infty)$ ;
- (ii) The function  $f(t, \omega)$  is  $\mathcal{F}_t$ -adapted;
- (iii)  $\mathbb{E} \left[ \int_S^R f^2(t, \omega) dt \right] < \infty$ .

**Definition 2.4.** Given a function  $f \in \mathcal{W}(S, R)$ , we define the stochastic Itô integral as

$$\mathcal{I}[f](\omega) = \int_S^R f(t, \omega) dW_t(\omega), \tag{2.1}$$

where  $W_t$  represents the standard Brownian motion.

**Theorem 2.5.** [23] (Integration by parts) Suppose  $y(s, \omega) = y(s)$  is a continuous function of bounded variation in  $s \in [0, t]$  for almost every  $\omega$ . Then the following property is valid

$$\int_0^t y(s) dW(s) = y(t)W(t) - \int_0^t W(s) dy(s). \tag{2.2}$$

**2.2. Poly-sinc interpolation.** Suppose that  $u : [a, b] \rightarrow \mathbb{R}$  is a continuous function. We apply the poly-sinc interpolation method for approximating  $u$  at the sinc data  $\{t_k, u(t_k)\}_{k=0}^n$ . This approximation, which belongs to the family of polynomial-like approximations, exhibits desirable accuracy and provides reliable results provided that the function  $u$  with  $u_k = u(t_k)$  belongs to suitable spaces of analytic functions. Define the conformal map

$$w = \phi(z) = \ln \left( \frac{z - a}{b - z} \right),$$

which maps the simply connected eye-shaped region

$$\mathcal{D}_E = \left\{ z \in \mathbb{C} : \left| \arg \left( \frac{z - a}{b - z} \right) \right| < d \right\}, \tag{2.3}$$

to the strip region

$$\mathcal{D}_d = \{w \in \mathbb{C} : |\Im(w)| < d\}, \tag{2.4}$$

with width  $2d$ , where  $d \in (0, \frac{\pi}{2})$ . The map  $\phi$  satisfies  $a = \phi^{-1}(-\infty)$  and  $b = \phi^{-1}(\infty)$ . Furthermore, the separated sinc points  $\{t_k\}_{k=-N}^N$  are given by

$$t_k = \phi^{-1}(kh) = \frac{a + be^{kh}}{1 + e^{kh}}, \quad k = -N, \dots, N, \tag{2.5}$$

where  $h$  is a positive step size. Let  $\mathcal{P}_N$  denote the space of polynomials of degree at most  $n = 2N$ . We now concentrate on the construction of Lagrange interpolating polynomials of degree  $\leq n$  utilizing sinc points. These polynomials are constructed based on the interpolation point set  $(t_k, u(t_k))$ , where the sinc points  $\{t_k\}_{k=-N}^N$  are obtained from (2.5) and the corresponding function values  $\{u_k\}_{k=-N}^N$  are evaluated at these sinc points. The one-dimensional interpolation approximation for  $u$  on the interval  $[a, b]$  constructed with  $n + 1$  sinc points is formulated as

$$U_N(t) = \sum_{k=-N}^N u(t_k) \mathfrak{L}_k(t), \tag{2.6}$$



where  $\{\mathfrak{L}_k\}_{k=-N}^N$  are the Lagrange basis polynomials evaluated at sinc points, expressed as

$$\mathfrak{L}_k(t) = \prod_{\substack{j=-N \\ j \neq k}}^N \frac{t - t_j}{t_k - t_j}, \quad k = -N, \dots, N. \quad (2.7)$$

Following the approach presented in [34], we utilize the interpolation  $U_N$  as follows:

$$U_N(t) = \sum_{k=1}^{2N+1} u(\check{t}_k) \check{\mathfrak{L}}_k(t), \quad (2.8)$$

within this notation,  $\check{t}_k$  is equivalent to  $t_{k-N-1}$ , and  $\check{\mathfrak{L}}_k(t)$  represents the expansion of  $\mathfrak{L}_{k-N-1}(t)$  in terms of powers of  $t$ ,

$$\check{\mathfrak{L}}_k(t) = \sum_{i=0}^{2N} a_{i,k} t^i, \quad k = 1, \dots, 2N+1. \quad (2.9)$$

A brief discussion of the function space for  $u$  is presented here. Let  $\rho = e^\phi$ , and for fixed positive numbers  $\alpha, \beta$ , let  $L_{\alpha,\beta}(\mathcal{D}_E)$  denote the family of functions analytic in  $\mathcal{D}_E = \phi^{-1}(\mathcal{D}_d)$  satisfying

$$|u(z)| \leq c \frac{|\rho(z)|^\alpha}{(1 + |\rho(z)|)^{\alpha+\beta}}, \quad \forall z \in \mathcal{D}_E. \quad (2.10)$$

For  $0 \leq \alpha, \beta \leq 1$ , the set  $M_{\alpha,\beta}(\mathcal{D}_E)$  consists of functions  $g$  defined on  $\mathcal{D}_E$  with finite limits  $g(a) = \lim_{z \rightarrow a} g(z)$  and  $g(b) = \lim_{z \rightarrow b} g(z)$  within  $\mathcal{D}_E$ , such that

$$u = g - \frac{g(a) + \rho g(b)}{1 + \rho}, \quad (2.11)$$

and  $u \in L_{\alpha,\beta}(\mathcal{D}_E)$ . This ensures  $u$  vanishes at the endpoints of  $[a, b]$ . Assume  $u$  is analytic and uniformly bounded by  $B(u)$  in the extended region

$$\mathcal{D} = \mathcal{D}_E \cup_{t \in [a,b]} \mathcal{B}_r(t), \quad (2.12)$$

where  $\mathcal{B}_r(t) = \{z \in \mathbb{C} : |z - t| < r\}$  and  $r > 0$ .

For the domain  $[0, 1]$ , the error of approximating  $u(t)$  by  $U_N(t)$  is estimated by setting  $[a, b] = [0, 1]$ .

**Theorem 2.6.** [34] Let  $\mathcal{D}_E$  be the inverse image of  $\mathcal{D}_d$  under the conformal map  $\phi(t) = \ln\left(\frac{t}{1-t}\right)$ , with  $\mathcal{D}_d$  as in (2.4). Define  $\mathcal{D}_1 = \mathcal{D}_E \cup_{t \in [0,1]} \mathcal{B}_\iota(t)$ , with  $\iota \geq \frac{1}{2}$ . Assume  $y$  is an analytic function and uniformly bounded on  $\mathcal{D}_1$ . Set  $h = \gamma/\sqrt{N}$ , where  $\gamma > 0$  is independent of  $N$ . The error bound for  $|y(t) - y_N(t)|$ , with  $y_N$  as in (2.6), can be expressed as

$$|y(t) - y_N(t)| \leq \mathcal{S} \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2 \sqrt{N}}{2\gamma}\right), \quad (2.13)$$

where  $\mathcal{S}$  is a constant, independent of  $N$ .

In poly-sinc interpolation, sinc nodes are employed as the interpolation points. This choice is motivated by the well-known exponential accuracy of sinc-based interpolation for functions analytic on suitable domains. Specifically, under the hypotheses of Theorem 2.6, the associated interpolant achieves exponential convergence with respect to the number of nodes, making sinc nodes a highly efficient choice for numerical approximation. The use of such non-uniform nodal sets is a cornerstone of high-order spectral methods, and their analysis is well-established in numerical analysis; see, for example, the discussions on spectral convergence by Trefethen [38], Boyd [7], and Stenger [35].



### 3. COMPUTATIONAL FRAMEWORK

This section describes the application of poly-sinc interpolation to approximate the solution  $y$  of Eq. (1.1), expressed as

$$y(t) \approx y_N(t) = \sum_{i=1}^{2N+1} y_i \check{\mathfrak{S}}_i(t), \tag{3.1}$$

where  $N$  is a positive integer and  $\{y_i := y_N(\check{t}_i)\}_{i=1}^{2N+1}$  are unknown coefficients. Substituting (3.1) into (1.1) yields

$$y_N(t) = y_0(t) + \int_0^t \mathcal{R}_0(t, s, y_N(s))ds + \sum_{j=1}^m \int_0^t \mathcal{R}_j(t, s, y_N(s))dW_j(s). \tag{3.2}$$

Utilizing Theorem (2.5), this becomes

$$y_N(t) = y_0(t) + \sum_{j=1}^m \mathcal{R}_j(t, t, y_N(t))W_j(t) + \int_0^t \mathcal{R}_0(t, s, y_N(s))ds - \sum_{j=1}^m \int_0^t \Lambda_j(t, s, y_N(s))W_j(s)ds, \tag{3.3}$$

where  $\Lambda_j(t, s, y_N(s)) = \frac{\partial}{\partial s}(\mathcal{R}_j(t, s, y_N(s)))$ ,  $j = 1, \dots, m$ .

We approximate the integrals in (3.3) using Gauss-Legendre and Newton-Cotes quadratures. Focusing on the first integral in (3.3), we can utilize the Gauss-Legendre quadrature rule to obtain a suitable approximation. By introducing the change of variable  $s = t\tau$ , we can obtain

$$\begin{aligned} \int_0^t \mathcal{R}_0(t, s, y_N(s))ds &= t \int_0^1 \mathcal{R}_0(t, t\tau, y_N(t\tau))d\tau \\ &= t \sum_{k=1}^{M_1} \mathcal{R}_0(t, t\zeta_k, y_N(t\zeta_k))\alpha_k + Ef(G-L), \end{aligned} \tag{3.4}$$

where the quadrature nodes  $\{\zeta_k\}_{k=1}^{M_1}$  are the zeros of the shifted Legendre polynomial of degree  $M_1$  on  $[0, 1]$ , and  $\{\alpha_k\}_{k=1}^{M_1}$  denote the corresponding Gauss-Legendre weights. Also,  $Ef(G-L)$  representing the associated quadrature error. For further details on the construction of the Gauss-Legendre quadrature, see [28]. For numerical evaluation of Itô integrals in (3.3), we utilize a composite Newton-Cotes quadrature rule. This method has proven effective for integrals with non-smooth integrands [8]. To implement this approach, we utilize an  $M_2$ -point quadrature rule with  $K$  panels for a fixed set of positive integers  $M_2$  and  $K$ . To achieve this objective, we implement a variable transformation  $\eta = \frac{s}{t}$ . Then we have

$$\begin{aligned} \sum_{j=1}^m \int_0^t \Lambda_j(t, s, y_N(s))W_j(s)ds &= t \sum_{j=1}^m \int_0^1 \Lambda_j(t, t\eta, y_N(t\eta))W_j(t\eta)d\eta \\ &= t \sum_{j=1}^m \left( \sum_{r=1}^K \sum_{i=1}^{M_2} \Lambda_j(t, t\eta_i^{(r)}, y_N(t\eta_i^{(r)}))W_j(t\eta_i^{(r)})w_i^{(r)} + E_{M_2, K}^{(j)} \right), \end{aligned} \tag{3.5}$$

where  $\{\eta_i^{(r)}\}_{i=1}^{M_2}$  and  $\{w_i^{(r)}\}_{i=1}^{M_2}$ ,  $r = 1, \dots, K$ , are the quadrature nodes and weights of the closed Newton-Cotes rule, respectively and  $E_{M_2, K}^{(j)}$  represents the quadrature error corresponding to the  $j$ -th integral.



By substituting (3.4) and (3.5) into (3.3), we discretize the equation and arrive at

$$\begin{aligned}
y_N(t) &= y_0(t) + \sum_{j=1}^m \mathcal{R}_j(t, t, y_N(t)) W_j(t) + t \sum_{k=1}^{M_1} \mathcal{R}_0(t, t\zeta_k, y_N(t\zeta_k)) \alpha_k \\
&\quad - t \sum_{j=1}^m \sum_{r=1}^K \sum_{i=1}^{M_2} \Lambda_j(t, t\eta_i^{(r)}, y_N(t\eta_i^{(r)})) W_j(t\eta_i^{(r)}) w_i^{(r)} + t \text{E}f(\text{G-L}) - t \sum_{j=1}^m E_{M_2, K}^{(j)},
\end{aligned} \tag{3.6}$$

By ignoring the error term and applying the collocation condition at the sinc points  $\{\check{t}_r\}_{r=1}^{2N+1}$ , the discretization of Eq. (3.6) takes the following form

$$\begin{aligned}
y_N(\check{t}_r) &= y_0(\check{t}_r) + \sum_{j=1}^m \mathcal{R}_j(\check{t}_r, \check{t}_r, y_N(\check{t}_r)) W_j(\check{t}_r) \\
&\quad + \check{t}_r \sum_{k=1}^{M_1} \mathcal{R}_0(\check{t}_r, \check{t}_r \zeta_k, y_N(\check{t}_r \zeta_k)) \alpha_k \\
&\quad - \check{t}_r \sum_{j=1}^m \sum_{r=1}^K \sum_{i=1}^{M_2} \Lambda_j(\check{t}_r, \check{t}_r \eta_i^{(r)}, y_N(\check{t}_r \eta_i^{(r)})) W_j(\check{t}_r \eta_i^{(r)}) w_i^{(r)}, \quad r = 1, \dots, 2N+1.
\end{aligned} \tag{3.7}$$

Define the vector  $\mathbf{y} = (y_1, \dots, y_{2N+1})^T$ , where  $y_r = y_N(\check{t}_r)$ , and the residual functions

$$\begin{aligned}
Q_r(\mathbf{y}) &= y_N(\check{t}_r) - y_0(\check{t}_r) - \sum_{j=1}^m \mathcal{R}_j(\check{t}_r, \check{t}_r, y_N(\check{t}_r)) W_j(\check{t}_r) \\
&\quad - \check{t}_r \sum_{k=1}^{M_1} \mathcal{R}_0(\check{t}_r, \check{t}_r \zeta_k, y_N(\check{t}_r \zeta_k)) \alpha_k \\
&\quad + \check{t}_r \sum_{j=1}^m \sum_{r=1}^K \sum_{i=1}^{M_2} \Lambda_j(\check{t}_r, \check{t}_r \eta_i^{(r)}, y_N(\check{t}_r \eta_i^{(r)})) W_j(\check{t}_r \eta_i^{(r)}) w_i^{(r)},
\end{aligned}$$

for  $r = 1, \dots, 2N+1$ . Let  $\mathbf{Q}(\mathbf{y}) = (Q_1(\mathbf{y}), \dots, Q_{2N+1}(\mathbf{y}))^T$ . Thus, the system of Eqs. (3.7) takes the following form

$$\mathbf{Q}(\mathbf{y}) = \mathbf{0}. \tag{3.8}$$

This nonlinear system can be solved iteratively, e.g., using Newton's method, to obtain the coefficients  $y_r$ , yielding the numerical approximation  $y_N(t)$ . Note that the unknown coefficients  $y_r$  of the interpolating polynomial correspond to the values of the approximate solution at the sinc nodes, which are known to be close to zero. Therefore, initializing Newton's method with the zero vector is a well-founded choice. Moreover, since Newton's method exhibits quadratic convergence in a neighborhood of the solution, this initial guess ensures a rapid convergence. In the following, we present the algorithm of the proposed scheme.



---

**Algorithm 1** Poly-sinc collocation method.

---

- 1: **Input:**
  - 2: Number of sample paths:  $1 \leq q \leq 200$
  - 3: Stochastic processes of SVIE (1.1):  $\mathcal{R}_0, \mathcal{R}_j, j = 0, \dots, m$
  - 4: Brownian motion processes:  $W_j(t), j = 1, \dots, m$
  - 5: Positive integer for generating sinc points:  $N$
  - 6: Positive parameter associated with mesh size  $h = \frac{\gamma}{\sqrt{N}}$ :  $\gamma$
  - 7: Number of nodal points for Legendre-Gauss quadrature:  $M_1$
  - 8: Number of panels for computing integrals (3.5) of Itô type:  $K$
  - 9: Number of nodal points for closed Newton-Cotes formula:  $M_2$
  - 10: Stopping criterion for Newton's method:  $\epsilon = 10^{-16}$
  - 11: Maximum iterations for Newton's method: Max-Iter = 6
  - 12: Initial guess:  $\mathbf{y}^{(0)} = \mathbf{0}$
  - 13: **Output:** Approximate solution after  $q$  simulations:  $y(t) \approx y_N(t)$
  - 14: **procedure** POLY-SINC COLLOCATION
  - 15:     Calculate  $t_k, k = 1, \dots, 2N + 1$ , as sinc points.
  - 16:     Calculate Lagrange basis polynomials  $\{\tilde{\mathcal{L}}_k\}_{k=1}^{2N+1}$  defined in (2.8).
  - 17:     Calculate  $\zeta_k, k = 0, \dots, M_1$ , as zeros of Legendre polynomial of degree  $M_1 + 1$ .
  - 18:     Calculate weights  $\alpha_k, k = 0, \dots, M_1$ , for  $M_1$ -point Legendre-Gauss quadrature rule.
  - 19:     Calculate quadrature nodes  $\eta_i^{(r)}, i = 1, \dots, M_2, r = 1, \dots, K$ , and weights  $w_i^{(r)}, i = 1, \dots, M_2, r = 1, \dots, K$ , of the closed Newton-Cotes formula defined in section 3.
  - 20:     Generating the system  $Q_i(y) = 0, i = 1, \dots, 2N + 1$  defined in section 3.
  - 21:     Applying Newton's iterative method to solve (3.8)
  - 22: **end procedure**
- 

4. ERROR ESTIMATES AND CONVERGENCE ANALYSIS

This section presents a rigorous error analysis for our numerical method. We establish an upper bound for the error by employing the infinity norm, defined as [3]

$$\|g(t)\|_\infty = \max_{0 \leq t \leq 1} |g(t)|. \tag{4.1}$$

**Theorem 4.1.** *Under the assumptions of Theorem 2.6, Let  $y(t)$  be the exact solution of Eq. (1.1) and  $y_N(t)$  its poly-sinc approximation. Suppose that for each  $t \in [0, 1]$ , the function  $\mathcal{R}_0(t, s, y(s)) \in C^{2M_1}[0, 1]$ , and that the functions  $\mathcal{R}_j, j = 1, \dots, m, \mathcal{R}_0$ , and  $\Lambda_j, j = 1, \dots, m$ , satisfy Lipschitz conditions with constants  $L_{\mathcal{R}_j}, j = 1, \dots, m, L_{\mathcal{R}_0}$ , and  $L_{\Lambda_j}, j = 1, \dots, m$ , respectively. Additionally, assume that for each  $j = 1, \dots, m$ , the infinity norm of  $W_j$  is bounded above by  $R_j$ . Then, there exists a constant  $\mathcal{K}$  independent of  $N$  such that*

$$\|e_N(y(t))\|_\infty \leq \frac{2\mathcal{M}}{(2M_1 - 1)!} + E_{M_2, K} + 3\mathcal{K} \left( \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2 \sqrt{N}}{2\gamma}\right) \right),$$

where

$$\mathcal{M} = \sup_{0 \leq t \leq 1} \left| \frac{\partial^{2M_1}}{\partial \tau^{2M_1}} \mathcal{R}_0(t, t\tau, y(t\tau)) \right|.$$

$\mathcal{M}$  is obtained from the convergence analysis of the Gauss-Legendre quadrature [4]. Moreover,

$$E_{M_2, K} = \sum_{j=1}^m \left| E_{M_2, K}^{(j)} \right|,$$



and the error term is defined as

$$e_N(y(t)) = \Phi_1(y(t)) - \Phi_2(y_N(t)).$$

Here, the operators  $\Phi_1$  and  $\Phi_2$  are given by

$$\Phi_1(y(t)) = y(t) - y_0(t) - \sum_{j=1}^m \mathcal{R}_j(t, t, y(t))W_j(t) - t \int_0^1 \mathcal{R}_0(t, t\tau, y_N(t\tau))d\tau + t \sum_{j=1}^m \int_0^1 \Lambda_j(t, t\eta, y(t\eta))W_j(t\eta)d\eta, \quad (4.2)$$

$$\begin{aligned} \Phi_2(y(t)) &= y(t) - y_0(t) - \sum_{j=1}^m \mathcal{R}_j(t, t, y(t))W_j(t) - t \sum_{k=1}^{M_1} \mathcal{R}_0(t, t\zeta_k, y(t\zeta_k))\alpha_k \\ &\quad + t \sum_{j=1}^m \sum_{r=1}^K \sum_{i=1}^{M_2} \Lambda_j(t, t\eta_i^{(r)}, y(t\eta_i^{(r)}))W_j(t\eta_i^{(r)})w_i^{(r)}, \end{aligned} \quad (4.3)$$

and the nodes and weights  $\{\zeta_k\}_{k=1}^{M_1}$ ,  $\{\alpha_k\}_{k=1}^{M_1}$ ,  $\{\eta_i^{(r)}\}_{i=1}^{M_2}$ , and  $\{w_i^{(r)}\}_{i=1}^{M_2}$ , for  $r = 1, \dots, K$ , are defined in Eqs. (3.4) and (3.5).

*Proof.* We begin by applying the triangle inequality to the error term  $\|e_N(y(t))\|$ , which yields

$$\|e_N(y(t))\|_\infty \leq \|\Phi_1(y(t)) - \Phi_2(y(t))\|_\infty + \|\Phi_2(y(t)) - \Phi_2(y_N(t))\|_\infty. \quad (4.4)$$

To estimate the first term, we subtract the right-hand sides of (4.2) and (4.3), take absolute values, and apply the triangle inequality once more. This gives

$$\begin{aligned} |\Phi_1(y(t)) - \Phi_2(y(t))| &\leq \left| \int_0^1 \vartheta_0(t, t\tau, y(t\tau))d\tau - \sum_{k=1}^{M_1} \mathcal{R}_0(t, t\zeta_k, y(t\zeta_k))\alpha_k \right| \\ &\quad + \left| \sum_{j=1}^m \int_0^1 \Lambda_j(t, t\eta, y(t\eta))W_j(t\eta)d\eta - \sum_{j=1}^m \sum_{r=1}^K \sum_{i=1}^{M_2} \Lambda_j(t, t\eta_i^{(r)}, y(t\eta_i^{(r)}))W_j(t\eta_i^{(r)})w_i^{(r)} \right|, \end{aligned} \quad (4.5)$$

By invoking classical results on numerical quadrature over the interval  $[0, 1]$  (see the convergence analysis in [4]) and using relation (4.5), we obtain the bound

$$\|\Phi_1(y(t)) - \Phi_2(y(t))\|_\infty \leq \frac{2\mathcal{M}}{(2M_1 - 1)!} + E_{M_2, K}. \quad (4.6)$$

Next, we consider the second term in (4.4). From the representation in (4.3), we have

$$\begin{aligned} |\Phi_2(y(t)) - \Phi_2(y_N(t))| &\leq |y(t) - y_N(t)| + \sum_{j=1}^m |W_j(t)| |\mathcal{R}_j(t, t, y(t)) - \mathcal{R}_j(t, t, y_N(t))| \\ &\quad + |t \sum_{k=1}^{M_1} \alpha_k \left| \mathcal{R}_0(t, t\zeta_k, y(t\zeta_k)) - \mathcal{R}_0(t, t\zeta_k, y_N(t\zeta_k)) \right| \\ &\quad + |t \sum_{j=1}^m \sum_{r=1}^K \sum_{i=1}^{M_2} w_i^{(r)} \left| W_j(t\eta_i^{(r)}) \left| \Lambda_j(t, t\eta_i^{(r)}, y(t\eta_i^{(r)})) - \Lambda_j(t, t\eta_i^{(r)}, y_N(t\eta_i^{(r)})) \right| \right|. \end{aligned}$$



By the Lipschitz continuity assumptions of the theorem together with Theorem 2.6, we obtain the following upper bound

$$\begin{aligned} \|\Phi_2(y(t)) - \Phi_2(y_N(t))\|_\infty &\leq \mathcal{S} \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2\sqrt{N}}{2\gamma}\right) + \sum_{j=1}^m R_j L_{\mathcal{R}_j} \left( \mathcal{S} \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2\sqrt{N}}{2\gamma}\right) \right) \\ &\quad + L_{\mathcal{R}_0} \left( \mathcal{S} \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2\sqrt{N}}{2\gamma}\right) \right) \sum_{k=1}^{M_1} \alpha_k + \sum_{j=1}^m R_j L_{\Lambda_j} \left( \mathcal{S} \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2\sqrt{N}}{2\gamma}\right) \right) \sum_{r=1}^K \sum_{i=1}^{M_2} w_i^{(r)} \\ &\leq 3\mathcal{K} \left( \frac{\sqrt{N}}{(2\iota)^{2N}} \exp\left(\frac{-\pi^2\sqrt{N}}{2\gamma}\right) \right), \end{aligned} \tag{4.7}$$

where

$$\mathcal{K} = \mathcal{S} \max \left\{ 1 + \sum_{j=1}^m R_j L_{\mathcal{R}_j}, L_{\mathcal{R}_0}, \sum_{j=1}^m R_j L_{\Lambda_j} \right\}.$$

Finally, combining (4.4), (4.6), and (4.7), the desired error estimate follows, which completes the proof.  $\square$

The foregoing theorem implies that as  $M_1, N, K \rightarrow \infty$ , then  $\|e_N(y(t))\|_\infty \rightarrow 0$ . Furthermore, the numerical experiments demonstrate that, under sufficient smoothness assumptions and with an appropriate selection of parameters, the proposed method guarantees convergence.

### 5. NUMERICAL RESULTS

In this section, the efficiency of the proposed method in the numerical solution of SVIEs is investigated through comparison with previous studies. It is worth noting that choosing an appropriate  $\gamma$  for the mesh size  $h$  plays a crucial role in achieving desirable results. The computations are performed on a personal computer featuring an Intel Core i5-4200 CPU with 4 GB of RAM, utilizing Maple software version 2018.

Before presenting the numerical examples, we introduce the definitions of the absolute and relative errors employed in this work. Let  $y(t_i)$  and  $y_N(t_i)$  denote the exact and approximate solutions at the grid points  $t_i \in [0, 1]$ , respectively. The absolute error is defined as

$$\text{AbsErr}(t_i) = |y(t_i) - y_N(t_i)|,$$

and the relative error is defined as

$$\text{RelErr}(t_i) = \frac{|y(t_i) - y_N(t_i)|}{|y(t_i)|}.$$

**Example 5.1.** In this example, we consider an Itô-Volterra stochastic integral equation driven by a four-dimensional Brownian motion. The equation is given by

$$y(t) = y_0(t) + \int_0^t \beta(s) y(s) ds + \sum_{i=1}^4 \int_0^t \rho_i(s) y(s) dW_i(s), \quad 0 \leq t \leq 1. \tag{5.1}$$

The exact solution to this equation can be expressed in closed form as

$$y(t) = y_0(t) \exp\left( \left( \beta(t) - \frac{1}{2} \sum_{i=1}^4 \rho_i^2(t) \right) t + \sum_{i=1}^4 \rho_i(t) W_i(t) \right),$$

where  $y_0(t) = \frac{1}{200}$ ,  $\beta(t) = \frac{1}{20}$ , and  $\rho_1(t) = \frac{1}{50}$ ,  $\rho_2(t) = \frac{2}{50}$ ,  $\rho_3(t) = \frac{4}{50}$ ,  $\rho_4(t) = \frac{9}{50}$ .

Table 1 reports a comparative analysis of the absolute error of the proposed method for  $n = 5$  and  $n = 11$  basis functions, against two existing techniques: the stochastic operational matrix approach based on block-pulse basis functions [22] and the stochastic operational matrix approach based on modified hat functions [16] which employ  $m+1$  and  $m$  basis functions, respectively. In addition, for  $n = 5$  and  $n = 11$ , the corresponding relative errors of the proposed method are also presented in Table 1. The numerical results demonstrate that our proposed method



achieves a competitive, and in some cases superior, level of accuracy compared to the methods in [22] and [16], even when using a relatively small to moderate number of basis functions. The results summarized in Tables 1 and 3 are based on  $q = 200$  simulations. The desired values of the parameter  $\nu$  were empirically determined to be 3.95 for  $n = 5$  and 1.90 for  $n = 11$ . Table 4 shows the absolute errors of our method for  $n = 17$  and  $n = 19$  at different values of  $t$ , obtained from a single evaluation. The corresponding  $\gamma$  values for  $n = 15$  and  $n = 19$  are taken as 1.70 and 1.72, respectively. Figures 1 and 2 further illustrate the comparison between the exact and approximate solutions for  $n = 9$  and  $n = 15$ , with the corresponding optimal values of  $\nu$  determined as 2.15 and 1.60, respectively. Finally, Figure 3 illustrates the absolute error of the proposed method for  $n = 9$  and the relative error for  $n = 11$  in Example 5.1.

TABLE 1. Comparison of absolute error obtained using the proposed method and the methods of [16, 22] for Example 5.1, along with the relative errors for  $n = 5$  and  $n = 11$ .

$t$	present method $n = 5$	relative error $n = 5$	the method of [22] $m = 10$	the method of [16] $m = 10$	present method $n = 11$	relative error $n = 11$	the method of [22] $m = 40$	the method of [16] $m = 40$
0.0	$2.4e-06$	$4.7e-04$	0	$7.9e-05$	$7.8e-06$	$1.5e-03$	0	$3.6e-05$
0.1	$2.3e-06$	$4.5e-04$	$8.9e-06$	$5.4e-05$	$5.9e-06$	$1.1e-03$	$1.0e-05$	$1.6e-05$
0.2	$3.3e-05$	$6.5e-03$	$1.5e-05$	$1.4e-04$	$3.8e-06$	$7.3e-04$	$2.2e-05$	$6.6e-05$
0.3	$8.9e-06$	$1.7e-03$	$3.8e-05$	$3.8e-05$	$2.9e-05$	$5.8e-03$	$3.8e-05$	$1.7e-06$
0.4	$1.9e-05$	$3.7e-03$	$3.7e-05$	$3.7e-05$	$7.4e-07$	$1.4e-04$	$5.1e-05$	$7.4e-05$
0.5	$4.9e-05$	$9.5e-03$	$6.7e-05$	$7.2e-05$	$4.7e-05$	$9.2e-03$	$6.5e-05$	$8.9e-05$
0.6	$9.8e-05$	$1.9e-02$	$6.2e-05$	$7.1e-05$	$8.2e-05$	$1.6e-02$	$7.9e-05$	$4.8e-05$
0.7	$8.8e-05$	$1.7e-02$	$9.0e-05$	$1.0e-04$	$7.1e-05$	$1.3e-02$	$9.6e-05$	$4.8e-05$
0.8	$1.0e-04$	$1.9e-02$	$8.2e-05$	$9.7e-05$	$8.0e-05$	$1.5e-02$	$1.1e-04$	$1.2e-04$
0.9	$1.0e-04$	$2.0e-02$	$1.0e-04$	$5.7e-05$	$8.7e-05$	$1.6e-02$	$1.0e-04$	$5.2e-05$
1.0	$1.0e-04$	$2.0e-02$	$9.4e-05$	$1.0e-04$	$9.3e-05$	$1.7e-02$	$1.0e-04$	$5.1e-03$

TABLE 2. Mean, standard deviation, and 95% mean confidence intervals based on 200 simulations for Example 5.1.

$n$	$\bar{X}_E$	$S_E$	95% confidence interval for mean of $E$	
			Lower bound	Upper bound
3	$5.182e-05$	$3.956e-05$	$4.411e-05$	$5.954e-05$
5	$5.512e-05$	$4.358e-05$	$4.662e-05$	$6.362e-05$
7	$6.007e-05$	$3.874e-05$	$5.251e-05$	$6.763e-05$
9	$5.029e-05$	$3.470e-05$	$4.352e-05$	$5.706e-05$
11	$4.901e-05$	$3.491e-05$	$4.220e-05$	$5.582e-05$
13	$5.718e-05$	$3.680e-05$	$5.000e-05$	$6.435e-05$
15	$5.594e-05$	$3.559e-05$	$4.900e-05$	$6.288e-05$
17	$5.726e-05$	$4.598e-05$	$4.829e-05$	$6.623e-05$



TABLE 3. Mean, standard deviation, and 95% mean confidence intervals based on 200 simulations for Example 5.1.

$n$	$\bar{X}_E$	$S_E$	95% confidence interval for mean of $E$		CPU times (s)
			Lower bound	Upper bound	
3	$5.182e-05$	$3.956e-05$	$4.411e-05$	$5.954e-05$	3.968
5	$5.512e-05$	$4.358e-05$	$4.662e-05$	$6.362e-05$	3.157
7	$6.007e-05$	$3.874e-05$	$5.251e-05$	$6.763e-05$	4.001
9	$5.029e-05$	$3.470e-05$	$4.352e-05$	$5.706e-05$	4.141
11	$4.901e-05$	$3.491e-05$	$4.220e-05$	$5.582e-05$	5.122
13	$5.718e-05$	$3.680e-05$	$5.000e-05$	$6.435e-05$	5.492
15	$5.594e-05$	$3.559e-05$	$4.900e-05$	$6.288e-05$	7.626
17	$5.726e-05$	$4.598e-05$	$4.829e-05$	$6.623e-05$	10.502

TABLE 4. The absolute errors of Example 5.1 for  $n = 17, n = 19$  at different values of  $t$  and  $q = 1$ .

$t$	$n = 17$	$n = 19$
0.0	$2.623e-06$	$4.686e-07$
0.1	$1.599e-06$	$7.894e-07$
0.2	$4.779e-06$	$1.135e-06$
0.3	$9.752e-06$	$6.094e-07$
0.4	$5.176e-06$	$1.470e-05$
0.5	$4.857e-06$	$5.260e-06$
0.6	$4.374e-05$	$2.606e-06$
0.7	$3.226e-05$	$2.752e-05$
0.8	$2.296e-05$	$1.670e-05$
0.9	$9.517e-07$	$2.183e-06$
CPU times (s)	0.047	0.078

**Example 5.2.** We consider the following Itô–Volterra stochastic integral equation driven by a three-dimensional Brownian motion

$$y(t) = y_0(t) + \int_0^t \beta(s)y(s) ds + \sum_{i=1}^3 \int_0^t \rho_i(s)y(s) dW_i(s), \quad 0 \leq t \leq 1, \tag{5.2}$$

where  $y_0(t) = \frac{1}{12}$ ,  $\beta(t) = \frac{1}{30}$ ,  $\rho_1(t) = \frac{1}{10}$ ,  $\rho_2(t) = t^2$ , and  $\rho_3(t) = \frac{\sin(t)}{3}$ . The exact solution of this equation is given by

$$y(t) = y_0(t) \exp \left( \int_0^t \left( \beta(s) - \frac{1}{2} \sum_{i=1}^3 \rho_i^2(s) \right) ds + \sum_{i=1}^3 \int_0^t \rho_i(s) dW_i(s) \right).$$

Table 5 presents a comparison of the absolute errors obtained by the proposed method for  $n = 5$  and  $n = 9$ , with those achieved by the fifth-order spline collocation method [20], using  $m = 16$  and  $m = 32$ . In addition, for  $n = 9$ , the corresponding relative errors of the proposed method are also presented in Table 5. The reported results are based on  $q = 1$ . Moreover, the parameter  $\gamma$  is empirically set to 2.80 and 1.65 for  $n = 5$  and  $n = 9$ , respectively. Figure 4 illustrates a comparison between the exact and approximate solutions for  $n = 9$ , whereas Figure 5 depicts the absolute error of the proposed method for  $n = 5$ .



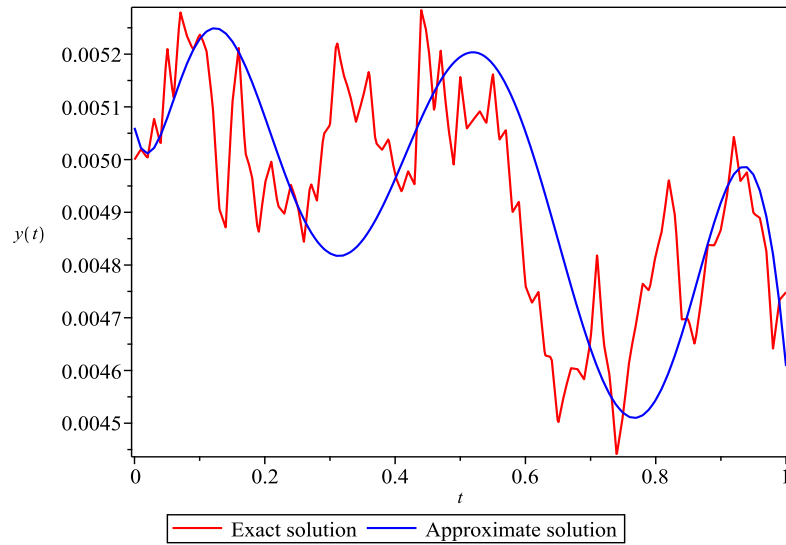


FIGURE 1. Comparison between the exact solution and the approximate solution with  $n = 9$  for Example 5.1.

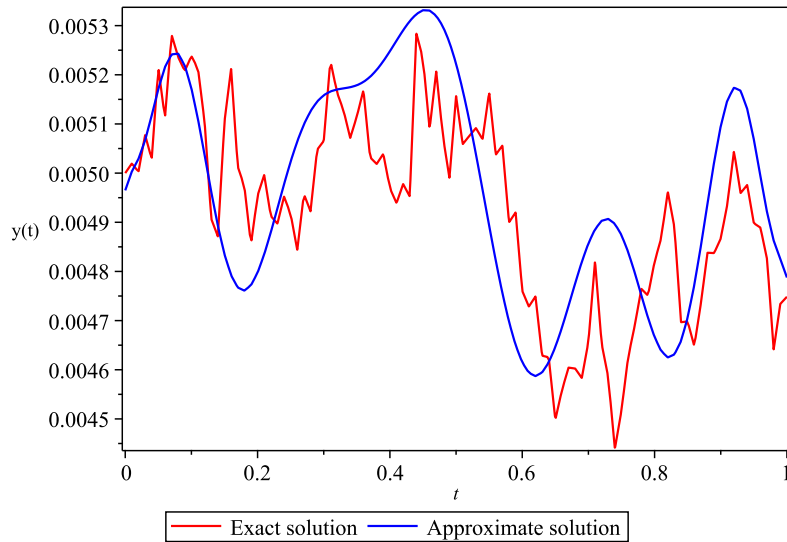
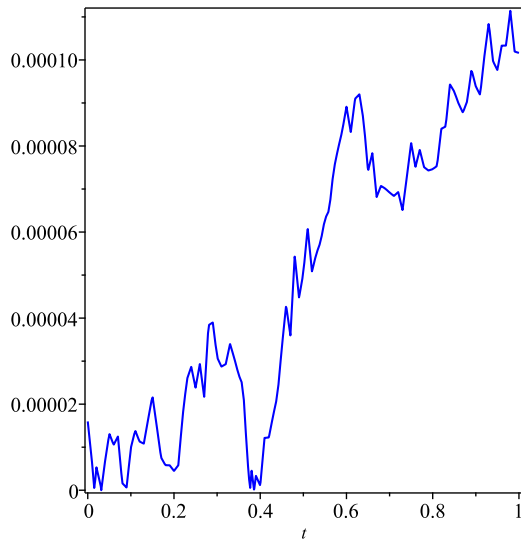


FIGURE 2. Comparison between the exact solution and the approximate solution with  $n = 15$  for Example 5.1.

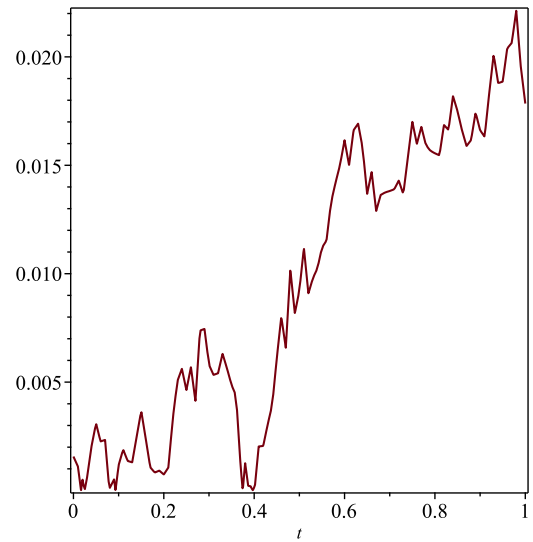
**Example 5.3.** In this example, we consider a nonlinear Itô–Volterra stochastic integral equation driven by a two-dimensional Brownian motion, defined as

$$\begin{aligned}
 y(t) = & y_0(t) + \int_0^t t \exp(-(t+s)) y(s) ds + \mu \int_0^t t y(s)(s - y(s)) dW_1(s) \\
 & + \lambda \int_0^t \sin(ty(s)) dW_2(s),
 \end{aligned} \tag{5.3}$$





(A) Absolute error for  $n = 9$ .



(B) Relative error for  $n = 11$ .

FIGURE 3. Plots of the absolute and relative errors for Example 5.1.

TABLE 5. Comparison of absolute error obtained using the proposed method and the methods of [16, 22] for Example 5.2, along with the relative error for  $n = 9$ .

$t$	Proposed method $n = 5$	Method of [20] $m = 16$	Proposed method $n = 9$	relative error $n = 9$	Method of [20] $m = 32$
0.0	$1.2e - 04$	0.0	$6.5e - 03$	$7.8e-02$	0.0
0.1	$3.6e - 04$	$2.7e - 03$	$3.7e - 04$	$4.4e-03$	$2.0e - 03$
0.2	$1.4e - 04$	$2.0e - 03$	$3.0e - 04$	$3.5e-03$	$2.8e - 03$
0.3	$4.2e - 03$	$3.1e - 03$	$3.0e - 04$	$3.6e-03$	$1.5e - 03$
0.4	$3.9e - 03$	$1.5e - 03$	$1.3e - 03$	$1.5e-02$	$4.8e - 03$
0.5	$2.4e - 04$	$9.7e - 03$	$1.5e - 04$	$1.6e-03$	$5.8e - 03$
0.6	$4.9e - 03$	$2.2e - 03$	$1.2e - 03$	$1.3e-02$	$2.0e - 03$
0.7	$1.2e - 02$	$7.7e - 03$	$3.5e - 03$	$4.4e-02$	$2.2e - 03$
0.8	$2.4e - 03$	$3.1e - 03$	$3.7e - 04$	$4.0e-03$	$8.2e - 03$
0.9	$1.5e - 02$	$2.7e - 03$	$3.1e - 02$	$3.9e-01$	$5.4e - 03$
1.0	$2.3e - 02$	—	$1.4e - 03$	$1.7e-02$	—

where  $y_0(t) = t^2 + 1$  and  $t \in [0, 1]$ .

Figure 6 illustrates the behavior of the numerical solutions in the deterministic case ( $\mu = \lambda = 0$ ) and for various combinations of  $\mu$  and  $\lambda$ , obtained using  $n = 5$ ,  $\gamma = 2.25$ , and  $q = 1$ . Furthermore, Figure 7 presents the absolute



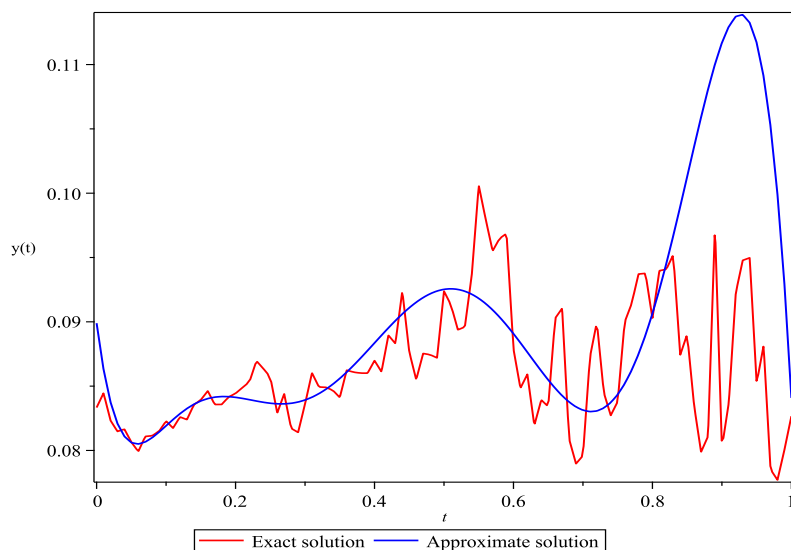


FIGURE 4. Comparison between the exact solution and the approximate solution for  $n = 9$  in Example 5.2.

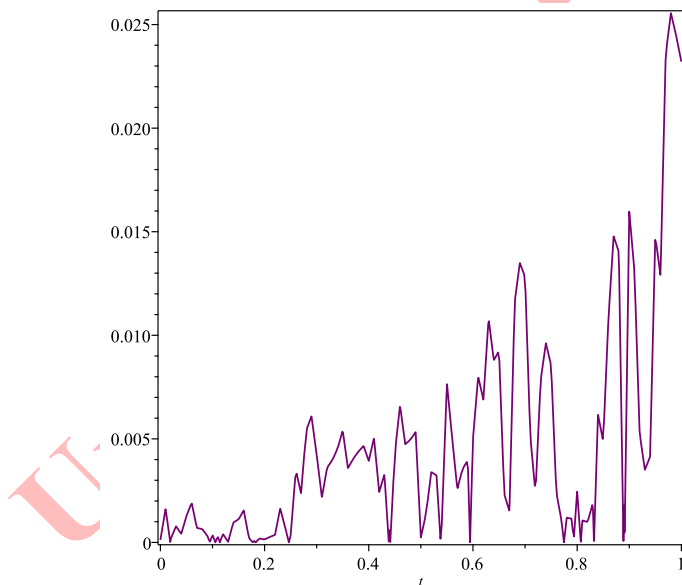


FIGURE 5. Absolute error plot for Example 5.2 with  $n = 5$ .

values of the residual errors at the nodal points  $t_i = \frac{i}{50}$ , for  $i = 0, 1, \dots, 50$ , in the deterministic scenario ( $\mu = \lambda = 0$ ), computed with  $n = 15$ ,  $\gamma = 1.55$ , and  $q = 1$ .

## 6. CONCLUSION

In this study, a spectral collocation method for solving m-dimensional stochastic Volterra integral equations is presented. This method uses sinc polynomial basis functions. We transform the original problem into a system of algebraic equations, which is then efficiently solved by standard numerical techniques. The deterministic and Itô



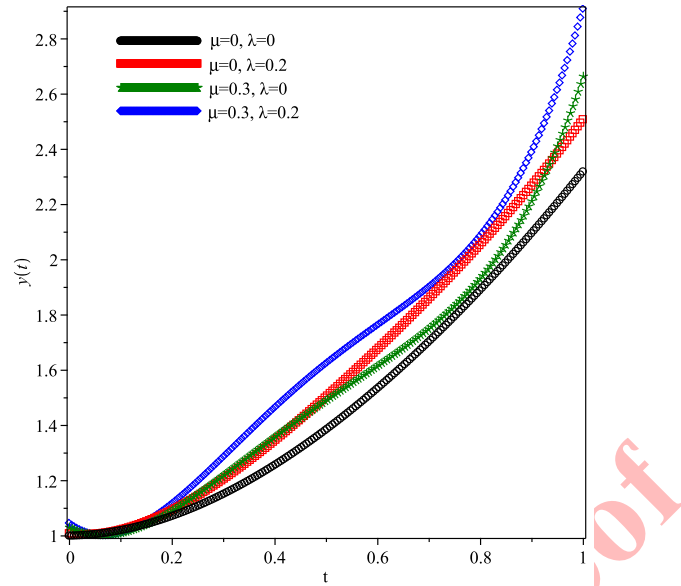


FIGURE 6. Comparison between the deterministic solution and the stochastic solutions for Example 5.3.

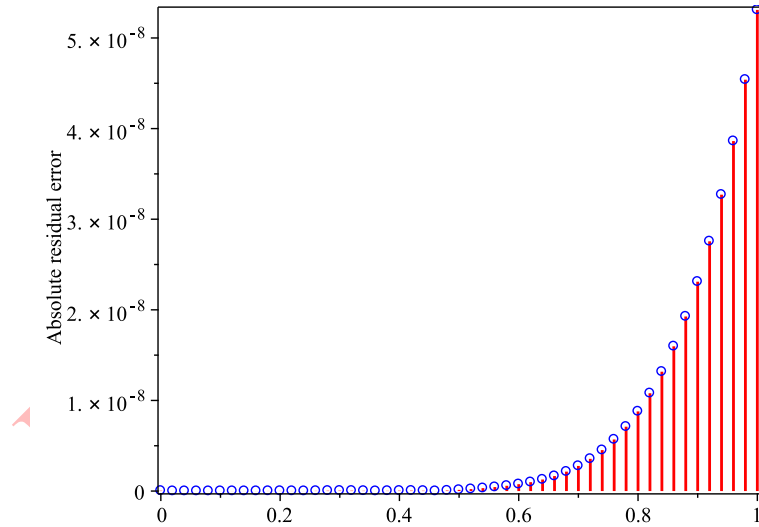


FIGURE 7. Absolute residual error curve for Example 5.3 with  $\mu = \lambda = 0$ .

integrals are approximated by Gauss–Legendre and composite Newton–Cotes quadratures, respectively. One main advantage of our approach is that it achieves high accuracy with a relatively small number of basis functions. This reduces the computational complexity in comparison to some existing methods. Numerical experiments demonstrate that the accuracy and efficiency of the method are strongly influenced by the choice of the parameter  $\gamma$ . When this parameter is well-tuned, our results are competitive and, in many cases, superior to those of other methods. In conclusion, the proposed framework provides a simple, accurate, and computationally efficient tool for the numerical



solution of SVIEs. we believe it holds strong potential for extension to more general classes of stochastic systems and it provides a powerful and computationally efficient tool for researchers working with SVIEs in various scientific and engineering disciplines.

#### ACKNOWLEDGMENT

We sincerely thank the anonymous reviewer for their valuable comments and constructive suggestions, which significantly improved our manuscript.

#### REFERENCES

- [1] M. Ahmadiania, H. Afshariarjmand, and M. Salehi, *Numerical solution of multi-dimensional Itô Volterra integral equations by the second kind Chebyshev wavelets and parallel computing process*, Appl. Math. Comput., 450 (2023), 127988.
- [2] A. Alipanah, K. Mohammadi, and R. M. Haji, *Numerical solution of singularly perturbed singular third order boundary value problems with nonclassical sinc method*, Results Appl. Math., 22 (2024), 100459.
- [3] K. Atkinson, and W. Han, *Theoretical numerical analysis: a functional analysis framework*, New York, NY: Springer New York, 2005.
- [4] F. Bahmani and A. Eftekhari, *Numerical Solution of Stochastic Fractional Integro-Differential Equations: The Poly-sinc Collocation Approach*, Iran. J. Sci., 49(1) (2025), 117–132.
- [5] G. Baumann, *New Sinc Methods of Numerical Analysis*, Springer International Publishing, 2021.
- [6] I. Boukhelkhal and R. Zeghdane, *Lagrange interpolation polynomials for solving nonlinear stochastic integral equations*, Numer. Algorithms, 96(2) (2024), 583–618.
- [7] J. P. Boyd, *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.
- [8] L. M. Delves and J. L. Mohamed, *Computational methods for integral equations*, CUP Archive, 1985.
- [9] A. Eftekhari, *Spectral Poly-Sinc Collocation Method for Solving a Singular Nonlinear BVP of Reaction-Diffusion with Michaelis-Menten Kinetics in a Catalyst/Biocatalyst*, Iran. J. Math. Chem., 14(2) (2023), 77–96.
- [10] G. Jiang, T. Ke, and M. T. Deng, *Least square method based on Haar wavelet to solve multi-dimensional stochastic Itô-Volterra integral equations*, Appl. Math. J. Chin. Univ., 38(4) (2023), 591–603.
- [11] T. Ke, G. Jiang, and M. Deng, *Numerical Solution of Multidimensional Stochastic Itô-Volterra Integral Equation Based on the Least Squares Method and Block Pulse Function*, Math. Probl. Eng., 2021(1) (2021), 6662604.
- [12] O. A. Khalil and G. Baumann, *An h-Adaptive Poly-Sinc-Based Local Discontinuous Galerkin Method for Elliptic Partial Differential Equations*, Axioms, 12(3) (2023), 227.
- [13] O. Khalil, H. El-Sharkawy, M. Youssef, and G. Baumann, *Adaptive piecewise Poly-Sinc methods for ordinary differential equations*, Algorithms, 15(9) (2022), 320.
- [14] M. Khodabin, K. Maleknejad, and F. Hossoini Shekarabi, *Application of triangular functions to numerical solution of stochastic Volterra integral equations*, IAENG Int. J. Appl. Math., 43(1) (2013), 1–9.
- [15] J. L. Lund and K. L. Bowers, *Sinc Methods for Quadrature and Differential Equations*, SIAM, Philadelphia, PA, 1992.
- [16] K. Maleknejad, M. Khodabin, and M. Rostami, *A numerical method for solving m-dimensional stochastic Itô-Volterra integral equations by stochastic operational matrix*, Comput. Math. Appl., 63(1) (2012), 133–143.
- [17] F. Mirzaee, N. Samadyar, and S. F. Hoseini, *Euler polynomial solutions of nonlinear stochastic Itô-Volterra integral equations*, J. Comput. Appl. Math., 330 (2018), 574–585.
- [18] F. Mirzaee, E. Solhi, and N. Samadyar, *Moving least squares and spectral collocation method to approximate the solution of stochastic Volterra-Fredholm integral equations*, Appl. Numer. Math., 161 (2021), 275–285.
- [19] F. Mirzaee, S. Naserifar, and E. Solhi, *Accurate and stable numerical method based on the Floater-Hormann interpolation for stochastic Itô-Volterra integral equations*, Numer. Algorithms, 94(1) (2023), 275–292.
- [20] F. Mirzaee and S. Alipour, *Quintic B-spline collocation method to solve n-dimensional stochastic Itô-Volterra integral equations*, J. Comput. Appl. Math., 384 (2021), 113153.
- [21] F. Mohammadi, *Haar wavelets approach for solving multidimensional stochastic Itô-Volterra integral equations*, Appl. Math. E-Notes, 15 (2015), 80–96.



- [22] N. Momenzade, A. R. Vahidi, and E. Babolian, *A computational method for solving stochastic Itô–Volterra integral equation with multi-stochastic terms*, *Math. Sci.*, *12* (2018), 295–303.
- [23] B. Oksendal, *Stochastic differential equations: an introduction with applications*, Springer Science & Business Media, 2013.
- [24] S. S. Ray and P. Singh, *Numerical solution of stochastic Itô–Volterra integral equation by using Shifted Jacobi operational matrix method*, *Appl. Math. Comput.*, *410* (2021), 126440.
- [25] A. Saadatmandi, A. Asadi, and A. Eftekhari, *Collocation method using quintic B-spline and sinc functions for solving a model of squeezing flow between two infinite plates*, *Int. J. Comput. Math.*, *93*(11) (2016), 1921–1936.
- [26] M. Saffarzadeh, M. Heydari, and G. B. Loghmani, *Convergence analysis of an iterative numerical algorithm for solving nonlinear stochastic Itô–Volterra integral equations with  $m$ -dimensional Brownian motion*, *Appl. Numer. Math.*, *146* (2019), 182–198.
- [27] M. Saffarzadeh, G. B. Loghmani, and M. Heydari, *An iterative technique for the numerical solution of nonlinear stochastic Itô–Volterra integral equations*, *J. Comput. Appl. Math.*, *333* (2018), 74–86.
- [28] J. Shen, T. Tang, and L.L. Wang, *Spectral methods: algorithms, analysis and applications*, Vol. 41. Springer Science & Business Media, 2011.
- [29] P. K. Singh and S. Saha Ray, *A novel operational matrix method based on Genocchi polynomials for solving  $n$ -dimensional stochastic Itô–Volterra integral equation*, *Mathematical Sciences*, *18*(2) (2024), 305–315.
- [30] E. Solhi, F. Mirzaee, and S. Naserifar, *Enhanced moving least squares method for solving the stochastic fractional Volterra integro-differential equations of Hammerstein type*, *Numer. Algorithms*, *95*(4) (2024), 1921–1951.
- [31] F. Stenger, *Approximations via Whittaker’s cardinal function*, *J. Approx. Theory*, *17*(3) (1976), 222–240.
- [32] F. Stenger, *A Sinc-Galerkin method of solution of boundary value problems*, *Math. Comp.*, *33*(145) (1979), 85–109.
- [33] F. Stenger, *Numerical Methods Based on Sinc and Analytic Functions*, Springer-Verlag, New York, 1993.
- [34] F. Stenger, *Polynomial function and derivative approximation of Sinc data*, *J. Complexity*, *25* (2009), 292–302.
- [35] F. Stenger, *Handbook of Sinc numerical methods*. CRC Press, 2016.
- [36] F. Stenger, M. Youssef, and J. Niebsch, *Improved approximation via use of transformations*, *Multiscale Signal Analysis and Modeling*, 2013, 25–49.
- [37] S. Taherkhani, I. Najafi Khalilsaraye, and B. Ghayebi, *A pseudospectral Sinc method for numerical investigation of the nonlinear time-fractional Klein-Gordon and sine-Gordon equations*, *Comput. Methods Differ. Equ.*, *11*(2) (2023), 357–368.
- [38] L. N. Trefethen, *Approximation theory and approximation practice, extended edition*. Society for Industrial and Applied Mathematics, 2019.
- [39] M. Youssef and G. Baumann, *Solution of nonlinear singular boundary value problems using polynomial-Sinc approximation*, *Commun. Fac. Sci. Univ. Ankara Ser. A1 Math. Stat.*, *63*(2) (2014), 41–58.
- [40] M. Youssef, H. A. El-Sharkawy, and G. Baumann, *Lebesgue constant using sinc points*, *Adv. Numer. Anal.*, *2016*(1) (2016), 6758283.
- [41] M. Youssef and R. Pulch, *Poly-Sinc solution of stochastic elliptic differential equations*, *J. Sci. Comput.*, *87*(3) (2021), 82.
- [42] X. Zhang, J. Huang, and X. Wen, *A new numerical algorithm based on least squares method for solving stochastic Itô–Volterra integral equations*, *Numer. Algorithms*, *98*(1) (2025), 117–132.

