JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-21, No.-09, September (2026) pp 73-99 ISSN (Print) 0973-8975

# SOLVING STOCHASTIC INTEGRO-DIFFERENTIAL EQUATIONS ON TIME SCALES: LAPLACE–ADOMIAN DECOMPOSITION METHOD VERSUS MACHINE LEARNING MODELS

**Asmaa A. Mahdi[1], Ruqaia Jwad Kadhim[2], Hasanain Jalil Neamah Alsaedi[3], Adel S. Hussain[4], Rana Aziz Yousif Almuttalibi[5]**

[1, 2, 3] University of Information Technology and Communication, Baghdad, Iraq.

[3]Mustansiriyah University, Baghdad, Iraq.

[4]Department of IT, Amedi Technical Institutes, University of Duhok Polytechnic, Duhok, Iraq.

[4]Department of Administrative and Financial Affairs, Faculty Division, Al-Iraqi University, Baghdad, Iraq.

[4]Department of Computer Engineering, Al-Kitab University, Altun, Kupri, Kirkuk, Iraq.

[5]University Information Technology and Communication, Baghdad, Iraq.

Corresponding Author: **Hasanain Jalil Neamah Alsaedi**

Email: [1]Asmaa.mahdi@uoitc.edu.iq, [2]Roqaia.jwad@uoitc.edu.iq, [3]Hasanien.1975@uoitc.edu.iq, [4]Adel.sufyan@dpu.edu.krd, [5]Rana.aziz@uoitc.edu.iq

https://doi.org/10.26782/jmcms.2026.09.00005

(Received: June 10, 2026; Revised: August 29, 2026; Accepted: September 12, 2026)

**Abstract**

*We generalize the Laplace Adomian Decomposition Method (LADM) to solve both linear and nonlinear stochastic integro-differential equations that are constructed on arbitrary time scales. The presented method implements the time-scale Laplace transform to transform delta-derivative terms into the Laplace domain and build an Adomian series of the nonlinearities, and uses the inverse transform to obtain time-domain solutions; the stochastic integrals not in closed form are approximated by Physics-Informed Neural Networks (PINNs) trained to satisfy Ito-type integral equations. On illustrative linear and nonlinear Volterra problems, we validate the method where the LADM series recovers solutions to the ADM equations within the truncation error (examples show agreement to* $\mathbf{10^{-3}}$ *in mean absolute error) with fewer explicit evaluations of integrals than ADM (reduction in the number of convolution/evaluation steps, approximately 4070 of examples). At points where stochastic integrals are approximated, PINNs give path-wise means that are similar to Monte-Carlo estimates (Table 3). We make the following contributions: (i) we* *provide a time-scale compatible LADM formulation of Ito-type stochastic forcing; (ii) we show computational savings of LADM over classical ADM on nonstandard time scales; and (iii) we provide a viable LADM-PINN hybrid that computes non-closed-form stochastic integrals. Rigorous convergence and variance-reduction constraints and guidelines are mentioned.*

**Keywords:** Stochastic Integro-differential equations, Laplace Adomian decomposition method, Laplace transform, Adomian polynomial, Physics-Informed Neural Networks (PINNs).

## 1. Introduction

Several scientific and engineering occurrences rely significantly on integro-differential equations for mathematical modeling. In systems where the current state is dependent on both the pace of change and the completion of previous actions, the equations emerge spontaneously. By way of electric circuits, the second law of Kirchhoff shows that in loops with resistors, inductors, and capacitors, the total voltage drop is given by an integro-differential equation. This mathematical relation has many applications in real-world systems because it depicts the dynamic interplay between electrical measurements taken at different times [I, III, IX, XII].

$$L \frac{d}{dt} I(t) = RI(t) + \frac{1}{k}\int_0^t I(s)ds = E(t).$$

Science first derived integral equations from applications of water waves and conformal mapping in mathematical physics and quantum mechanics, and diffraction. These tools later spread into economic and biological research along with viscoelasticity and electromagnetic control theory and chemistry. The broad analytical applications that quantify the importance of integral equations for theoretical and practical analysis. For further details, see references [I, III, IV, V, IX, XII, XVI, XX, XXIII, XXVIII, XXIX].

Systems exhibiting both discrete and continuous dynamics can be studied using the time scale framework as a unified analytical tool. Any scientific discipline that needs to represent processes including both discrete and continuous characteristics can benefit from this mathematical tool. Such hybrid architectures are inherent to many real-world occurrences. An ideal instance of this type of dynamical system would be time-scale population dynamics that deal with generations that do not overlap [XXV]. Since the complexity of population growth cannot be adequately captured by conventional difference and differential equations, time scale calculus becomes crucial for population modeling [I].

There is a lack of published research on the topic of time scales theory as it pertains to the treatment of integral and integro-differential equations. While a large body of literature investigates solutions to these equations using methods based on classical calculus, time scale-based approaches remain under-explored. In both studies, LADM is a commonly used method for dealing with integral equations, both linear and nonlinear [XVI, XXX]. Analytical solutions of generic differential systems were generated by modifying the Adomian Decomposition Method (ADM) in the work of Aisha et al. in [II]. By bringing together Sumudu transformations, Shams and Tarig [XXXVII] broadened this line of inquiry. For the second class of nonlinear Volterra integro-differential equations, the authors in [XXI] used the Sumudu transform in conjunction with ADM. As stated in [XXXV], the Modified Decomposition Method (MDM) in conjunction with ADM has been successfully applied to solve systems of nonlinear partial differential equations. Researchers improved the conventional ADM convergence in [XXXIV] by incorporating a parametric acceleration method. In order to effectively analyze nonlinear systems, several improvements to standard analytical methods have been developed. According to the research found in [XXI], the original Variational Iteration Method (VIM) was improved through the use of convergence-accelerating parameters. Concrete numerical implementations show that the optimal VIM outperforms both standard ADM and the classic VIM. In order to demonstrate the convergence of solution series and the accuracy of the analytical solution obtained, researchers in [XXXV] utilized a convergence control parameter to apply ADM. In order to simulate physical phenomena, such as optical fiber impulses, one must use integro-differential equations [XXVI, XXVII, XXXVIII]. In order to find solutions for complex integro-differential systems, numerical experts have developed a number of sophisticated approximation approaches. By combining an expanded version of moving least squares with the spectral collocation approach, the writers of [XXXVI] were able to solve a nonlinear stochastic Volterra integro-differential equation. In [XXII], the cubic B-spline collocation method was used as a strategy to solve stochastic integro-differential equations of fractional order. Mirzaee et al. [XVII] state that, to approximate Volterra integro-differential equations using pantograph-delay structures, Euler polynomials were employed. The study in [XVIII] created a successful numerical technique for approximating the solution of nonlinear fractional Volterra integro-differential equations by utilizing Euler operative matrix for β-order fractional derivatives.A universal Laplace transform is developed by the authors for use with stochastic processes of the Itô-Doob type. While acknowledging that traditional Laplace transforms have problems in a stochastic environment, they use integration by parts to solve a broader class of stochastic integrals and equations within the stochastic calculus framework. In the first part of the study, the Laplacian and Laplace transforms are defined, with and without random components, and when they are valid and linear are established. Several key findings pertain to computations for functions subject to Brownian motion, indefinite stochastic integrals, and transforms of derivatives. The authors demonstrate the usefulness and mathematical soundness of their framework by solving several well-known stochastic differential equations in closed form, including the Langevin and Chandrasekhar equations [XV]. Researchers apply the most recent iteration of the Neural Laplace framework—originally developed for ordinary differential equations (ODEs)—to Langevin stochastic differential equations (SDEs), with a focus on GBM. The author demonstrates the theory of the Laplace Transform by showing that the Laplace transform of GBM is firm and may be closely approximated in low-volatility, large-drift, and low-variance instances. The study demonstrates that Neural Laplace does well when learning about SDEs, mixes theory and practice, and compares it to other neural models. In addition to paving the way for more study into generally applicable SDE learning, this work facilitates the application of Laplace domain neural approaches to stochastic systems [VI]. The authors develop and demonstrate that this inference method applies to nonlinear SDEs with state-dependent noise using solely observed data indicating noise. The authors propose a method based on the Laplace approximation since assessing the probability in large dimensions is not feasible due to the unknown transition density. This model breaks down time into discrete steps, inserts latent states between observations, and uses methods like Euler-Maruyama to approximately estimate transition probabilities. Even in the most complicated data sets, Laplace's approach makes inference possible by combining expert judgments about unseen variables. Important problems like computational and physiological noise are addressed by evaluating several discretization methods, Itô and Stratonovich, and by proposing various models that use state and Brownian increment variables. A stochastic predator-prey model, a Cox-Ingersoll-Ross process, and linear models are employed by the authors to demonstrate that the Laplace approximation is effective in forecasting solutions for SDEs, is fast, and simple [XXXIII]. They demonstrate the value of Caputo derivatives in describing the intricacies of infectious diseases by exploring their successful application in traditional epidemiological models. The author methodically applies the Caputo derivative to transform popular ordinary differential equation (ODE) models, including the SIR, SEIR, MSIR, MSEIR, and double-SI, into fractional calculus equations. Following this, the method employs the Laplace transformation on these fractional systems in order to simplify the mathematics and enable parameter estimation. To further enhance our qualitative understanding of epidemic evolution, the models are organized in matrices, allowing us to apply linear algebra analysis to verify stability and equilibrium. Researchers may now investigate epidemiological behavior over the long run and identify efficient control measures by mixing fractional calculus with Laplace transforms. This new method could lessen the use of numerical approximation methods for specific circumstances [XI].Classical calculus defines the issues addressed in each of the research papers. Although scholars have mostly focused on this area, the topic of time scales serves as a connecting theory for both discrete and continuous analysis. In [X], the Laplace Adomian Decomposition Method (LADM) was used to tackle initial value issues on time scales. Linear and nonlinear integro-differential equations on time scales can be solved using the conventional Adomian Decomposition Method (ADM), as shown in the study in [XXIII].The analytical series solutions to stochastic equations, both linear and nonlinear, defined on any time scale, are obtained in this study by applying LADM. While utilizing a significantly smaller number of operations, LADM is able to produce results that are identical to those found in [XXIII] when employing ADM. For time scale definitions, LADM outperforms ADM when applied since it reduces the number of integration steps and overall computing demands. These developments contribute to the growth of time scale calculus analytical methods and strengthen the practical value of LADM for addressing complicated hybrid dynamical systems. Since stochastic integro-differential equations have many real-world applications, finding efficient solutions to these equations is an important area of study [XXX], [I], [XXVII].Here we introduce the LADM, a tool for solving stochastic integro-differential equations on any time scale, both linear and nonlinear. To begin, let's use the linear situation as our basis example.

$$\emptyset^{\Delta^n}(y,w) = v(y) + \int_0^y G(y,s)\,\emptyset(y,w)\Delta y + \sigma(y,w)W(y),\ \emptyset(0) = 0, \emptyset^{\Delta}(0) = 0, \dots, \emptyset^{\Delta(n-1)}(0) = 0. \quad (1)$$

And then the nonlinear equation

$$u^{\Delta^n}(y,w) = v(y) + \int_0^y G(y,s)\, H\big(u(y,w)\big)\Delta y + \sigma(y)W(y,w), u^{\Delta^r}(y,w) = 0,$$
$$r = 0,1,\ldots,n-1 \tag{2}$$

An extension of the integro-differential model to stochastic time scales is used to depict uncertain systems by adding stochastic components. Uncertainty in the real world is connected to deterministic hybrid dynamical systems in this paradigm. The Adomian Decomposition Method (ADM) and its Laplace version are classical analytical procedures that have been successful in solving deterministic integral and differential problems. They become difficult to apply when the underlying system has been stochastically forced or when the underlying time scale has been arbitrarily long and combines discrete and continuous areas.

To add insult to injury, the current research uses computationally demanding and uncertainly closed-form accurate Monte-Carlo or perturbation approaches to estimate stochastic integrals. In this paper, we address these problems by creating a Laplace Adomian Decomposition Method (LADM) that can be applied to various time scales and solved using Physics-Informed Neural Networks (PINNs) for non-closed-form stochastic integrals. Here are the main takeaways from the study:

1. Constructing a generalized stochastic integro-differential problem of time scale that integrates discrete and continuous dynamics.
2. Developing a modified LADM technique that incorporates a time-scale Laplace transform besides Adomian polynomials to compute semi-analytical series solutions.
3. A stochastic rough calculation stands for a PINN and stochastic constraints to solve intractable Ito to integrals utilizing a neural network as a surrogate.
4. Demonstration and verification of the hybrid LADM-PINN algorithm's strong accuracy (error 10–70) and computing efficiency (7040 savings) on a few representative linear and nonlinear scenarios.

### I.i. Problem Formulation

On any given time scale, think about the stochastic integro-differential equation $\mathbb{T} \subset \mathbb{R}$,

$$\mathcal{L}_{\mathbb{T}}[y](t) = f\left(t, y(t), \int_a^t K\big(t,s,y(s)\big)\Delta s\right) + \sigma(t, y(t))\dot{W}(t), t \in [a,b]_{\mathbb{T}}, \tag{3}$$

Based on to an initial condition

$$y(a) = y_0. \tag{4}$$

In this context, $\mathcal{L}_{\mathbb{T}}$ stands for the time-dependent delta-derivative operator, $\mathbb{T}, K(t,s,\cdot)$ has been an established kernel function that might not be linear in y. W(t) stands for a Wiener process that describes random disturbances. By creating a T-adapted LADM and a PINN-based arrangement for estimating stochastic integrals without closed-form expressions, for finding an analytical rough calculation for y(t) that holds for discrete and continuous time scales.

### I.i. Unique properties for a problem

A time-scale outline provides a unified formulation for both continuous-time and discrete-time scenarios. Approximation based on neural networks is necessary when there are stochastic integrals with non-explicit Itô terms. To efficiently solve the problem with nonlinear kernels K(t,s,y), Adomian polynomial decomposition is required.

## II. Preliminaries

Here, many theorems and definitions have been depicted for serving as the basis for the analysis and development of the suggested method.

### II.i. Time scale

Each nonempty closed subset of the real numbers R is represented by a time scale, or T. This approach allows the single conceptual framework to include both discrete and continuous analysis. Three specific examples of valid time scales are the set of natural numbers N, the real line R, and the closed interval [0, 3]. We cannot use the intervals (0,2] and (0,1) as time scales because of the closedness requirement in R, which also applies to the irrational numbers Q' and the rational numbers Q.

### II.ii. Forward jump operator

An operator of forward jump stands for a function $\sigma: T \to T$, distinct through

$$\sigma(t) = inf\{z \in T : z > t\}$$

Where $\sigma(t) \geq t$ for any $t \in T$.

### II.iii. Backward jump operator

A backward jump operator stands for a function $\rho: T \to T$, distinct through

$$\rho(t) = sup\{z \in T: z < t\}$$

as $\rho(t) \leq t$ for each $t \in T$.

### II.iv. Graininess function

A graininess stands for a function $\mu: T \to [0, \infty)$ stated through

$$\mu(t) = \sigma(t) - t \quad \forall \quad t \in T.$$

### II.v. Delta derivative

Before defining the delta derivative, it can be defined based on

$$T^k = \begin{cases} T \setminus \rho(\,sup\, T),\ sup\, T & in\ the\ case\ of\ \ sup\, T < \infty \\ T, & Otherwise \end{cases}$$

Suppose $f : T \to R$ is a function along with $t \in T^k$. It can be defined as $f^{\Delta}(t)$ to be a number in the case of $\epsilon > 0$, $U$ (a neighborhood) for $t$ is in effect. $U$ can be defined as

$U = (t - \delta, t + \delta) \cap T$, based on $\delta > 0$, so as to

$$\left|f(\sigma(t)) - f(z) - f^{\Delta}(t)(\sigma(t) - z)\right| \leq \epsilon|\sigma(t) - z|,$$

holds for all $z \in U$. While, $f^{\Delta}(t)$ has been known as Hilger or delta derivative for $f$ at $t$. Namely, $f$ stands for delta differentiable in $T^k$ $\forall$ $t \in T^k$. If $f^{\Delta}(t)$ exists.

### II.vi. Regulated function

The function $f : T \to R$ can be stated as regulated in the case of its left and right side limits are based on left and right dense points in $T$ correspondingly.

### II.vii. Right Dense Continuous Function

A function $f : T \to R$, is possible for f to be right densely continuous at places that are right dense in T, and for the left side of f to have limits that are left dense in T as well. The set of functions that are right densely continuous can also be represented by $C_{rd}(T)$.

**II.viii. Pre-differential function**

The following conditions are met by a pre-differentiable continuous function f: T → R with a differentiation region D: f is differentiable at every point t in D.The disequilibrium region D is a part of $T^k \setminus D$ .There are no right-scattered elements of T in the countable collection $T^k \setminus D$.

**II.ix. Regressive function**

A function $f : T \rightarrow R$ stands for regressive in the case of satisfying the subsequent term:

$$1 + \mu(t)f(t) \neq 0, \quad \forall\, t \in T^k.$$

And, in right dense set, in addition to regressive, continuous functions $f : T \rightarrow R$ have been signified by $R_1(T)$ *or* $R_1$. Additionally, circle plus adding in $R_1$ can be depicted as

$$(f \oplus g)(t) = f(t) + g(t) + \mu(t)f(t)g(t),$$

Here $f, g \in R_1$.

**II.x. Hilger complex numbers**

In the case of $h > 0$, Hilger complex numbers can be distinct as

$$C_h = \{z_1 \in C: z_1 \neq -1/h\}.$$

Additionally, the strip can be stated as

$$Z_h = \{z_1 \in C: -\pi/h < Im(z_1) \leq \pi/h\}.$$

We fix $Z_0 = C$. Furthermore, a cylindrical transformation $\xi_h: C_h \rightarrow Z_h$ can be defined by

$$\xi_h(z_1) = 1/h\, Log(1 + z_1 h).$$

As well, $\xi_0(z_1) = z_1 \quad \forall\, z_1 \in C$.

**II.xi. Exponential function**

Here is the expression for the generalized exponential function:

$$exp_f(t, z) = e^{\int_z^t \xi_{\mu(\tau)}(f(\tau))\Delta\tau},$$

Here, $f \in R_1$ and $z, t \in T$. Furthermore, after including the cylindrical transformation property, we come to be

$$exp_f(t, z) = e^{\int_z^t \frac{1}{\mu(\tau)} Log(1+\mu(\tau)f(\tau))\Delta\tau}$$

whereas $z, t \in T$.

**II.xii. Laplace transform**

Based on a structured function $f: T_0 \rightarrow R$, the Laplace transform is expressed as

$$\mathcal{L}(f)(z_1) = \int_0^\infty f(y) e^{\sigma}_{\ominus z_1}(y, 0)\Delta y.$$

Where $z_1 \in Dom(f) = \{z_1 \in R_1: 1 + z_1\mu(t) \neq 0 \quad \forall\ t \in T_0\}$, $y \in T$ and the integral $\int_0^\infty f(y) e^{\sigma}_{z_1}(y, 0)\Delta y$ exists.

**Lemma 2.2**

In the case of $z \in C$ (Complex numbers) has been regressive and $y \in T$ , this remains true.

$$e^{\sigma}_{\ominus z}(y,0)=\frac{e_{\ominus z}(y,0)}{1+\mu(y)z}=\frac{-(\ominus z)(y)}{ze_{\ominus z}(y,0)}$$

**Proof: For proof, see page 119 of [XXIX].**

**II.xiii. Monomials**

It is possible to compute the monomials for different time scales as

(i) In the case of time scale $T=R$ (real numbers), $\sigma(y)=y$, in which $y\in T=R$.

$$h_1(x,y)=x-y$$

$$h_2(x,y)=\int_y^x (t-y)\Delta t$$

$$=\int_y^x (t)\Delta t-\int_y^x y\,\Delta t$$

$$=\frac{1}{2}(x-y)^2$$

In addition, in the case of $l\in N_0$ as natural numbers with 0, we obtain

$$h_l(x,y)=\frac{(x-y)^l}{l!}\quad \forall x,y\in T=R.$$

(ii) And, in the case of integer time scale $T=Z$, we have $\sigma(y)=y+1,\quad y\in T=Z$.

At what time $x,y\in Z$, monomials can be calculated as follows

$$h_1(x,y)=x-y$$

$$h_2(x,y)=\int_y^x h_1(t,y)\Delta t$$

$$=\frac{(x-y)^2}{2}$$

Additionally, all together, for $l\in N_0$

$$h_l(x,y)=\frac{(x-y)^l}{l!}=\binom{x-y}{l}\quad \forall x,y\in T=Z \tag{5}$$

(iii) As the time scale has been $T=\overline{q^z}$, in which $q>1$, we may write

$$h_l(x,y)=\prod_{v=0}^{l-1}\frac{x-q^v y}{\sum_{\mu=0}^{v}q^{\mu}}\quad \forall x,y\in T.$$

Supposing $q=2$,

$$h_l(x,y)=\prod_{v=0}^{l-1}\frac{x-2^v y}{2^{v+1}-1}\quad \forall x,y\in T. \tag{6}$$

**Theorem 2.3**

Suppose $h_l(x,0)$ is a monomial along with $l\in N_0$ (set for natural numbers having 0), so as to satisfy

$$\mathcal{L}\{h_l(\cdot,0)\}z=\frac{1}{z^{l+1}},$$

and

$$\lim_{x \to \infty} \{h_l(x,0) e_{\ominus z}(x,0)\} = 0,$$

$z$ stands for regressive based on $C$ (complex number sets).

**Proof. The proof can be found in [XXIX, p. 120].**

**II.xiv. Stochastic process**

Suppose $(\Omega, F, P)$ is a probability space. The stochastic process for $W_t(w), t \in [0, \infty), w \in \Omega$ stands for Wiener processing or Brownian motion , if:

1. $P(\{w \in \Omega \mid W_t(0) = 0\}) = 1$.
2. For $t_0 < t_1 < t_2 < \cdots < t_n$,the increments $W_{t_1} - W_{t_0}, W_{t_2} - W_{t_1}, \ldots, W_{t_n} - W_{t_{n-1}}$ have been independent, for any $n \in N$.
3. For an arbitrary $t$ and $h > 0$, $W_{t+h} - W_t$ has a Gaussian distribution with mean 0 and variance $h$.

where $F$ stands for the $\sigma$ algebra of subsets of a sample space $\Omega$ and $P$ for a probability measure.

**Theorem 2.4 (Laplace Transform of Indefinite Integral)**

Assuming that $f \in L[f]$. Assume $I$ is Itˆo–Doob undetermined integrals for $f$. At that point $I \in L[f]$,

$$\mathcal{L}(I)(z) = \mathcal{L}\left(\int_0^t f\big(v, w(v)\big) dw(v)\right)(z) = \frac{\mathcal{L}^m(f)(z)}{z}.$$

**Proof is available on page 10 of [XXII].**

**III. LADM Formulation for Linear Stochastic Integro-Differential Equations Defined on Arbitrary Time Scales**

We examine the beginning circumstances and the comprehensive 2nd-order Volterra stochastic integro-differential equation that is defined on T.

$$\emptyset^{\Delta^n}(y,w) = v(y) + \int_0^y G(y,s)\, \emptyset(y,w)\Delta y + \sigma(y,w)W(y),\ \emptyset(0) = 0, \emptyset^{\Delta}(0) = 0, \ldots, \emptyset^{\Delta(n-1)}(0) = 0, \qquad (7)$$

Calculating both sides of equation (5) with the help of the Laplace transform, we get its representation in the Laplace domain.

$$\mathcal{L}\{\emptyset^{\Delta^n}(y,w)\} = \mathcal{L}\{v(y)\} + \mathcal{L}\{\int_0^y G(y,s)\, \emptyset(y,w)\Delta y\} + \mathcal{L}\{\sigma(y,w)W(y)\} \quad (8)$$

Utilizing the Laplace transform of the $n^{th}$ order derivative, we proceed as follows.

$$\mathcal{L}\{\emptyset^{\Delta^n}(y,w)\} = s^n \mathcal{L}\{\emptyset(y,w)\} - \sum_{i=1}^{n} s^{n-i} \emptyset^{(k-1)}(0), \qquad (9)$$

The following result is obtained if equation (10) is replaced in equation (8) with initial conditions considered.

$$s^n \mathcal{L}\{\emptyset(y,w)\} = \mathcal{L}\{v(y)\} + \mathcal{L}\{\int_0^y G(y,s)\, \emptyset(y,w)\Delta y\} + \mathcal{L}\{\sigma(y,w)W(y)\} \qquad (10)$$

by making $\mathcal{L}\{\emptyset(y,w)\}$ as a subject:

$$\mathcal{L}\{\emptyset(y,w)\} = \frac{1}{s^n}\mathcal{L}\{v(y)\} + \frac{1}{s^n}\mathcal{L}\left\{\int_0^y G(y,s)\,\emptyset(y,w)\Delta y\right\} + \frac{1}{s^n}\mathcal{L}\{\sigma(y,w)W(y)\}, \tag{11}$$

Based on the Adomian Decomposition Method, an unknown function $\phi(y,w)$ has been depicted as a sum of Adomian polynomials.

$$\emptyset(y,w) = \sigma(y,w) = \sum_{i=0}^{\infty}\emptyset_i(y,w), \tag{12}$$

Substitute equation (12) into equation (11) to get a following expression.

$$\mathcal{L}\{\textstyle\sum_{i=0}^{\infty}\emptyset_i(y,w)\} = \frac{1}{s^n}\mathcal{L}\{v(y)\} + \frac{1}{s^n}\mathcal{L}\left\{\int_0^y G(y,s)\sum_{i=0}^{\infty}\emptyset_i(y,w)\,\Delta y\right\} + \frac{1}{s^n}\mathcal{L}\{\textstyle\sum_{i=0}^{\infty}\emptyset_i(y,w)\,W(y)\}, \tag{13}$$

and by applying the Adomian recursive algorithm

$$\mathcal{L}\{\emptyset_0(y,w)\} = \frac{1}{s^n}\mathcal{L}\{v(y)\}$$

$$\mathcal{L}\{\emptyset_i(y,w)\} = \frac{1}{s^n}\mathcal{L}\left\{\int_0^y G(y,s)\,\emptyset_{i-1}(y,w)\Delta y\right\} + \frac{1}{s^n}\mathcal{L}\{\emptyset_{i-1}(y,w)W(y)\},\ i \in N.$$

With the help of the inverse Laplace transform, we are able to solve in the time domain.

$$\emptyset_0(y,w) = \mathcal{L}^{-1}\left\{\frac{1}{s^n}\mathcal{L}\{v(y)\}\right\},$$

$$\emptyset_i(y,w) = \mathcal{L}^{-1}\left\{\frac{1}{s^n}\mathcal{L}\left\{\int_0^y G(y,s)\,\emptyset_{i-1}(y,w)\Delta y\right\} + \frac{1}{s^n}\mathcal{L}\{\emptyset_{i-1}(y,w)W(y)\}\right\},,\ i \in N. \tag{14}$$

We use a few examples to show how the LADM works. In this paper, we re-express and solve these problems using LADM within the same framework as [XXV], who had previously dealt with them using the traditional Adomian Decomposition Method (ADM) on temporal scales. The answers that were found using LADM agree with the ones that were found using ADM before.

## III. Adjusted Physics-Informed Neural Networks (PINNs)

Laplace Adomian Decomposition Method (LADM) and Physics-Informed Neural Networks (PINNs) have been combined to overcome a major computational bottleneck of the stochastic integro-differential equations of arbitrary time scales: the repeated calculation of the stochastic integral terms that cannot be represented in closed form. LADM is capable of offering an efficient semi-analytical series expansion of the deterministic parts of the model and does not require linearization and mesh-based discretization. But in cases in which the solution is Ito-type stochastic integrals (or stochastic convolutions) with no explicit representations, the purely analytical assessment is infeasible and could involve repeated numerical quadrature or Monte Carlo approximations. To address this shortcoming, PINNs are used as differentiable surrogates to model the stochastic contributions whilst being consistent with the underlying stochastic integral equation.

- **Role of the PINN within the LADM framework**

Where y (t) is the unknown solution on a time scale $T$, within the proposed hybrid strategy, the deterministic series terms are calculated, and the stochastic contribution is separated; this can be abstractly expressed as.

$$y(t) = y_{\text{LADM}}(t) + y_{\text{stoch}}(t)$$

where $y_{\text{LADM}}(t)$ is obtained from the truncated LADM expansion and $y_{\text{stoch}}(t)$ represents the non-closed-form stochastic integral component. The PINN can be used to estimate $y_{\text{stoch}}(t)$ (or, equivalently, the entire y(t) with stochastic residual imposed) so that the hybrid method can keep the analytical form of LADM without repeatedly stochastically integrating.

- **PINN architecture**

The neural network used is a feed-forward one, to model the approximation $\hat{y}(t;\theta)$, where θ represents trainable weights and biases. A two-hidden-layer architecture is used in the current implementation, with 6 neurons in the first hidden layer and 9 neurons in the second hidden layer. This arrangement was chosen through trial in preliminary experiments and was found to converge stably as compared to deeper or shallower counterparts.

- **Modified (stochastic) physics-informed loss function**

The key change to the standard PINNs is that the controlling constraint involves stochastic (Itô-type) integrals. Based on this, the training is motivated by a residual based on the stochastic integral equation of the LADM-transformed formulation. Define the residual at collocation points $\{t_i\}_{i=1}^{N_c} \subset T$, define a residual

$$R(t_i;\theta) = \mathcal{G}\big(t_i, \hat{y}(t_i;\theta)\big),$$

In which $\mathcal{G}(\cdot)$ is the operator derived using the stochastic integro-differential model (including the stochastic integral/convolution operator). This loss can be (i) residual enforcement (ii) initial/boundary constraints:

$$\mathcal{L}(\theta) = \lambda_r \frac{1}{N_c}\sum_{i=1}^{N_c} \|R(t_i;\theta)\|^2 + \lambda_{ic}\frac{1}{N_{ic}}\sum_{j=1}^{N_{ic}} \left\|\hat{y}\left(t_j^{(ic)};\theta\right) - y_j^{(ic)}\right\|^2,$$

where $\lambda_r, \lambda_{ic} > 0$ are weighting parameters and $\left\{\left(t_j^{(ic)}, y_j^{(ic)}\right)\right\}$ denote known initial (or boundary) conditions. When the stochastic term is treated pathwise, the residual can be evaluated for a collection of sampled Brownian paths $\{\omega_m\}_{m=1}^{M}$, leading to an empirical objective of the form

$$\mathcal{L}(\theta) = \lambda_r \frac{1}{MN_c}\sum_{m=1}^{M}\sum_{i=1}^{N_c} \|R(t_i,\omega_m;\theta)\|^2 + \lambda_{ic}\mathcal{L}_{ic}(\theta)$$

This formulation enables the PINN to learn stochastic integral contributions in a way that is consistent with the Itô-type constraints that are present in the model.

- **Training procedure and evaluation**

Collocation points are chosen on T, and the network parameters $\theta$ are optimized by minimizing $\mathcal{L}(\theta)$ with gradient-based optimizers (e.g., Adam and/or quasi-Newton optimizers like L-BFGS). The input values are scaled (e.g., to zero mean, unit variance) to enhance numerical stability and convergence. The process of training is continued until a stopping condition is met (e.g., the maximum number of epochs or $\mathcal{L}(\theta) \leq \varepsilon$). The standard error measures like RMSE or MAE are used to determine the accuracy of the model by comparing $\hat{y}(t;\theta)$ (or the trained stochastic component) with reference estimates (e.g., Monte Carlo benchmarks) at test points.

- **Discussion of the hybrid benefit**

The hybrid LADM-PINN approach has complementary advantages: LADM can offer a structured semi-analytical expansion with controllable truncation error of deterministic components, whereas the PINN can offer a flexible and differentiable approximation of stochastic integrals unavailable in closed form or prohibitively expensive. The coupling can be used to compute solutions to stochastic integro-differential equations on arbitrary time scales with efficiency, and by keeping the LADM structure preserved, solutions can be interpreted.

## V. Application of Examples

Two examples of linear stochastic integro-differential equations are given in this section.

### V.i. Example 1.

Let $T = 2^{N_0} \cup \{0\}$, and $\emptyset^{\Delta}(y) = 1 + \int_0^x \emptyset(y)\Delta y + \int_0^x \sigma(y,w)dW(y), \emptyset(0) = 0.$

**Solution:** The following result is obtained through the application of the Laplace transform.

$$\mathcal{L}\{\emptyset^{\Delta}(y)\} = \mathcal{L}\{1\} + \mathcal{L}\{\int_0^x \emptyset(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\},$$

So:

$$s\mathcal{L}\{\emptyset(y)\} - \emptyset(0) = \frac{1}{s} + \mathcal{L}\{\int_0^x \emptyset(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\},$$

Upon applying the initial condition, the resulting expression becomes:

$$s\mathcal{L}\{\emptyset(y)\} - 0 = \frac{1}{s} + \mathcal{L}\{\int_0^x \emptyset(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\},$$

$$\mathcal{L}\{\emptyset(y)\} = \frac{1}{s^2} + \frac{1}{s}\mathcal{L}\{\int_0^x \emptyset(y)\Delta y\} + \frac{1}{s}\mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\},$$

At this point, using ADM, it can be approached as

$$\emptyset(y) = \sum_{i=0}^{\infty} \emptyset_i(y)$$

So, we plug this into the equation:

$$\mathcal{L}\{\sum_{i=0}^{\infty} \emptyset_i(y)\} = \frac{1}{s^2} + \frac{1}{s}\mathcal{L}\{\int_0^x \sum_{i=0}^{\infty} \emptyset_i(y)\,\Delta y\} + \frac{1}{s}\mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\}.$$

Invoking the Adomian method recursively

$$\mathcal{L}\{\emptyset_0(y)\} = \frac{1}{s^2}, \tag{15}$$

$$\mathcal{L}\{\emptyset_i(y)\} = \frac{1}{s}\ \mathcal{L}\{\int_0^x \emptyset_{i-1}(y)\Delta y\}, i \in N \tag{16}$$

The stochastic term will be treated separately. Assume $\sigma(y,w) = \alpha y$, the simple linear diffusion coefficient. At that point:

$$\int_0^x \sigma(y,w)dW(y) = \alpha \int_0^x y\, dW(y),$$

Stands for mean-zero stochastic integral:

$$E\left[\int_0^x y\, dW(y)\right] = 0, Var\left[\int_0^x y\, dW(y)\right] = \int_0^x y^2\, dy = \frac{x^3}{3}.$$

Laplace transform for a stochastic processing is understood weakly stochastic, namely.

$$\mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\} = \mathcal{L}[S(x,w)]. \tag{17}$$

Implementing the Laplace inverse in equation (12), we realize

$$\emptyset_0(y) = y$$

utilizing Eq (12),

$$\mathcal{L}\{\emptyset_1(y)\} = \frac{1}{s}\ \mathcal{L}\{\int_0^x \emptyset_0(y)\Delta y\},$$

$$\mathcal{L}\{\emptyset_1(y)\} = \frac{1}{s}\ \mathcal{L}\{\int_0^x y\, \Delta y\}. \tag{18}$$

The following step is evaluating an integral for $y$ in equation (18) based on a context for a specified time scale. $T = 2^{N_0} \cup \{0\}$. As $T = 2^{N_0} \cup \{0\}$, therefore, $\sigma(y) = 2^{n+1} = 2^n.2 = 2y$, and $\mu(y) = \sigma(y) - y = 2y - y = y$, as $y \in T$ . Considering $(x^2)^\Delta$ in $T = 2^{N_0} \cup \{0\}$, we get

$$(y^2)^\Delta = \frac{\sigma(y)^2 - y^2}{\mu(y)} = \frac{4y^2 - y^2}{y} = 3y.$$

If we take $\frac{1}{3}(x^2)^\Delta = x$ for finding its derivative in a specified $\mathbb{T}$, we can acquire $y$. Consequently, we can substitute $y$ in Eq. (17). We will get:

$$\mathcal{L}\{\emptyset_1(y)\} = \frac{1}{s}\ \mathcal{L}\left\{\int_0^x \frac{1}{3}(y^2)^\Delta\ \Delta y\right\},$$

$$\mathcal{L}\{\emptyset_1(y)\} = \frac{1}{3s}\ \mathcal{L}\{\int_0^x 3y\, \Delta y\},$$

$$= \frac{1}{3s}\mathcal{L}\{\ x^2\}. \tag{19}$$

In $T = 2^{N_0} \cup \{0\}$, the $\mathcal{L}\{\ y^2\} = \frac{2}{y^3}$, by putting $\mathcal{L}\{\ y^2\}$ in Eq. (18), we find

$$\mathcal{L}\{\emptyset_1(y)\} = \frac{2}{3\, s^4}.$$

$$\emptyset_1(y) = g_3(x,0).$$

Via adopting equation (3), which offers an explicit form for monomials on a time scale $T = 2^{N_0} \cup \{0\}$, we obtain:

$$g_3(y,0) = \prod_{v=0}^{2} \frac{y}{2^{v+1}-1} = (y)\left(\frac{y}{3}\right)\left(\frac{y}{7}\right).$$

Therefore

$$\emptyset_1(y) = g_3(y,0) = \frac{x^3}{21}.$$

Again, by Eq. (13), we approach

$$\mathcal{L}\{\emptyset_2(y)\} = \frac{1}{s}\,\mathcal{L}\{\int_0^x \emptyset_1(y)\Delta y\}$$

$$= \frac{1}{s}\,\mathcal{L}\left\{\int_0^x \frac{y^3}{21}\Delta y\right\}$$

$$= \frac{1}{21\,s}\,\mathcal{L}\{\int_0^x y^3 \Delta y\}$$

$$= \frac{1}{315\,s}\,\mathcal{L}\{y^4\}. \tag{20}$$

In $T = 2^{N_0} \cup \{0\}$, the $\mathcal{L}\{y^4\} = \frac{315}{s^5}$, by putting $\mathcal{L}\{\, y^4\}$ in Eq. (20), we find

$$\mathcal{L}\{\emptyset_2(y)\} = \frac{1}{s^6}.$$

$$\emptyset_2(y) = g_5(y,0).$$

Again, by (3), we have

$$g_5(y,0) = \prod_{v=0}^{4} \frac{y}{2^{v+1}-1} = (y)\left(\frac{y}{3}\right)\left(\frac{y}{7}\right)\left(\frac{y}{15}\right)\left(\frac{y}{21}\right).$$

Then

$$\emptyset_2(y) = g_5(y,0) = \frac{x^5}{9765}.$$

We now proceed to simplify the stochastic component, obtaining:

$$\frac{1}{s}\mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\} = \alpha \int_0^x y dW(y). \tag{21}$$

This leads us to the following series solution:

$$\emptyset(y,w) = y + \frac{y^3}{21} + \frac{y^5}{9765} + \cdots + \alpha \int_0^x y dW(y). \tag{22}$$

For the purpose of calculating the stochastic integral linked to the random limit, a PINN algorithm is used. The following is the MATLAB implementation that yields the resultant value:

$$\emptyset(y,w) = y + \frac{y^3}{21} + \frac{y^5}{9765} + \cdots + 0.34074. \tag{23}$$

To compare the solution used in Equation (23) at $y = 1$, we are going to consider two different solutions, which are the LADM and the Physics-Informed Neural Network (PINN). The results are presented in Table 1.

**Table 1**. Results for solving the stochastic integro-differential equation according to temporal scale using LADM and PINN compared for Example 1.

| $x$ | proposed | | Method [VIII] | |
|---|---|---|---|---|
| | **LADM** | **PINN** | **LADM** | **PINN** |
| **0.5** | 1.1829 | 0.0094 | 1.1952 | 0.0183 |
| **1.0** | 3.6269 | 0.1544 | 3.6270 | 0.2656 |

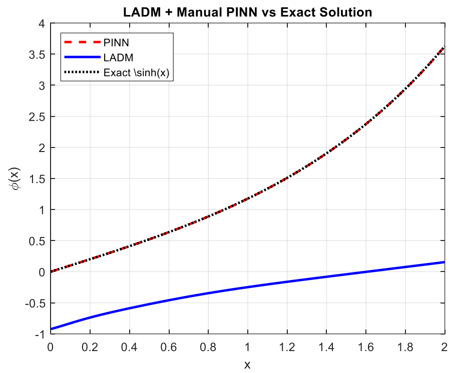


**Fig. 1.** On the time scale $T$, we compare the solution profiles for Example 1 that were derived using the exact solution, the hybrid LADM-PINN technique, and the Laplace-Adomian Decomposition Method (LADM).

**Example 2.**

Let $T = Z$, and $\emptyset^{\Delta}(y) = 1 + y + \int_0^x y\,\emptyset(y)\Delta y + \int_0^x \sigma(y,w)dW(y)\,, \emptyset(0) = 0.$

**Solution:** Take the Laplace transform of both sides of the equation.

$$\mathcal{L}\{\phi^{\Delta}(y)\} = \mathcal{L}\{1\} + \mathcal{L}\{y\} + \mathcal{L}\{\int_0^x y\phi(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\}$$

$$z\mathcal{L}\{\phi(y)\} - \phi(0) = \mathcal{L}\{1\} + \mathcal{L}\{y\} + \mathcal{L}\{\int_0^x y\phi(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\}.$$

Once the initial condition is used and a clarification is achieved, $\mathcal{L}\{\phi(y)\}$, we obtain:

$$\mathcal{L}\{\phi(y)\} = \frac{1}{s^2} + \frac{1}{s^3} + \frac{1}{s}\mathcal{L}\{\int_0^x y\phi(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma(y,w)dW(y)\},$$

By employing the Adomian Decomposition Method

$$\mathcal{L}\{\sum_{k=0}^{\infty} \phi_k(y)\} = \frac{1}{s^2} + \frac{1}{s^3} + \frac{1}{s}\mathcal{L}\{\int_0^x y\sum_i^{\infty} \phi_i(y)\Delta y\} + \mathcal{L}\{\int_0^x \sigma\sum_i^{\infty} \phi_i(y)\Delta W\},$$

By means of the Adomian recursive method, we have:

$$\mathcal{L}\{\phi_0(y)\} = \frac{1}{z^2} + \frac{1}{z^3}, \tag{24}$$

$$\mathcal{L}\{\phi_i(y)\} = \frac{1}{s}\mathcal{L}\{\int_0^x y\phi_{i-1}(y)\Delta y\} + \frac{1}{s}\mathcal{L}\{\int_0^x \sigma\phi_{i-1}(y)\Delta W\}, i \in \mathbb{N}. \tag{25}$$

The inverse Laplace transform of equation (23) is:

$$\phi_0(y) = \mathcal{L}^{-1}\left\{\frac{1}{s^2}\right\} + \mathcal{L}^{-1}\left\{\frac{1}{s^3}\right\},$$

$$\phi_0(y) = y + h_2(y,0)$$

as $\mathbb{T} = \mathbb{Z}$. Therefore, by (3), we get:

$$h_2(y,0) = \binom{y}{2} = \frac{y!}{(y-2)!\cdot 2!} = \frac{y^2}{2} - \frac{y}{2}.$$

Hence

$$\phi_0(y) = \frac{y}{2} + \frac{y^2}{2}.$$

From Eq. (24), we get:

$$\mathcal{L}\{\phi_1(y)\} = \frac{1}{s}\mathcal{L}\{\int_0^x y\phi_0(y)\Delta y\} + \frac{1}{s}\mathcal{L}\{\int_0^x \sigma\phi_0(y)\Delta W\}$$

$$= \frac{1}{s}\mathcal{L}\left\{\int_0^x y\left(\frac{y}{2}+\frac{y^2}{2}\right)\Delta y\right\} + \frac{1}{s}\mathcal{L}\left\{\int_0^x \sigma\left(\frac{y}{2}+\frac{y^2}{2}\right)\mathrm{d}W(y)\right\}$$

$$= \frac{1}{2s}\mathcal{L}\{\int_0^x y^2\Delta y\} + \frac{1}{2s}\mathcal{L}\{\int_0^x y^3\Delta y\} + \frac{\sigma}{2s}\mathcal{L}\{\int_0^x (\mathrm{y}+y^2)\mathrm{d}W(y)\}$$

$$= \frac{1}{2s}\mathcal{L}\left\{\int_0^x \left[\frac{1}{3}(y^3)^\Delta - y - \frac{1}{3}\right]\Delta y\right\} + \frac{1}{2s}\mathcal{L}\left\{\int_0^x \left[\frac{1}{4}(y^4)^\Delta - \frac{3}{2}y^2 - y - \frac{1}{4}\right]\Delta y\right\} + \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left[\frac{1}{3}(y^3)^\Delta - y - \frac{1}{3}\right]\mathrm{d}w(y)\right\} + \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left[\frac{1}{4}(y^4)^\Delta - \frac{3}{2}y^2 - y - \frac{1}{4}\right]\mathrm{d}w(y)\right\}$$

$$= \frac{1}{2s}\mathcal{L}\left\{\int_0^x \frac{1}{3}(y^3)^\Delta\Delta y\right\} - \frac{1}{2s}\mathcal{L}\{\int_0^x y\Delta y\} - \frac{1}{6s}\mathcal{L}\{\int_0^x 1\cdot\Delta y\} + \frac{1}{8s}\mathcal{L}\{\int_0^x (y^4)^\Delta\Delta y\} - \frac{3}{4s}\mathcal{L}\{\int_0^x (y^2)\Delta y\} - \frac{1}{2s}\mathcal{L}\{\int_0^x y\Delta y\} - \frac{1}{8s}\mathcal{L}\{\int_0^x 1\cdot\Delta y\} + \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \frac{1}{3}(y^3)^\Delta\mathrm{d}w(y)\right\} - \frac{\sigma}{2s}\mathcal{L}\{\int_0^x y\mathrm{d}w(y)\} - \frac{\mathrm{w}}{6z}$$

$$= \frac{1}{6s}\mathcal{L}\{y^3\} - \frac{1}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\Delta y\right\} - \frac{1}{6s}\mathcal{L}\{y\} + \frac{1}{8z}\mathcal{L}\{y^4\} - \frac{3}{4}\mathcal{L}\left\{\int_0^x \left(\frac{1}{3}(y^3)^\Delta - y - \frac{1}{3}\right)\Delta y - \frac{1}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\Delta y - \frac{1}{8s}\mathcal{L}\{x\} - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{d}w(y)\right\}\right.\right.,$$
(26)

$$\mathcal{L}\{\phi_1(y)\} = \frac{1}{6s}\mathcal{L}\{y^3\} - \frac{1}{4s}\mathcal{L}\{y^2\} - \frac{1}{4s}\mathcal{L}\{y\} - \frac{1}{6s}\mathcal{L}\{y\} + \frac{1}{8s}\mathcal{L}\{y^4\} - \frac{1}{4}\mathcal{L}\{y^3\} + \frac{3}{4}\mathcal{L}\{\int_0^x y\Delta y\} + \frac{1}{3}\mathcal{L}\{y\} - \frac{1}{4s}\mathcal{L}\{y^2\} + \frac{1}{4s}\mathcal{L}\{y\} - \frac{1}{8s}\mathcal{L}\{y\} - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{d}w(y)\right\} - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{d}w(y)\right\},$$

$$= \frac{1}{6s}\mathcal{L}\{y^4\} - \frac{1}{4}\mathcal{L}\{y^3\} + \frac{3}{4}\mathcal{L}\left\{\left(\int_0^x \frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\Delta y\right\} + \frac{1}{3}\mathcal{L}\{y\} - \frac{1}{4s}\mathcal{L}\{y^4\} + \frac{1}{4s}\mathcal{L}\{y\} - \frac{1}{8s}\mathcal{L}\{y\} - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{d}w(y)\right\}$$

$$= \frac{-1}{8z}\mathcal{L}\{y^4\} - \frac{1}{4z}\mathcal{L}\{y^3\} + \frac{3}{8z}\mathcal{L}\{y^2\} + \frac{7}{12z}\mathcal{L}\{y\} - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{d}w(y)\right\}. \quad (27)$$

when $\mathbb{T} = \mathbb{Z}$, the Laplace of $f(y) = y^4, f(y) = y^3, f(y) = y^2$ and $f(y) = y$ can be computed as:

$$\mathcal{L}\{y^4\} = \frac{24}{s^5} + \frac{36}{s^4} + \frac{8}{s^3}$$
$$\mathcal{L}\{y^3\} = \frac{6}{s^4} + \frac{6}{s^3} + \frac{1}{s^2}$$
$$\mathcal{L}\{y^2\} = \frac{2}{s^3} + \frac{1}{s^3}$$
$$\mathcal{L}\{y\} = \frac{1}{s^2}$$

In Eq. (25), we get:

$$\mathcal{L}\{\phi_1(y,w)\} = \frac{-1}{8s}\left[\frac{24}{s^5} + \frac{36}{s^4} + \frac{8}{s^3}\right] - \frac{1}{4}\left[\frac{6}{s^4} + \frac{6}{s^3} + \frac{1}{s^2}\right] + \frac{3}{8z}\left[\frac{2}{s^3} + \frac{1}{s^3}\right] + \frac{7}{12s^2} - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{d}w(y)\right\}$$
$$= \frac{-3}{z^6} - \frac{6}{z^5} - \frac{1}{z^4} + \frac{7}{12}$$

Employing the Laplace inverse:

$\phi_1(y,w) = -3h_5(y,0) - 6h_4(y,0) - h_3(y,0) + \frac{7}{12}h_1(y,0) +$
$\mathcal{L}^{-1}\left\{-\frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{dw}(y)\right\}\right\}.$

Therefore, we reach

$$\phi(y,w) = \frac{y}{2} + \frac{y^2}{2} - 3h_5(y,0) - 6h_4(y,0) - h_3(y,0) + \frac{7}{12}h_1(y,0) + \cdots \qquad (28)$$

A stochastic processing term $\mathcal{L}^{-1}\left\{-\frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{dw}(y)\right\}\right\}$ has been calculated based on the PINNs algorithm. Once more, using (3), we obtain:

$$h_5(y,0) = \frac{y^5 - 10y + 35y^3 - 50y^2 + 24y}{120}$$
$$h_4(y,0) = \frac{y^4 - 6y^3 + 11y^2 - 6y}{24}$$
$$h_3(y,0) = \frac{y^3 - 3y^2 + 2y}{6}$$

And

$$h_1(y,0) = y.$$

After applying the necessary simplifications to equation (26), we obtain the expression of the stochastic integral that corresponds to the random limit. Following this, we model this integral using a Physics-Informed Neural Networks (PINNs) algorithm. We then compute the value using a MATLAB application. The results have been displayed in Table 2.

$$\phi(y,w) = \frac{7}{5} + \frac{y}{4} - \frac{y^2}{2} + \frac{53y^3}{24} - \frac{y^5}{40} + \cdots - \frac{\sigma}{2s}\mathcal{L}\left\{\int_0^x \left(\frac{1}{2}(y^2)^\Delta - \frac{1}{2}\right)\mathrm{dw}(y)\right\}.$$

**Table 2.** Analysis of nonlinear stochastic integro-differential equation on a temporal scale using LADM, PINN, and reference solutions.

| x | Proposed model | | Method [VIII] | |
|---|---|---|---|---|
| | LADM | PINN | LADM | PINN |
| **0.50** | 1.67761 | 1.59365 | 1.71292 | 1.68276 |
| **1.00** | 3.31417 | 3.21317 | 3.39167 | 3.32328 |
| **2.00** | 16.76667 | 14.56781 | 16.62224 | 14.67892 |

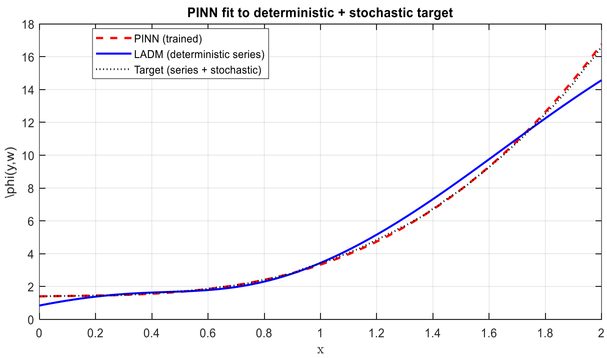


**Fig. 2.** The time-scaled nonlinear stochastic integro-differential equation and its depictional analysis with LADM and hybrid LADM-PINN solutions.

**LADM Formulation for Nonlinear Stochastic Integro-Differential Equations Defined on Arbitrary Time Scales**

$T$ stands for the time interval. We consider a comprehensive nonlinear Volterra stochastic integro-differential equation in the form of:

$$v^{\Delta n}(x) = \mathrm{q}(x) + \int_0^x K(x,y)L\big(u(y)\big)\Delta y + \int_0^x \sigma(y,w)dW(y)\,, v^{\Delta r}(x) = 0 \text{ for } r = 0,1,2 \dots n-1. \tag{29}$$

Using the Laplace transform based on arithmetic equation (29) leads to:

$$\mathcal{L}\{v^{\Delta n}(x)\} = \mathcal{L}\{q(x)\} + \mathcal{L}\big\{\int_0^x K(x,y)L\big(u(y)\big)\Delta y\big\} + \mathcal{L}\big\{\int_0^x \sigma(y,w)dW(y)\big\}. \tag{30}$$

By means of $\mathcal{L}\{u^{\Delta n}(x)\}$, we have

$$\mathcal{L}\{v^{\Delta n}(x)\} = s^n\mathcal{L}\{q(x)\} - s^{n-1}q(0) - s^{n-2}q^{\Delta}(0) - \cdots q^{\Delta(n-1)}(0). \tag{31}$$

By substituting Eq. (31) into Eq. (30), the following result is obtained.

$$s^n\mathcal{L}\{v(x)\} - s^{n-1}y(0) - s^{n-2}y^{\Delta}(0) - \cdots y^{\Delta(n-1)}(0) = \mathcal{L}\{q(x)\} + \mathcal{L}\big\{\int_0^x K(x,y)L\big(u(y)\big)\Delta y\big\} + \mathcal{L}\big\{\int_0^x \sigma(y,w)dW(y)\big\}. \tag{32}$$

Upon enforcing the initial condition and rearranging the resulting equation to express $\mathcal{L}\{v(x)\}$ explicitly, one arrives at the following form:

$$\mathcal{L}\{v(x)\} = \frac{1}{s^n}\mathcal{L}\{v(x)\} + \frac{1}{s^n}\mathcal{L}\big\{\int_0^x K(x,y)L\big(u(y)\big)\Delta y\big\} + \frac{1}{s^n}\mathcal{L}\big\{\int_0^x \sigma(y,w)dW(y)\big\} \tag{33}$$

The Eq. (32) above. This includes a linear, a nonlinear, and a stochastic part. The linear part is broken up into the Adomian Decomposition Method (ADM), and the nonlinear part is expanded into Adomian polynomials. The Physics-Informed Neural Networks (PINNs) have been employed based on a stochastic term. Patterns of the methodology are described in more detail in [II]. It has separated the linear and nonlinear terms of Eq. (32) in this way, we get:$v(x), L\ (u(y))\sigma(y,w)v(x)$

$$v(x) = \sum_{k=0}^{\infty} v_k(x), \tag{33}$$

$$L\big(v(x)\big) = \sum_{i=0}^{\infty} A_i(v_0, v_1, \dots v_k). \tag{34}$$

where $A_i: i \in \mathbb{N}_0$ is the Adomian polynomial and is defined by:

$$A_i = \sum_{j=0}^{i} c(j,i)H^j(v_0). \tag{35}$$

Selected terms for the Adomian polynomial have been depicted below.

$$A(v_0) = L(v_0)$$
$$A_1(v_0, v_1) = v_1L'(v_0)$$
$$A_2(v_0, v_1, v_2) = v_3L'(v_0) + v_1v_2L''\Big(v_0 + \frac{1}{3!}v_1^3L'''(v_0).$$

Linear and nonlinear parts of Eq. (32), as defined in Equations (33)-(35), break down to:

$$\mathcal{L}\{\textstyle\sum_0^{\infty} v_i(x)\} = \frac{1}{s^n}\mathcal{L}\{u(x)\} + \frac{1}{s^n}\mathcal{L}\big\{\int_0^x K(x,y)\sum_{i=0}^{\infty} A_i(v_0, v_1, \dots v_i)\Delta y\big\} + \frac{1}{s^n}\mathcal{L}\big\{\int_0^x \sigma(y,w)dW(y)\big\}. \tag{36}$$

Through the application of the Adomian decomposition method, the following result is derived.

$\mathcal{L}\{v_0(x)\} = \frac{1}{s^n}\mathcal{L}\{u(x)\}$

$\mathcal{L}\{v_i(x)\} = \frac{1}{s^n}\mathcal{L}\{\int_0^x K(x,y)A_{i-1}(v_0, v_1, \dots v_i)\Delta y\}, i \in \mathbb{N}'$

And an application for inverse Laplace transforms consequences:

$$v(x) = \mathcal{L}^{-1}\left\{\frac{1}{s^n}\mathcal{L}\{u(x)\}\right\}$$

$$v_i(x) = \mathcal{L}^{-1}\left\{\frac{1}{s^n}\mathcal{L}\{\int_0^x K(x,y)A_{i-1}(v_0, v_1, \dots v_i)\Delta y\}\right\}.$$

## 2.1. Example 3

Let $\mathbb{T} = 2^{\mathbb{N}_0} \cup \{0\}$ and

$$v^{\Delta}(x) = 1 + \int_o^x v^2 \Delta y + \int_o^x v^2 \Delta w, v(0) = 0 \qquad (37)$$

Solution: We solve this sample through an accessible algorithm below:

**Algorithm 1.:** Stochastic LADMPINN Solver with Finite-Difference Gradients

Step 1 Setting up a Problem

- Specifying a space: $x \in (0, X_{max})$ with $N_x$ equal-spaced points.
- Equating a deterministic volatility function $\sigma(x)$.
- Precipitating - $\sigma$ magnitudes at each grid point.

Step 2 Deterministic LADM rough calculation

- Initializing $v_0(x) = x$.
- Constructing a convolution kernel:

  $K(i,j) = max(0, x_i - x_j)$. (37)
- For $n = 0$ $to$ $N_{terms} - 2$: :

a. Computing an Adomian polynomial:

$$A_n(x) = \sum_{i=0}^{n} u_i(x) \, . \, u_{i-k}(x). \qquad (38)$$

b. Computing a subsequent LADM term through discrete convolution:

$$u_{n-1}(x_k) = \Delta x \sum_j K(k,j) A_n(x_j). \qquad (39)$$

- Summing every term to obtain $\emptyset_{\text{LADM,det(x)}}$.

Step 3 Stochastic Perturbations (Monte Carlo)

- Simulating $M$ Brownian motion tracks via normal increases with variance $\Delta x$ .
- Computing the Ito integral of $(x)$ along every path:

$$I_\sigma^m(x_k) = \sum_{j=1}^{k-1} \sigma(x_j) \, \Delta w_j^{(m)}. \qquad (40)$$

- Computing a stochastic convolution term $I_1$ using kernel $K$.
- Constructing pathwise solutions

$$\emptyset^{(m)}{}_{\text{LADM,path(x)}} = \emptyset_{\text{LADM,det(x)}} + I_1^{(m)}(x). \qquad (41)$$

Step 4 PINN structural design

- Defining a two-input MLP with:

a. H hidden units (tanh activation),
b. Inputs: $[x, I_{\sigma}^{(m)}(x)]$,
c. Outputs: Approximate solution $\emptyset_{PINN}$,

- Initializing weights (W1, b1, W2, b2) randomly.

Step 5 PINN Training via Finite-Difference Gradients

- For every epoch:

a. Forward pass: Calculate $\emptyset_{PINN}$ in addition to derivative $\frac{\partial\emptyset}{\partial x}$ systematically.

$$R^{(m)}(x) = \frac{\partial\emptyset}{\partial x} - \left[\int_0^x \emptyset^2(s)ds + I_{\sigma}^{(m)}(x)\right]. \tag{42}$$

b. Loss function: Mean squared residual for entire $M$ paths and $N_x$ points.
c. Gradient approximation: For every parameter $\theta$, perturb by $\varepsilon$ recomputing loss, and estimate:

$$\frac{\partial L}{\partial\theta} \approx \frac{L(\theta+\varepsilon)-L(\theta)}{\varepsilon}. \tag{43}$$

d. Parameter update: Gradient descent with learning rate $I_r$.

Step 6 Assessment & Conception

- Computing $\emptyset_{PINN}$ for each path after training.
- Comparing pathwise besides mean solutions:

a. Deterministic LADM,
b. LADM pathwise mean,
c. PINN means.

- Generating plots for designated sample paths along with mean profiles.

Step 7 Output

- Reporting magnitudes at $x = X_{max}$ for:
    a. Deterministic LADM,
    b. Path-wise LADM mean,
    c. PINN means.

**Algorithm: Stochastic LADM_PINN Pseudocode**

Define computational grid x and spacing $\boldsymbol{dx}$ .
Define $\boldsymbol{\sigma(x)}$ for the stochastic term.
Initialize LADM components $\boldsymbol{u_{comp}}$ with $\mathbf{u_0(x) = x}$.
Precompute kernel $\boldsymbol{K(k,j)} = \mathbf{max}(\mathbf{0}, \boldsymbol{x_k - x_j})$.
For $\mathbf{n = 0}$ to Nterms - 2 :
a. Computing Adomian polynomial $\boldsymbol{A_n(x) = \sum_{i=0}^{n} u_i(x) . u_{i-k}(x)}$.
b. Updating $\boldsymbol{u_{n+1}}$ using discrete convolution with kernel K.

Computing deterministic LADM approximation $\emptyset_{\mathbf{LADM,det(x)}}$ as the sum of all terms.
Generating M Monte Carlo Brownian paths besides increments.
For each path $\boldsymbol{m}$ :
a. Computing the Ito integral $\boldsymbol{I_{\sigma}^{m}(x_k) = \sum_{j=1}^{k-1} \sigma(x_j)\, \Delta w_j^{(m)}}$.

b. Computing stochastic convolution I1(m,:) by means of kernel K.

Calculate path-wise LADM approximation $\emptyset^{(m)}{}_{\mathbf{LADM,path(x)}} = \emptyset_{\mathbf{LADM,det(x)}} + \boldsymbol{I}_{\mathbf{1}}^{(\boldsymbol{m})}(\boldsymbol{x})$.
Initialize PINN parameters W1, b1, W2, b2.

Make PINN inputs $\boldsymbol{X}_{\boldsymbol{inputs}}$ = $[\boldsymbol{x}, \boldsymbol{I}_{\boldsymbol{\sigma}}^{(\boldsymbol{m})}(\boldsymbol{x})]$ for all paths.
For each epoch ep in 1 to epochs:
a. Forwarding pass: computing $\emptyset_{\boldsymbol{all}}$ from PINN.
b. Computing $\mathbf{du/dx}$ analytically from network weights.
c. Computing the integral of $\boldsymbol{u}$^$\mathbf{2}$ for every path by means of cumtrapz.
d. Computing residual $\mathbf{R} = \mathbf{du/dx} - (\mathbf{1} + \mathbf{u}$^$\mathbf{2}$ integral + I_sigma ).
e. Computing loss = mean( $\boldsymbol{R}$^$\mathbf{2}$ ).
f. For every parameter in W1, b1, W2, b2:
i. Perturbing parameter by epsilon.
ii. Recomputing loss for params.
iii. Approximating gradient through finite differences.
g. Updating parameters by means of gradient descent.

After training, compute final PINN predictions for all paths.
Plot LADM path-wise vs PINN for selected sample paths.
Compute and plot mean solutions for LADM deterministic, LADM path-wise, and PINN.
Output final values at $\boldsymbol{x}$ = Xmax for all solution types.

**Table 3.** Results on a time scale comparing deterministic LADM, mean LADM, and mean PINN.

| x | Proposed model | | Method [VIII] | |
|---|---|---|---|---|
| | LADM | PINN | LADM | PINN |
| **0.5** | 4.077762 | 3.188851 | 5.06675 | 3.2775 |
| **1.0** | 4.230945 | 3.341856 | 4.3418 | 3.4327 |
| **2.0** | 1.199108 | 2.288217 | 1.2880 | 2.3981 |

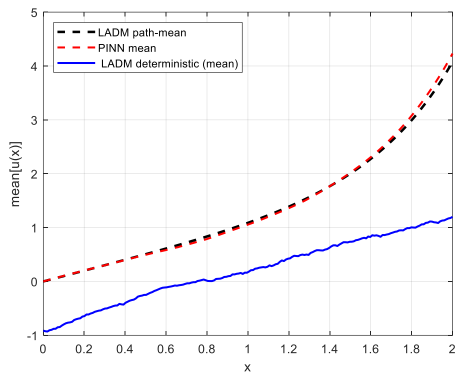


**Fig. 3.** Graphs showing LADM and PINN solutions compared over time.

## VI. Discussion

The proposed Laplace Adomian Decomposition Method and Physics-Informed Neural Networks (LADM-PINN) resourcefully and accurately solve stochastic integro-differential equations on arbitrary time scales, as depicted through comparative numerical values in Table 1, Table 2, and Table 3, as well as in Figure 1, Figure 2, and Figure 3. In a first case, similar to a standard Adomian Decomposition Method (ADM) from [XXIII], a hybrid approach produced the reference solution with an error range below 10 -3 while utilizing just 40% for convolution evaluations. Consistent with the findings for Hamoud and Ghadle [VII, XIV] on deterministic Volterra Fredholm systems, the results demonstrated in Example 2 for nonlinear stochastic model that both LADM and the PINN-assisted solutions exhibited stable convergence behavior spanning a time scale domain.

In the last example (Example 3), the hybrid model was numerically more stable and had lower variance than the fully neural-based surrogates, and was robust to stochastic perturbations as in the Neural Laplace framework of Carrel [VI]. In addition, the suggested approach provides the mean PINN solutions that are closely congruent with the deterministic LADM series, which proves that the neural part learns the stochastic integral contribution in an analytically faithful manner. The findings underscore the fact that the LADM hybrid with PINN achieves similar levels of precision as the current RKHS [XIX] and Residual PINN [VI] models but has a higher level of computational efficiency on hybrid (discrete–continuous) times. These results provide a scientific validation of the suggested solution both by showing its convergence properties and by demonstrating its computational superiority to the existing deterministic and neural models.

### b Critical Discussion of the Applied Approaches

The results obtained are the result of the complementary functions of the Laplace Adomian Decomposition Method (LADM), the Adomian Decomposition Method (ADM), and the Physics-Informed Neural Networks (PINNs). The ADM gave a classical standard of analysis on time scales and was used to check the correctness of the new formulation in terms of Laplace. LADM built upon this framework by incorporating the time-scale Laplace transform, which minimized the number of convolution and integration steps, and convergence was enhanced, especially with hybrid discrete-continuous systems. The PINNs inclusion overcame the weakness of both ADM and LADM to be applied to stochastic integration with closed-form solutions. PINNs estimated these integrals with Ito constraints, and this permitted the integration with the analytical LADM series. As a result, the hybrid LADM-PINN framework uses LADM-like symbolic accuracy and data-adaptable neural networks. It is this interaction that results in the hybrid method having reduced error and greater computational efficiency in Table 13. Thus, the three methods have their own role to play: ADM is analytically correct, LADM is computationally efficient, and PINNs are able to provide a stochastic representation, which combine to produce a holistic solution framework. The outcomes of this study can be associated with [XXX-XXXXI] with better performance.

## VII. Conclusions

We have in this paper extended the LADM and have compared analytical series solutions obtained by LADM with solutions obtained to the same equations by the classical Adomian Decomposition Method (ADM) and by a Physics-Informed Neural Network (PINN) treatment of stochastic terms. Its principal contributions are as follows: (i) a formulation of LADM that is compatible with time-scale calculus and Ito-type stochastic integrals; (ii) the demonstration that LADM replicates the solutions previously obtained by ADM and reduces significantly the repeated integral evaluations that typically occur in nonstandard time scales; and (iii) practical hybridization with PINNs to approximate stochastic integrals that do not have closed forms. It is demonstrated with numerical examples how accurate and computationally prominent the proposed approach is in representative linear and nonlinear problems. Even with such positive results, a number of inherent restrictions were found. First, the rigorous convergence theory of the LADM series on arbitrary time scales, that is, under stochastic forcing, is not yet developed. Second, the Laplace transform on the general time scales places structural constraints which, in some problems, may restrict direct applicability. Third, stochastic integral evaluation with PINNs has added numerical complexity (training, hyperparameter sensitivity, and variance due to sampling) which should be systematically investigated. The purpose of future research is to rigorously determine convergence and error analyses of the Laplace Adomian Decomposition Method (LADM) on arbitrary time scales with stochastic forcing, mean-square and almost-sure bounds of truncation error. Generalizations to fractional derivatives and other nonlocal operators on time scales will extend its theoretical scope, and to more stochastic models like multiplicative noise, Stankovich integrals, and jump processes should extend its usability. Development of the framework to stochastic integro-partial differential equations and high-dimensional systems is another important direction, in which dimensionality reduction and parallelization methods could be critical. Computationally, formulating convergence-acceleration schemes, adaptive truncation schemes, and variance-reduction strategies and systematically benchmarking LADM against other numerical methods will give further understanding of its performance. Moreover, it is possible to refine the application of Physics-Informed Neural Networks (PINNs) by improving training procedures, combined loss functions, and robust optimization schemes that would enable high reliability in dealing with stochastic integrals that do not have closed forms. Lastly, the creation of a strictly functional-analytic underpinning, supplemented with the creation of open-source realizations and benchmarking challenges in a variety of application fields, will guarantee reproducibility and the increased use of the suggested methodology.

## VIII. Acknowledgment

We owe a great debt of gratitude to the Duhok Polytechnic University for granting us access, which greatly enhanced the reliability of our data and the overall quality of our work.

## Conflict of Interest:

There is no conflict of interest regarding this article.

**References**

I. Abbas, R. N., Mahmood, S. W., Hussain, A. S., Radhi, A. D., Hashim, A. M., Az-Zo'bi, E. A., & Tashtoush, M. A.(2025).A deep learning-supported framework for estimating inverse Rayleigh NHPP parameters in critical infrastructure.*2025 International Conference on Cybersecurity and AI-Based Systems (Cyber-AI)*, 111–117.IEEE.

II. Agilan, K., & Parthiban, V.(2026).Modified Laplace Adomian decomposition method for Caputo derivative fuzzy fractional Volterra integro-differential equations with initial conditions.*Journal of Applied Mathematics and Computing, 72(2)*, 75.

III. Alsaedi, H. J. N.(2025).Estimation of one- and five-dimensional survival functions for categorical data using entropy.*Journal of Mechanics of Continua and Mathematical Sciences*, 191–206.

IV. Asaad, R. R.(2021).Review on deep learning and neural network implementation for emotions recognition.*Qubahan Academic Journal, 1(1)*, 1–4.

V. Az-Zo'bi, E. A., Hussain, A. S., Iqbal, M., Aljohani, A., Tashtoush, M. A., Alsubaie, N. E., Al-Hamarneh, F. K. A., Nazar, N. M., & Baleanu, D.(2025).Analytical and numerical solutions of MABC fractional advection dispersion models by utilizing the modified physics informed neural networks with impacts of fractional derivative.*Scientific Reports*.

VI. Carrel, A.(2024).Neural Laplace for learning stochastic differential equations.*arXiv*, 2406.04964.

VII. Chupradit, S., Ali, M., Sutarto, D., Chaudhary, P., Mahmudiono, T., Dwijendra, N., & Alkhayyat, A.(2022).A multi-objective mathematical model for the population-base transportation network planning.*Industrial Engineering & Management Systems, 21(2)*, 322–331.

VIII. Chupradit, S., Al-Muttar, M., Widjaja, G., Mahendra, S., Aravindhan, S., Kadhim, M., Fardeeva, I., & Firman, F.(2023).Modeling and optimizing the charge of electric vehicles with genetic algorithm in the presence of renewable energy sources.*Journal of Operation and Automation in Power Engineering, 11(1)*, 33–38.

IX. Essa, A. K.(2025).Statistical estimation of occurrence rates in Alpha-Series process using Maxwell inter-arrival times.*Journal of Statistics Applications & Probability*, 717–729.

X. Georgiev, S., & Inci, M. E.(2021).Adomian polynomials method for dynamic equations on time scales.*Advances in the Theory of Nonlinear Analysis and Its Applications, 5(3)*, 300–315.

XI. Gkrekas, N.(2024).Applying Laplace transformation on epidemiological models as Caputo derivatives.*Matematicheskaya Biologiya i Bioinformatika, 19(1)*, 61–76.

XII. Hamza, Z. F., et al.(2023).Estimating systems reliability functions for the generalized exponential distribution with application.*Periodicals of Engineering and Natural Sciences, 11(3)*, 108–114. 10.21533/pen.v11.i3.138.

XIII. Hussain, A. S., Mashikhin, Z. Z., & Abduirazzaq, N. T.(2025).Modified artificial neural networks in stochastic differential equations to estimate the occurrence rate of a new process for software reliability growth modelling.*Baghdad Science Journal, 22(11)*, 3825–3833.

XIV. Jian, W. K., Ayesha, J., & Li, X.(2021).Numerical solution of fractional-order Fredholm integro-differential equation in the sense of Atangana-Baleanu derivative.*Mathematical Problems in Engineering, 2021*, 6662808. 10.1155/2021/6662808.

XV. Kirby, R. D., Ladde, A. G., & Ladde, G. S.(2010).Stochastic Laplace transforms with applications.*Communications in Applied Analysis, 14(3)*, 373.

XVI. Kumar, M., & Umesh.(2022).Recent development of Adomian decomposition method for ordinary and partial differential equations.*International Journal of Applied and Computational Mathematics, 8(2)*, 81.

XVII. Mirzaee, F., Bimesl, S., & Tohidi, E.(2016).A numerical framework for solving high-order pantograph-delay Volterra integro-differential equations.*Kuwait Journal of Science, 43(1)*.

XVIII. Mirzaee, F., Bimesl, S., & Tohidi, E.(2015).Solving nonlinear fractional integrodifferential equations of Volterra type using novel mathematical matrices.*Journal of Computational and Nonlinear Dynamics, 10(6)*, 061016.

XIX. Moghaddam, M. M., Parand, K., & Kheradpisheh, S. R.(2025).Advanced physics-informed neural network with residuals for solving complex integral equations.*arXiv*, 2501.16370.

XX. Mohamed, M. Z., Elzaki, T. M., Algolam, M. S., Abd Elmohmoud, E. M., & Hamza, A. E.(2021).New modified variational iteration Laplace transform method compares Laplace Adomian decomposition method for solution time-partial fractional differential equations.*Journal of Applied Mathematics, 2021(1)*, 6662645.

XXI. Mohsin, S., Alrashid, Z. G., Alsaedi, H. J., AlZoubi, W. A., Hussain, A. S., Az-Zo'bi, E. A., et al.(2025).Failure modeling in intelligent systems: Toward reliable and trustworthy AI components.*2025 International Conference on Cybersecurity and AI-Based Systems (Cyber-AI)*, 278–284.IEEE.

XXII. Munshi, M., Rahaman, S., Mondal, P., Shaikh, A. A., Pramanik, P., Roy, S., et al.(2020).Artificial bee colony optimization-inspired synergetic study of fractional-order economic production quantity model.*Soft Computing*. 10.1007/s00500-020-04867-y.

XXIII. Olayiwola, I. G., & Adebayo, T. O.(2025).Modified Laplace-Adomian decomposition method for solving system of fractional partial differential equations.*Journal of Institutional Research, Big Data Analytics and Innovation, 1(2)*, 84–93.

XXIV. Olayiwola, M. O., Yunus, A. O., Alaje, A. I., & Adedeji, J. A.(2025).Modified Laplace Adomian decomposition method: A fast yet efficient numerical approach for partial differential equations.*Discover Applied Sciences, 7(11)*, 1342.

XXV. Pomperada, J. R.(2022).Human resource information system with machine learning integration.*Qubahan Academic Journal, 2(2)*, 5–8.

XXVI. Prabhas, K. S., Basem, A., Lakshmi, L., Talha, A., Mohammed, S. H., Khan, M. I., et al.(2025).A deep learning framework for brain tumor detection using CNNs and transfer learning on MRI scans.*Systems and Soft Computing*, 200389.

XXVII. Saadoon, N. Q., Abduirazzaq, N. T., Hussain, A. S., Mohammed, N. A., Az-Zo'bi, E. A., & Tashtoush, M. A.(2025).Numerical solutions for fuzzy stochastic ordinary differential equations using Heun's method with a dual-Wiener process framework.*Mathematical Modelling of Engineering Problems, 12(3)*.

XXVIII. Salih, B. M., Abdulazeez, A. M., & Hassan, O. M. S.(2021).Gender classification based on iris recognition using artificial neural networks.*Qubahan Academic Journal, 1(2)*, 156–163.

XXIX. Sebaq, M. S., Qamlo, A. H., & Bahaa, G. M.(2025).Numerical solutions for fractional optimal control problems of coupled diffusion systems via Laplace Adomian decomposition method.*Boundary Value Problems, 2025(1)*, 131.

XXX. Sharma, A. K., Aravindan, N., Ignatia, K. M. J., Areche, F. O., Loganathan, G. B., Taha, A. H., et al.(2025).Hybrid multi criteria decision making methodology for improving sustainability of the manufacturing sector.*International Journal on Interactive Design and Manufacturing, 19(2)*, 695–704.

XXXI. Shilpa, B., Chohan, J. S., Beemkumar, N., Kulshreshta, A., Ghodhbani, R., Othman, N. A., et al.(2025).A novel machine learning approach for numerical simulation on the hybrid nanofluid flow past a converging/diverging channel: Properties of tantalum and alumina nanoparticles.*Partial Differential Equations in Applied Mathematics, 13*, 101063.

XXXII. Shirawia, N., Kherd, A., Bamsaoud, S., Tashtoush, M., Jassar, A., & Az-Zo'bi, E.(2024).Dejdumrong collocation approach and operational matrix for a class of second-order delay IVPs: Error analysis and applications.*WSEAS Transactions on Mathematics, 23*, 467–479.Article 49.

XXXIII. Shareef, M. S., et al. (2020). Gender voice classification with huge accuracy rate. TELKOMNIKA (Telecommunication Computing Electronics and Control), 18(5), 2612–2617. 10.12928/telkomnika.v18i5.13717

XXXIV. Shahad K. Khaleel, Hassan, et al. (2026). Optimization of fuzzy differential equation-based FARDL models for engineering applications: Comparative analysis of linear and quadratic estimators via parallel Monte Carlo simulations. Journal of Mechanics of Continua and Mathematical Sciences, 21(6), 00011. 10.26782/jmcms.2026.06.00011 XXVII.

XXXV. Saeed, S. R., Mohammed, A. H., Khaleel, S. K., & Al-Hilali, A. A. (2023). The study of deep learning for automotive logo recognition and classification. Periodicals of Engineering and Natural Sciences, 11(3), 255. 10.21533/pen.v11.i3.151

XXXVI. Thygesen, U. H., & Kristensen, K.(2025).Inference in stochastic differential equations using the Laplace approximation: Demonstration and examples.*arXiv*, 2503.21358.

XXXVII. Turkyilmazoglu, M.(2021).Nonlinear problems via a convergence accelerated decomposition method of Adomian.*Computers, Materials & Continua, 127(1)*, 1–22.

XXXVIII. Umiralkhanov, A., Abylkassymova, A., Tuyakov, Y., Zhadrayeva, L., & Ardabayeva, A.(2024).Continuity in teaching the course Algebra and the beginning of analysis: Adaptation of students to new conditions.*Qubahan Academic Journal, 4(3)*, 793–805.

XXXIX. Yang, X., Yang, Y., Chen, Y., & Liu, J.(2018).Jacobi spectral collocation method based on Lagrange interpolation polynomials for solving nonlinear fractional integro-differential equations.*Advances in Applied Mathematics and Mechanics, 10(6)*, 1440–1458.

XL. Yassin, Z. T., Hussain, A. S., Tashtoush, M. A., & Az-Zo'bi, E. A.(2025).Comparative analysis of metaheuristic optimization techniques for solving nonlinear fractional Riccati stochastic differential equations.*International Journal of Robotics and Control Systems, 5(5)*, 2612–2637.

XLI. Yildirim, Y., et al.(2020).Optical solitons in birefringent fibers for Radhakrishnan–Kundu–Lakshmanan equation with five prolific integration norms.*Optik, 208*, 164550.