*Journal of Discrete Mathematical Sciences & Cryptography*
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 27 (2024), No. 7, pp. 2219–2230
DOI : 10.47974/JDMSC-2093

# Enhanced network security through algebraic cryptanalysis of elliptic curve cryptography

Madan Mohan Tito Ayyalasomayajula *
*Department of Computer Science*
*College of Engineering*
*Osmania University*
*Hyderabad 500007*
*Telangana*
*India*

Hayder M. A. Ghanimi †
*Department of Information Technology*
*College of Science*
*University of Warith Al-Anbiyaa*
*Karbala 56001*
*Iraq*

*and*

*Department of Computer Science*
*College of Computer Science and Information Technology*
*University of Kerbala*
*Karbala 56001*
*Iraq*

M. Manimaran §
*Department of Cyber Security*
*Easwari Engineering College*
*Chennai 600089*
*Tamil Nadu*
*India*

* *E-mail:* **mail2tito@gmail.com (Corresponding Author)**
† *E-mail:* **hayder.alghanami@uowa.edu.iq**
§ *E-mail:* **drmanimaranm@gmail.com**

Vijaya Krishna Sonthi [‡]
*Department of Computer Science and Engineering*
*Koneru Lakshmaiah Education Foundation*
*Vaddeswaram 522502*
*Andhra Pradesh*
*India*

Abrar Ahmed Katiyan [@]
*Department of Computer Science and Engineering*
*C Abdul Hakeem College of Engineering and Technology*
*Vellore 632509*
*Tamil Nadu*
*India*

Sudhakar Sengan [#]
*Department of Computer Science and Engineering*
*PSN College of Engineering and Technology*
*Tirunelveli 627152*
*Tamil Nadu*
*India*

Pankaj Dadheech [$]
*Department of Computer Science and Engineering*
*Swami Keshvanand Institute of Technology, Management & Gramothan (SKIT)*
*Jaipur 302017*
*Rajasthan*
*India*

**Abstract**

The growing threat of cyberattacks has rendered the research of secure online systems an issue of concern. Elliptic Curve Cryptography (ECC) is an enhanced security benchmark that provides high-security levels with small key sizes. This work exhibits an Algebraic Cryptanalysis (AC) technique to evaluate and enhance ECC security systems with Gröbner Bases (GB). This paper investigates ECC vulnerabilities and recommends a technique for using GB to address the Elliptic Curve Discrete Logarithm Problem (ECDLP). Analysed amid conventional techniques such as Baby-Step Giant-Step (BS-GS), Pollard's Rho (PR),

[‡] *E-mail:* **vijayakrishna1990@gmail.com**
[@] *E-mail:* **kaa406@yahoo.co.in**
[#] *E-mail:* **sudhasengan@gmail.com**
[$] *E-mail:* **pankajdadheech777@gmail.com**

and Pohlig-Hellman (PH), the approach was evaluated. The report proves that the proposed approach has a higher SR, averaging 96.33% throughout all tests.

***Subject Classification:*** *34A06.*

***Keywords:*** *Mathematical modeling, Ordinary differential equations, A duopoly economy, Market share.*

## 1. Introduction

Network security is a proven method to avert cyberattacks in today's dynamic, developing digital world, boosting the volume of data susceptible to attackers. As cyber-attacks grow increasingly sophisticated and complicated, improved security precautions for preventing attackers will be required [1]. Cryptography is a technique that has secured online communication, securing data integrity, confidentiality, and authentication over technologies [2]. Because of its significant security and small key sizes, Elliptic Curve Cryptography (ECC) is a robust method. ECC is fast and may guarantee security in settings with restricted computing power and bandwidth, but privacy is essential [3]. The level of integration of ECDLP has been ECC's core feature. So, fixing or hacking the ECDLP would maliciously break encryption and reveal secure information [4].

ECC risks must be investigated using Algebraic Cryptanalysis (AC). AC has investigated ECC's algebraic components for vulnerabilities [5]. Algebraic algorithms permit cryptanalysts to simulate attack scenarios and assess the ECC model's susceptibility to higher-level cypher attacks [6-8]. This method improves ECC security and enables cryptographic advances to protect against increasing cyber-attacks [9-11]. This paper reports the ECDLP by Gröbner Bases (GB) for AC. The technique exploits Elliptic Curves (EC) algebraic attributes to transform ECC functions into GB-solvable polynomial solutions. The study adapts the ECDLP into a polynomial network to identify new ECC vulnerabilities and assess its security measures against algebraic attacks. This study evaluated the recommended GB solution paradigm to Baby-step Giant-Step (BSGS), Pollard's Rho (PR), and Pohlig-Hellman (PH) algorithms. Computational time (CT), memory usage (MU), success rate (SR), and error rates (ER) serve to evaluate those techniques throughout the scenarios tested.

The paper's organization is as follows: Section 2 provides the background summary of the paper, Section 3 presents the recommended framework, Section 4 presents the study analysis, and Section 5 ends with future work.

## 2. Background

2.1 *ECC*

$$y^2 = x^3 + ax + b \tag{1}$$

ECC is based on the AC of ECs over finite fields, where $'a'$ and $'b'$ are co-efficient that meet condition $4a^3 + 27b^2 \neq 0$ to ensure that the curve does not

have any singular points. This EQU (1) is used over a finite element, either prime '$\mathbb{F}_p$' or binary '$\mathbb{F}_{2^m}$', which impacts the precise operation of the curve. The security of ECC is subject to the effort of the Elliptic Curve Discrete Logarithm Problem (ECDLP), which is the task of defining the number '$k$' assumed the points $P$ and $kP$ on the curve. In ECC, each user creates a PrK, a random number as '$d$' from the interval $[1, n-1]$, where '$n$'→curve. The equivalent PuK is the idea EQU (2).

$$Q = dP \tag{2}$$

where '$P$' is a predefined point on the curve as the base origin. This encryption is subject to the size of '$n$' and the cryptographic features of the curve itself.

ECC is widely used in various network security applications, including:

- ***SSL/TLS Protocols****: SSL/TLS protocols use ECC for the authentication process, key agreement, and key transportation to secure web communications.*
- ***Digital Signatures****: The EC-Digital Signature Algorithm is commonly utilised for software distribution, money transfers, and secure email.*
- ***Secure Messaging****: ECC secures end-to-end encryption in Signal so only the communicating users can read messages.*
- ***Cryptographic Smart Cards****: ECC is used in smart cards for identity verification to secure access to buildings, computer systems, and networks.*

2.2 *GB: Fundamental Concepts and Computational Techniques*

A GB is a set of polynomials that serves as a canonical depiction of a model in a polynomial ring. For a given ideal '$I$' is a set of polynomials $F = \{f_1, f_2, \ldots, f_s\}$ in the polynomial ring $R = k[x_1, x_2, \ldots, x_n]$ over a field $k$, a Gröbner basis for $I$ for a monomial value $>$ is a finite $G = \{g_1, g_2, \ldots, g_t\} \subseteq I$ that has the following properties:

- **Ideal Membership:** The ideal generated by $G$ is the same as the ideal generated by $F$, i.e., $\langle G \rangle = I$
- **Divisibility of Key Terms:** The key polynomial as '$I$' is separable by the key term of at least one polynomial in $G$.

Buchberger's algorithm, announced by Bruno Buchberger in 1965, is the most commonly used method for forming GBs.

- **Initial Setup:** Start with a basis $G = F$.
- **S-Polynomial Computation:** For the set of polynomials $g_i, g_j \in G$, compute the S-polynomial, defined as EQU (3)

$$S(f, g) = \frac{\mathrm{lcm}(\mathrm{LT}(f), \mathrm{LT}(g))}{\mathrm{LT}(f)} \cdot f - \frac{\mathrm{lcm}(\mathrm{LT}(f), \mathrm{LT}(g))}{\mathrm{LT}(g)} \cdot g \tag{3}$$

$\mathrm{LT}(f)$ denotes the key term of $f$, and $1\,\mathrm{cm}$ denotes the least common multiple.

- **Reduction Process:** Reduce each S-polynomial by all elements in $G$. If the reduction results in a non-zero remainder $r$, add $r$ to $G$.
- **Termination:** The process repeats until no new polynomials are added to $G$, indicating that $G$ is a GB.

**Properties and Applications**

- ***Canonical Form:*** *Every non-zero polynomial* $(f)$ *in the ideal I can be uniquely reduced to zero by the GB (*$G$*) of* $(I)$*, providing a canonical form for elements of the ideal.*
- ***Ideal Membership Testing:*** *A polynomial* $(f)$ *belongs to the ideal* $(I)$ *if and only if it decreases to '0' using the GB of* $(I)$*. This property is crucial for solving equations and automated reasoning in AC systems.*
- ***Solution of Polynomial Equations:*** *GB turns polynomial equations into triangular forms that can be solved iteratively with the most minor parameters.*

2.3 *The ECDLP*

Given an EC $(E)$ over a finite field $\mathbb{F}_p$ and two points $P$ and $Q$ on $E$, where $Q = kP$ for some integer $k$, the problem is determining $k$, known as the discrete logarithm. Here, $P$ typically serves as a publicly known base point, and $Q$ is a point obtained by adding $P$ to itself $k$ times. The equation for an EC $(E)$ over $\mathbb{F}_p$ is given in EQU (1). The curve includes a unique value at infinity $'O'$, which acts as the identity portion for the supereminent function defined on the curve.

*Algorithms for Solving ECDLP*

Several algorithms exist for attempting to solve the ECDLP, varying in efficiency based on the curve's properties and the size of the field:

- ***BSGS:*** *This general algorithm applies to any cyclic group, including ECs. It has a time limit of* $O(\sqrt{n})$*, where* $'n'$ *is the order of* $'P'$*.*
- ***PR for Logarithms:*** *Also having a complexity of* $O(\sqrt{n})$*, this algorithm is favored in practice due to its lower memory requirements than the BSGS.*
- ***PH Algorithm:*** *This method is simple if n is factorised, especially if 'P' has a flat order with small primes.*

## 3. Proposed Method for Solving the ECDLP Using GB

The ECDLP centres on finding an integer $'k'$, such that given a base point $'P'$ and a point $Q$ on an EC $(E, Q = kP)$. AC, this can be expressed through polynomial equations derived from the addition laws of ECs. Adapting the ECDLP into a set of linear equations lets GB solve them, but EC functions must be stated as polynomials. For an EC defined by $y^2 = x^3 + ax + b$ over a finite value $'\mathbb{F}_p'$ the addition of two points $(x_1, y_1)$ and $(x_2, y_2)$ to get a third point $(x_3, y_3)$ as EQU (4) to EQU (6).

$$x_3 = \lambda^2 - x_1 - x_2 \tag{4}$$

$$y_3 = \lambda(x_1 - x_3) - y_1 \quad (5)$$

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \quad (6)$$

For value repetition, the EQU (7) modify $'\lambda'$

$$\lambda = \frac{3x_1^2 + a}{2y_1} \quad (7)$$

These operations are AC manipulated to remove denominators and then expressed solely in terms of polynomials over $\mathbb{F}_p$.

***Constructing the System of Equations for the ECDLP:*** To solve the ECDLP using GB, it is essential to convert the problem into a system of polynomials.

**Step 1: Polynomial Representation of Point Addition**

Given an EC $(E)$ over a finite value $\mathbb{F}_p$ defined by $y^2 = x^3 + ax + b$, by setting up the polynomial for point addition.

For two points $P(x_1, y_1)$ and $Q(x_2, y_2)$ on $E$, their sum $R(x_3, y_3) = P + Q$ can be determined using the chord-and-tangent rule of ECs:

1 **Slope Calculation:** Convert EQU (8) into a polynomial by clearing the denominator:

$$\lambda \cdot (x_2 - x_1) - (y_2 - y_1) = 0 \quad (8)$$

2 **X-coordinate and Y-coordinate of the Resultant Point:** The EQU (4) and EQU (5) are polynomials and require no transformation.

**Step 2: Polynomial Representation of Point Doubling**

Point doubling is used when $P = Q$. The doubling a point $P(x_1, y_1)$ to get $R(x_3, y_3)$ are:

- *Slope Calculation for Doubling:* the EQU (9) is converted to a polynomial:

$$\lambda \cdot 2y_1 - (3x_1^2 + a) = 0 \quad (9)$$

- *X-coordinate of the Resultant Point:*

$$x_3 = \lambda^2 - 2x_1 \quad (10)$$

- The y-coordinate of the Resultant Point is the same as EQU (10).

**Step 3: System Formulation for ECDLP**

Given $Q = kP$, need to express $Q$ as the sum of $P$ added to itself $k$ times, *i.e.,* $P + P + \cdots + P(k$ times). To find $k$, establish a system where each successive addition corresponds to a set of polynomial equations based on the addition and doubling rules:

**Step 1: Initialization**

Start with $P_0 = P$, where $P$ is the base point on the EC.

**Step 2: Iteration and Equation Formulation**

- **Iterate:** For every addition up to *'k'*, find the next point using multiplication EQU.
- If $i = 0, P_1 = 2P_0$ (doubling)
- For $i > 0, P_{i+1} = P_i + P_0$ (addition)
- **Apply Formulas:**
- Doubling $P_i$ to get $P_{i+1}$ :

$$\lambda_i = \frac{3x_i^2+a}{2y_i};\ x_{i+1} = \lambda_i^2 - 2x_i;\ y_{i+1} = \lambda_i(x_i - x_{i+1}) - y_i$$

- Adding $P_i$ and $P_0$ to get $P_{i+1}$ :

$$\lambda_{i,0} = \frac{y_0-y_i}{x_0-x_i};\ x_{i+1} = \lambda_{i,0}^2 - x_i - x_0;\ y_{i+1} = \lambda_{i,0}(x_i - x_{i+1}) - y_i$$

- **Formulation:** For Each multiplication, the equations without denominators are multiplied by the denominators involved in $\lambda$.

**Step 3: Aggregation**

Collect all equations into a single system that includes every polynomial generated from the iterative point addition steps.

**Step 4: Solving the System**

- **Compute GB:** Use the complete system of polynomial equations as input and compute the GB.
- **Simplify the System:** Analyze the resulting GB to determine if the system has been simplified to a point where $k$ can be solved directly.
- **Solve for $k$ :** Extract $k$ from the simplified system

## 4. Experimental Setup

The simulation setup is led on a personal computer with an Intel Xeon processor, 32 GB of RAM, and a 512 GB SSD with the operating system Linux Ubuntu. Mathematica is utilized for general mathematical computations, and Macaulay2 is used for calculating GB.

a) **EC Parameters:** The set of points Q=kP is generated for experiments using an EC over the finite number '$F_p$', where '$p$' is a prime value. The monomial ordering is set to graded reverse lexicographical order, ensuring efficient management of the polynomial reduction process during the experiments.

**b) Baseline**

- *BSGS:* A general-purpose algorithm used in discrete logarithm problems, which operates with a time complexity of $O(\sqrt{n})$.
- *PR:* Another general-purpose algorithm for discrete logarithms with similar complexity to BSGS but usually preferred for its lower memory usage.

- *PH:* Group order factorization is key to this algorithm, which breaks equations into minor discrete logarithm problems when group order is smooth.

**c) Metrics for Analysis in GB Application to ECDLP**

The proposed model is compared against three baseline algorithms, BSGS, PR, and PH, for the following metrics :

- **Computational Time (CT):** EQU (11) collects CT from the start to finish of the GB computation for each ECDLP instance.

$$T = t_{\text{end}} - t_{\text{start}} \tag{11}$$

where $t_{\text{start}}$ is the start time, and $t_{\text{end}}$ is the end time.

- **Memory Usage (MU):** MU tracks for GB analysis to find the maximum RAM used, EQU (12).

$$M_{\text{peak}} = \max\left(M_{\text{usage}}(t)\right) \tag{12}$$

where $M_{\text{usage}}(t)$ represents the memory usage at time $t$.

- **Success Rate (SR):** SR = EQU (13), the number of tests where the GB finds the correct '$k$' value.

$$S = \frac{n_{\text{correct}}}{n_{\text{total}}} \times 100\% \tag{13}$$

where $n_{\text{correct}}$ is the number of tests where $k$ was correctly computed, and $n_{\text{total}}$ is the number of test executions.

- **Error Analysis (EA):** EA involves examining instances where the computations either fail or generate wrong results, EQU (14),

$$E = \{e_1, e_2, \ldots, e_n\} \tag{14}$$

where $e_i$ represents a specific type of error encountered during the computations. Table 1 shows models for the metrics in three test cases.

**Table 1**
**Test Case Descriptions**

| Test Case | Scenario Description | Focus of Evaluation |
|---|---|---|
| **Case 1** | Small prime number defining the finite field. | Primary algorithm performance and CT. |
| **Case 2** | Significant prime numbers complicate ECDLP. | Algorithm scalability and robustness. |
| **Case 3** | Complex points on a curve cause lengthy computations. | Efficiency in handling complex, non-linear cases. |

4.1 *Analysis of CT*

Figure 1 contrasts BSGS and PR- Computation Time (CT), with BSGS showing variability and PR increasing with complexity. PR's -CT increases with prime fields (Test Case 1 to 2) and group sizes, while PH's complexity increases CT (Test Case 3). PR works well for complex problems and even outperforms

BSGS when average times are similar. The proposed GB had the lowest average times across all test cases and linearly increased performance with complexity ($n$). Lower standard deviations indicate less impact on curve variation.

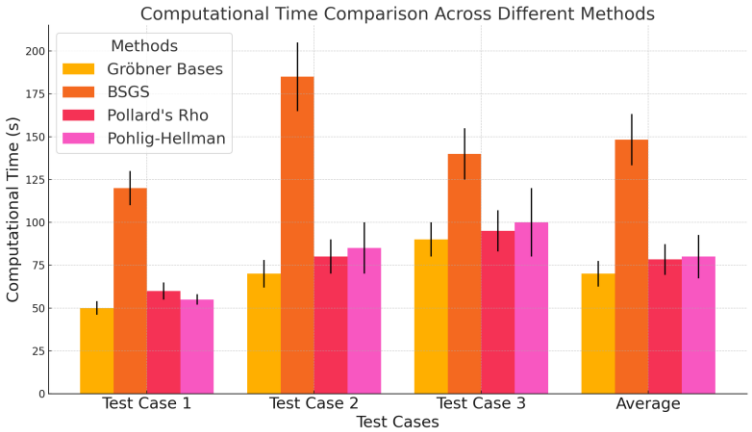


**Figure 1**
**CT Analysis**

4.2 *Analysis of MU*

Figure 2 presents the results for MU under different test cases. The proposed GB approach requires high memory due to the need to handle large polynomial systems and multiple terms during the computation of the bases.

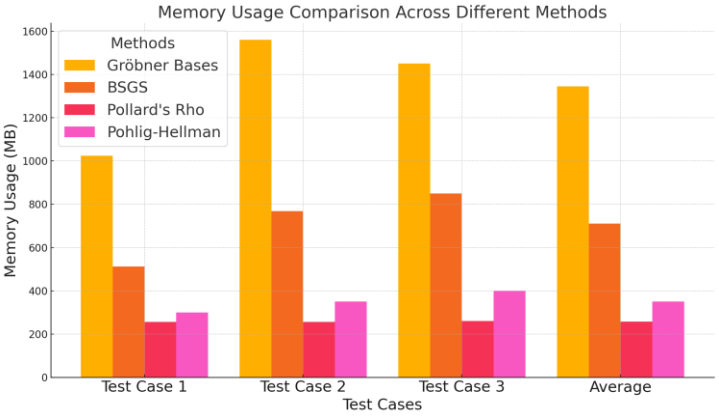


**Figure 2**
**MU Comparison**

The increase from Test Case 1 to Test Case 2 reflects the rising complexity of the EC settings. The BSGS method shows an increase in MU as the complexity of the test cases increases. The memory requirements are directly

related to the square root of the group's order. PR has shown low MU across all test cases, which is ideal for limited memory resources. PH exhibits moderate MU, which increases with the complexity of the EC.

### 4.3 *Analysis of SR*

Figure 3 displays the performance of compared models for the success rate. The proposed GB model has shown the highest SR across all test cases.


**Figure 3**
**SR Comparison**

This suggests that GB are well-suited to handling complex AC structures inherent in ECDLP. BSGS and PR both show high but slightly lower SR compared to GB. These methods are traditionally favored for their simplicity and lower computational overhead but struggle with intricate problems. PHn also demonstrates high SR, mainly when the group order is smooth. However, it slightly underperforms compared to GB in the most complex scenarios.

### 4.4 *Analysis of ER*

The output of the ER analysis is predictable in Figure 4. The findings exhibited that the proposed GB has demonstrated relatively low ER, and the gradual increase in ER with complexity aligns with the growing difficulty in accurately managing larger systems of polynomial equations. BSGS-ER increases significantly with complexity due to its susceptibility to inefficiencies in managing larger group sizes and more complex curve parameters. PR shows the highest ER due to the inherent randomness and detection failures in more complex scenarios, reflecting its probabilistic nature. PH maintains low ER, only moderately increasing with complexity.


**Figure 4**
**ER comparison**

## 5. Conclusion and Future Work

The study presented a mechanism by exploring the usage of GB for Algebraic Cryptanalysis (AC) in ECC, particularly for solving the ECDLP. The GB has provided an AC approach to enhance ECC's security by addressing the ECDLP. The proposed model was compared against other traditional models, like BSGS, PR, and PH, for different metrics. The proposed model showed better SR and ER reduction results but displayed higher CT and MU than traditional methods. This limitation showcases the need for the GB to be optimized to improve its efficiency in practical applications.

The future work will focus on optimizing the model by integrating hybrid approaches.

## References

[1] M. Thakur, "Cyber security threats and countermeasures in digital age," *Journal of Applied Science and Education* (JASE), vol. 4, no. 1, pp. 1-20 (2024).

[2] S. Gadde, G. S. Rao, V. S. Veesam, M. Yarlagadda, and R. S. M. Patibandla, "Secure data sharing in cloud computing: A comprehensive survey of two-factor authentication and cryptographic solutions," Ingénierie des Systèmes d'Information, vol. 28, no. 6 (2023).

[3] Z. Vahdati, S. Yasin, A. Ghasempour, and M. Salehi, "Comparison of ECC and RSA algorithms in IoT devices," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 16, p. 4293 (2019).

[4] J. J. Puthiyidam, S. Joseph, and B. Bhushan, "Temporal ECDSA: A timestamp and signature mask enabled ECDSA algorithm for IoT

client node authentication," *Computer Communications*, vol. 216, pp. 307-323 (2024).

[5] H. M. A. Ghanimi, T. Gopalakrishnan, G. J. Deol Sunny, K. Amarendra, P. Dadheech, and S. Sengan, "Chebyshev polynomial approximation in CNN for zero-knowledge encrypted data analysis," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 203–214 (2024). DOI: 10.47974/JDMSC-1880.

[6] H. M. A. Ghanimi, R. P. Melgarejo-Bolivar, A. Tumi-Figueroa, S. Ray, P. Dadheech, and S. Sengan, "Merkle-Damgård hash functions and blockchains: Securing electronic health records," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 237–248 (2024). DOI: 10.47974/JDMSC-1878.

[7] P. Vidya Sagar, H. M. A. Ghanimi, L. A. Prabhu, L. Raja, P. Dadheech, and S. Sengan, "Secure multi-party computation in deep learning: Enhancing privacy in distributed neural networks," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 249–259 (2024). DOI: 10.47974/JDMSC-1879.

[8] G. R. K. Rao, H. M. A. Ghanimi, V. Ramachandran, D. Al-Qahtani, P. Dadheech, and S. Sengan, "Enhanced security in federated learning by integrating homomorphic encryption for privacy-protected, collaborative model training," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 361–370 (2024). DOI: 10.47974/JDMSC-1891.

[9] R. J. Krishna, T. Gopalakrishnan, M. Divyapushpalakshmi, K. Amarendra, P. Dadheech, and S. Sengan, "Security and privacy concerns in social networks mathematically modified metaheuristic-based approach," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 371–382 (2024). DOI: 10.47974/JDMSC-1892.

[10] S. H. Nowfal, G. R. K. Rao, V. Velmurugan, S. Sengan, R. K. Bommisetti, and P. Dadheech, "Advancing viscoelastic material modeling: Tackling time-dependent behavior with fractional calculus," *Journal of Interdisciplinary Mathematics*, vol. 27, no. 2, pp. 307–316 (2024). DOI: 10.47974/JIM-1827.

[11] P. V. Sagar, M. Rajyalaxmi, A. V. V. S. Subbalakshmi, S. Sengan, R. K. Bommisetti, and P. Dadheech, "Utilizing stochastic differential equations and random forest for precision forecasting in stock market dynamics," *Journal of Interdisciplinary Mathematics*, vol. 27, no. 2, pp. 285–298 (2024). DOI: 10.47974/JIM-1822.

*Received March, 2024*