AN EFFICIENT (MBO-ED25519) ALGORITHM FOR SECURING STUDENT ACADEMIC RECORDS

S. Syed Nawas Husain[1], R. Balasubramanian[2]

[1]Research Scholar, Department of Computer Science and Engineering, Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India-627012.

[2]Professor, Department of Computer Science and Engineering, Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India-627012.

Article History: Received: 13.04.2023 **Revised:** 08.05.2023 **Accepted:** 19.06.2023

Abstract: *Security is the cornerstone of all kinds of digital transformation. As the demand for secure data encryption grows, the algorithms we use must be secure. Algorithms are not always the issue, it is also about settings, like key length. Security of students' academic records is essential when their relevant data are being transmitted over public networks using computerized data. Research on the improvement of different optimization algorithms has become increasingly important as optimization problems have become more complex. Monarch Butterfly Optimization (MBO) has proved to be an effective algorithm for solving a wide range of optimization problems. A hybrid MBO and Ed25519 is developed in this paper in order to address these issues. This paper proposes an approach for Ed25519 that optimizes the calculation steps for the need of a monarch butterfly optimization in order to speed up factorization time in affine coordinates. In general, proposed equations can be used for Elliptic Curve Cryptography (ECC) and Elliptic Curve Digital Signature Algorithm (ECDSA), which use the same principle and thus can be more efficient. In Ed25519, elliptic curves are combined with defined group operations to form groups. To ensure fast implementation, we optimize these operations based on specific architectures or approaches. In the realm of communication confidentiality, several essential expectations come into play to ensure the preservation and secure transmission of information. These expectations include data integrity, data origin authentication, entity authentication, non-repudiation, and maintaining confidentiality during data exchange. Tests and comparisons of performance revealed that the proposed algorithm improves Time computation efficiency significantly. As a result, it has practical significance for monarch butterfly optimization and will be applied to Ed25519 as well.*

Keywords: *MBO, Ed25519, MBO-Ed25519, Optimal key generation and Signature Verification.*

1. INTRODUCTION

Developing the best educational strategies for students requires lecturers and decision-makers to predict student performance [5]. Students' performance can be evaluated using several academic parameters [3], such as result grades, Grade Point Averages (GPAs), lecture absenteeism, and number of tries to pass a course or exam. The other demographic factors include gender, household composition, parent occupation, marital status, and personal habits. Data might still be biased, imbalanced, and missing after collection [6], as well as different types of data, such as strings, numbers, and letters. The large number of attributes (features) present one of the biggest challenges in this area. In order to improve the performance of students, it is necessary to identify the highly valuable features.

In recent years, the scholars have made many improvements to MBO in order to improve its performance. Following the implementation of the migration operator in MBO [18], a new monarch butterfly will be accepted regardless of its superiority or inferiority in the next generation. The implementation of Ed25519 [15] is evaluated from the point of view of security and performance. Three modules made up the implementation: field arithmetic, curve arithmetic, and the interface. Cryptographic implementations need to exhibit essential characteristics such as functionality correctness, memory safety, constant-time operations, and usability. It is based on the Curve25519, which is twisted Edwards curve, and provides 128 bits of security [7]. Ed25519 is the most popular Edwards-curve Digital Signature Algorithm (EdDSA).

1.1 Monarch Butterfly Optimization (MBO)

Implementing the Monarch Butterfly Optimization (MBO) [9] is simple and straightforward. In MBO, two subpopulations of equal size are called subpopulation1 and subpopulation2. In subpopulation1, half of the individuals have the best fitness values, and in subpopulation2, the rest have the lowest fitness values. Within the context of Management by Objectives (MBO), two distinct strategies are employed: the migration operator and the butterfly adjusting operator. These strategies play essential roles in the MBO framework and aim to enhance goal-setting, performance evaluation, and overall organizational effectiveness.

Table 1: Monarch Butterfly Optimization Operators [4]

S.No	Migration Operator	Adjusting Operator
1	As a result, solutions can explore different areas and potentially find better solutions by simulating monarch butterfly migration behavior	A search space's solutions are modified or adjusted within a given region.
2	By exchanging information between different regions, migration operators maintain diversity in the population.	This approach refines and improves a specific region's solutions based on individual solutions.
3	This process facilitates exploration by allowing solutions to move to unexplored areas, preventing premature convergence, and facilitating the discovery of new and improved solutions.	By improving the quality of solutions within a region, it promotes exploitation.

1.2 Elliptic Curve

A public key cryptographic algorithm, Elliptic Curve Cryptography (ECC) relies on two keys, private and public, and is used to both authenticate people and protect data. Depending on the usage, either key is used for encryption or decryption [13]. When encryption is performed by the user, the private key is used, and when authentication occurs, the public key is used to identify the user. In cases of confidentiality, the sender encrypts with the private key and decrypts with the public key. Mathematical foundations of elliptic curves come in many shapes and forms. All of them have different mathematical proofs and addition formulas. Ellipstic curves are mathematical curves used in modern cryptography, such as Weierstrass, Koblitz, and Edwards.

1.2.1 Weierstrass

Developed by the German mathematician Karl Weierstrass in the 19th century, the Weierstrass [13] function is a mathematical function. Functions display self-similarity across scales, which is part of what makes them fractals. As an infinite series of sine and cosine functions, the function is scaled and shifted by a parameter that controls the degree of self-similarity.

1.2.2 Koblitz

Its non-random construction makes Koblitz [8] curves one of the most efficient types of elliptic curves available. In contrast, the most commonly used elliptic curves have pseudo-random parameters that are selected by an algorithm.

1.2.3 Edwards

An elliptic curve form was introduced by Harold Edwards [16] in 2007. Edwards Curves were then named after this form. There is a growing interest in elliptic curves in Edwards form in the crypto community these days. As elliptic curves have a simple, elegant addition law, they are faster, simpler, and more elegant. It is difficult to solve the discrete logarithm problem with Edwards's curves since they are more efficient than most other elliptic curves. Cryptographic applications can be implemented using Edwards curves. Using EdDSA and Curve25519, Ed25519 is an elliptic curve signing algorithm.

1.2.3.1 Edwards25519 (Ed25519)

Using public key encryption, the [12] digital signature is computed, authenticated and then attached to an electronic document as a means of verifying its authenticity. According to Bernstein et al., EdDSA is based on Edwards curves. EdDSA is defined as one of Edwards's twisted curves (Ed25519). There are certain parameters that need to be specified when instantiating EdDSA. ECDLP (Elliptic Curve Discrete Logarithm Problem) provides security for Ed25519 and Curve25519. In Ed25519, there are three stages: generating keys, creating signatures, and verifying signatures [7].

Table 2: The Procedure of Digital Signature

Text	**Ed25519**
Private key length	32 bytes
Public key length	32 bytes
Signature size	64 bytes
Public key recovery	Not possible (Signature verification involves hashing of the public key)
Security level	128bit

1.3 Key Generation

In [8] process of creating cryptographic keys for various encryption, decryption, and authentication algorithms is known as key generation.

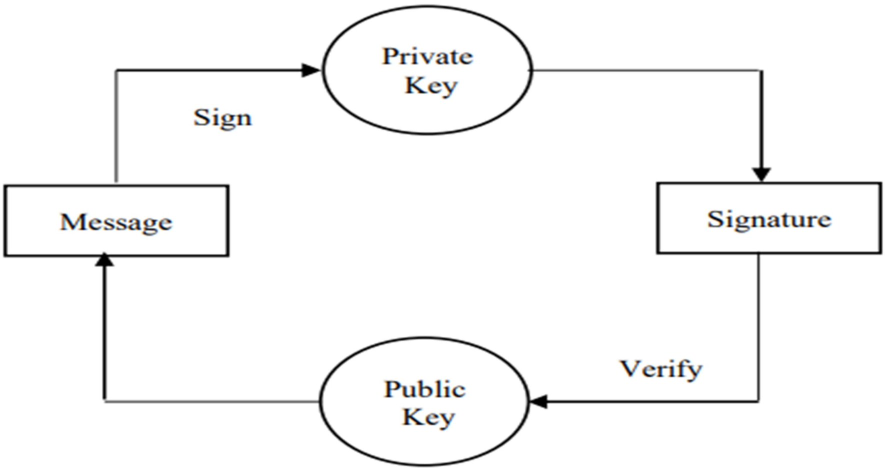


Figure 1: Key Generation Process

The purpose of this paper is therefore to present a hybrid model for key generation that uses combine monarch butterfly optimization and Edwards25519 (MBO-Ed25519).

2. LITERATURE REVIEW

2.1 Monarch Butterfly Optimization (MBO)

Through the information exchange between monarch butterflies, the MBO algorithm simulates the migration process of monarch butterflies. In MBO, all butterfly individuals reside in Land1 and Land2 after simplification and idealization, and butterfly position updates are achieved through migration and adjustment. Adjustment and migration operations can be executed concurrently, allowing the MBO to be processed in parallel and the global and local search capability to be considered [4].

In the [9] Management by Objectives (MBO) model, the population is divided into two equal-sized subpopulations: subpopulation1 and subpopulation2. Each subpopulation follows a different updating approach for their positions. Individuals in subpopulation1 utilize migration operators to update their positions, whereas individuals in subpopulation2 employ butterfly adjusting operators to modify their positions. This dual-strategy approach allows for diverse and efficient optimization processes within the MBO framework. Various optimization problems have been solved using [18] BOA because of its performance. Despite its advantages, BOA suffers from drawbacks, such as diminished diversity among populations and the tendency to get stuck in local optimum conditions. As a population-based algorithm, MBO belongs to a group of swarm intelligence algorithms influenced by the behavior of certain species with a swarm tendency such as bees, butterflies, and the like [17]. For the purpose of enhancing MBO's optimization ability, Meng et al [2] proposed an improved MBO (IMBO). As opposed to the original MBO approach, IMBO divides the two subpopulations dynamically and randomly at each generation.

2.2 Elliptic Curve

A high hardware cost component in ECC [8] is Elliptic Curve Point Multiplication (ECPM), which is applied on hardware targeted at accelerating its calculus. Especially for hardware implementation, polynomial basis and projective coordinates make this combination ideal. Choosing the best combination of methods and algorithms for different application classes can lead to high efficiency and performance on resource-constrained devices that use reliable algorithms. The DNA encoding method is used for Diffie-Hellman encryption, which transforms the plain image into DNA matrix using the elliptic curve Diffie-Hellman algorithm (ECDHE). Each component is added together, and ECDHE [1] produces the cipher image. The implementation of prime field arithmetic and elliptic curve operations through software is accelerated. We developed a high-performance AVX2-ready software library through our contributions. Ed25519 and Ed448 computations are 19% and 29% faster than previous optimized implementations, respectively. The point addition formula of twisted Edwards curves was optimized at a higher level by parallel decomposition. The performance and security of cryptographic software are affected by a variety of factors. As a result, we revisited countermeasures used in the Montgomery ladder algorithm [19].

2.3 Edwards25519

Compared to the best previous work, [7] efficient Ed25519 scheme achieves more than 8× speedup. Digital signature algorithms can be processed at a speed of over 6200 per second with the high-performance architecture. Area–time product efficiency is improved by over 84% using pipelined architectures and interleaved multiplication. We are able to perform over 6200 and 2200 signings per second as well as 5100 and 1500 verifications per second through our high-performance and efficient architectures. [20] Using Ed25519 algorithm, the digital signatures will be converted into different byte numbers, which will increase the security of the security system. This research work authenticates a security framework that improves security and solves problems.

2.4 Key Generation

All operations of DSA are supported by Cryptography in [14], including key generation, signature generation, and signature verification. Designed for multi-core processors, the system generates 21686 keys per second. Their design requires too many resources for only key generation, limiting their applications to only key generation despite having impressive throughput and number. By employing their Digital Signal Processor blocks (DSP), Curve25519 is more efficient on reconfigurable hardware in [15]. A second architecture with dedicated inverter stages can achieve more than 32,000 point multiplications per second, despite moderate resource requirements. In many future security applications, this speed will result from software-based and hardware-based deployments. Crypto wallets are signed using the [21] Edwards-curve Digital Signature Algorithm. Performance, security, and comparative analysis were examined to ensure maximum usability. It is the main conclusion of the article that Ed25519 can be used.

2.5 Hybrid Monarch Butterfly Optimization (MBO)

In [22], a new multi-objective cell formation problem (CFP) meta-heuristic is proposed based on monarch butterfly optimization (MBO) and firefly optimization (FF). The hybridized MBO-FF approach acquires optimal arrangements in a reasonable amount of time, especially for large-scale problems aimed at improving the CFP's performance. As a result of this algorithm, the search space can be investigated effectively within a short period of time and the global optimal can be recognized quickly. A performance enhancement is based on the percentage of exceptional elements, the use of machines, the efficiency of groupings, and the efficiency of cells.

Monarch butterfly optimization with attribute-based encryption (MBO-ABE) is described in [23]. The MBO method, which is based on monarch butterfly migration, is applied to the ABE technique to improve its security performance. Simulated results demonstrate the enhanced efficiency of MBO-ABE. Compared to the current state of art methods, the MBO-ABE technique displayed the highest performance. A public and private key pair are generated during the key generation process of the ECC method [10]. Randomly generated public keys are used during encryption. Using the genetic algorithm (GA)-based optimization technique, the proposed method generates the private key (H).

A hybrid swarm optimization approach will be used to select the optimal key in the elliptic curve cryptography process. This will include both grasshopper optimization and particle swarm optimization. To accomplish a right message, the proposed hybrid encryption algorithm ECC with PSO and GO uses a multinomial to encrypt and decode. Both the encryption and decryption processes are faster with the proposed algorithm. To achieve content-based respectability instead of the strict-integrity functionality currently employed, future work should focus on tamper localization schemes [11].

3.METHODOLOGY

Students' academic achievement greatly influences an educational institution's success. We propose to build a model to improve Security by utilizing appropriate optimization techniques and key generation algorithms. The purpose of optimization in real life is to generate an optimal solution by selecting a vector from a special domain. Students' performance applications have increasingly focused on constrained optimization as it is closer to real life. There have been many improvements in optimization algorithms as optimization problems have become more complex.

The monarch butterfly optimization (MBO) algorithm is a powerful method employed to tackle various optimization problems. However, when dealing with complex optimization challenges, the basic MBO algorithm may encounter issues leading to premature convergence and suboptimal performance. Specifically, the search strategy used in the basic MBO algorithm is prone to getting stuck in local optima, thereby hindering the algorithm's ability to explore and find better solutions. To overcome this limitation and improve performance, researchers and practitioners often employ advanced techniques and modifications to enhance exploration capabilities, escape local optima, and achieve better overall optimization results. In monarch butterfly optimization (MBO), two operators (migration and butterfly adjusting operator) are repeated until the stop conditions are satisfied. Regardless of whether the newly generated individuals have a better fitness level or not, they are passed on to the next generation in the MBO algorithm.

Ed25519 is designed to operate at around 128 bits of security. Among signature schemes at similar security levels, Ed25519-Original is substantially faster at signing and verifying signatures than almost all others. In addition to providing smaller signatures than comparable schemes, Ed25519-Original also generates 32-byte public keys and 64-byte signatures. In terms of efficiency and provably secure signatures, Ed25519 is probably one of the most familiar.

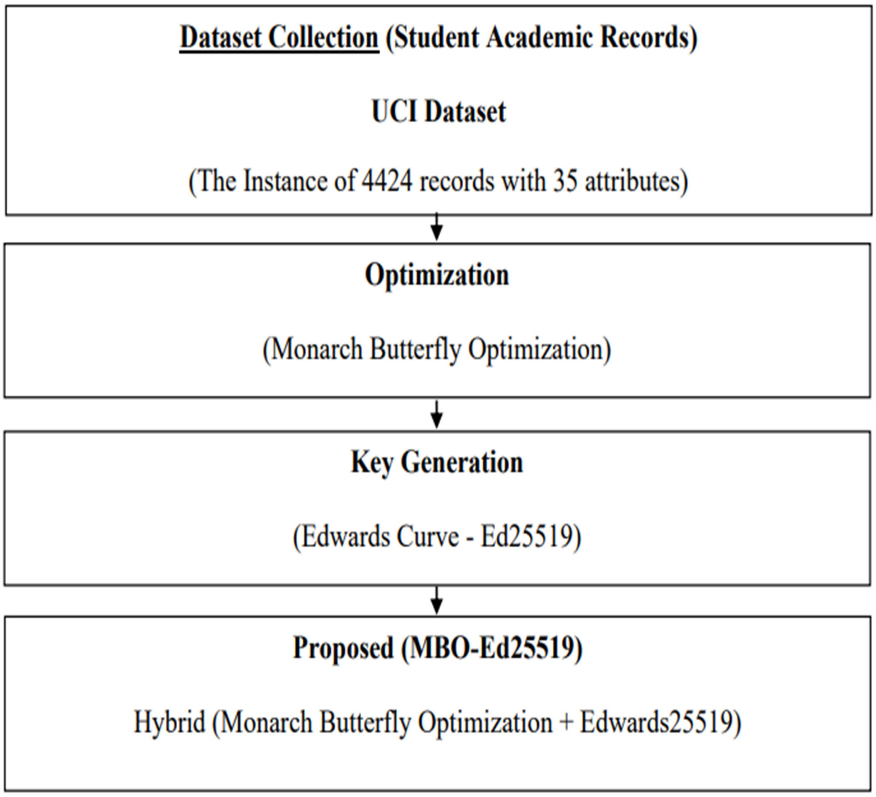

Dataset Collection (Student Academic Records)
UCI Dataset
(The Instance of 4424 records with 35 attributes)
Optimization
(Monarch Butterfly Optimization)
Key Generation
(Edwards Curve - Ed25519)
Proposed (MBO-Ed25519)
Hybrid (Monarch Butterfly Optimization + Edwards25519)

Figure 2: Overall Proposed Architecture

3.1 Monarch Butterfly Optimization (MBO)

A well-established metaheuristic algorithm should be compared with Monarch Butterfly Optimization (MBO) to evaluate its applicability and performance in a particular problem domain. Using exploration, exploitation, and return phases to search for optimal solutions to optimization problems, it is a nature-inspired optimization technique. By balancing exploration and exploitation, as well as maintaining population diversity, the Monarch Butterfly Optimization algorithm seeks to find optimal or near-optimal solutions. In each phase, specific mechanisms and parameters can be adjusted according to the problem characteristics.

1. Migration Operator

In subpopulation1 of the monarch butterfly optimization (MBO) algorithm, the movement of an individual, denoted as "i," is influenced by the positions of other individuals within subpopulation1 and subpopulation2. The extent of this influence is controlled by an adjusting ratio "p," which helps balance the impact of each subpopulation on the movement.

To achieve information exchange between the two subpopulations and promote exploration, the migration operator facilitates the movement of individuals not only within their own subpopulation but also between the two subpopulations.

$$x_{i,k}^{t+1} = \begin{cases} x_{r1,k}^{t}, & \text{if } r \leq p \\ x_{r2,k}^{t}, & \text{else} \end{cases} \tag{1}$$

where $x_{i,k}^{t+1}$ indicates the *kth* dimension ofx_i at generation t+1. In the monarch butterfly optimization (MBO) algorithm, the parameters "r1" and "r2" are integer indices that are randomly selected from subpopulation1 and subpopulation2, respectively. These indices are used to identify individuals within each subpopulation.

Additionally, there is a parameter "r" that is calculated as "rand * peri," where "rand" is a random real number in the range [0, 1], and "peri" represents the migration period.

To summarize:

✓ "r1" is an integer index randomly selected from subpopulation1.

✓ "r2" is an integer index randomly selected from subpopulation2.

✓ "r" is a random real number in the range [0, 1], multiplied by "peri," which represents the migration period.

These parameters play a crucial role in the monarch butterfly optimization algorithm, as they introduce randomness and determine the influence of individuals from different subpopulations during the optimization process.

2. Adjusting Operator

In the monarch butterfly optimization (MBO) algorithm, the movement of an individual "i" in subpopulation2 is influenced by three factors: the global best individual, a random individual in subpopulation2, and the Levy flight. These factors work together to guide the movement and exploration of individuals in subpopulation2 during the optimization process. Three factors are primarily considered when using butterfly adjusting operators:

1) Effect of moving to a global optimum on the social model.

(2) A random individual's cognitive effects on another individual.

(3) Introducing Levy flights in optimization algorithms is a common technique to enhance population diversity and explore a broader search space. Levy flights are characterized by random steps with heavy-tailed distributions, allowing for long jumps in the search space. In the monarch butterfly optimization (MBO) algorithm, Levy flights are utilized to promote exploration in subpopulation2.

$$x_{i,k}^{t+1} = \begin{cases} x_{best,k}^{t} & \text{if rand } \leq p \\ x_{r3,k}^{t} & \text{if rand } > p \wedge \text{ rand } \leq BAR \\ x_{i,k}^{t} + \alpha \times (\mathrm{d}x_k - 0.5) & \text{if rand } > p \wedge \text{ rand } > BAR \end{cases} \tag{2}$$

Where $x_{\text{best},k}^{t}$ is the element of the current global optimum is determined. The parameter r3 represents an integer index randomly selected from subpopulation2. BAR denotes the butterfly adjusting rate. The weighting factor α and dx can be assigned as follows:

$$\alpha = S_{max}/t^2 \tag{3}$$

$$dx = \text{Levy }(x_i^t) \tag{4}$$

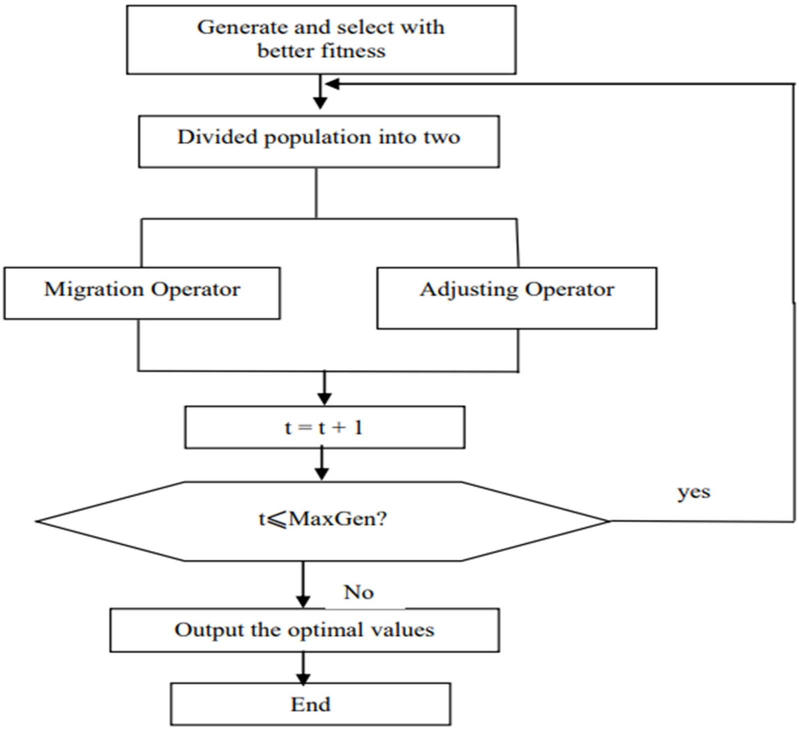
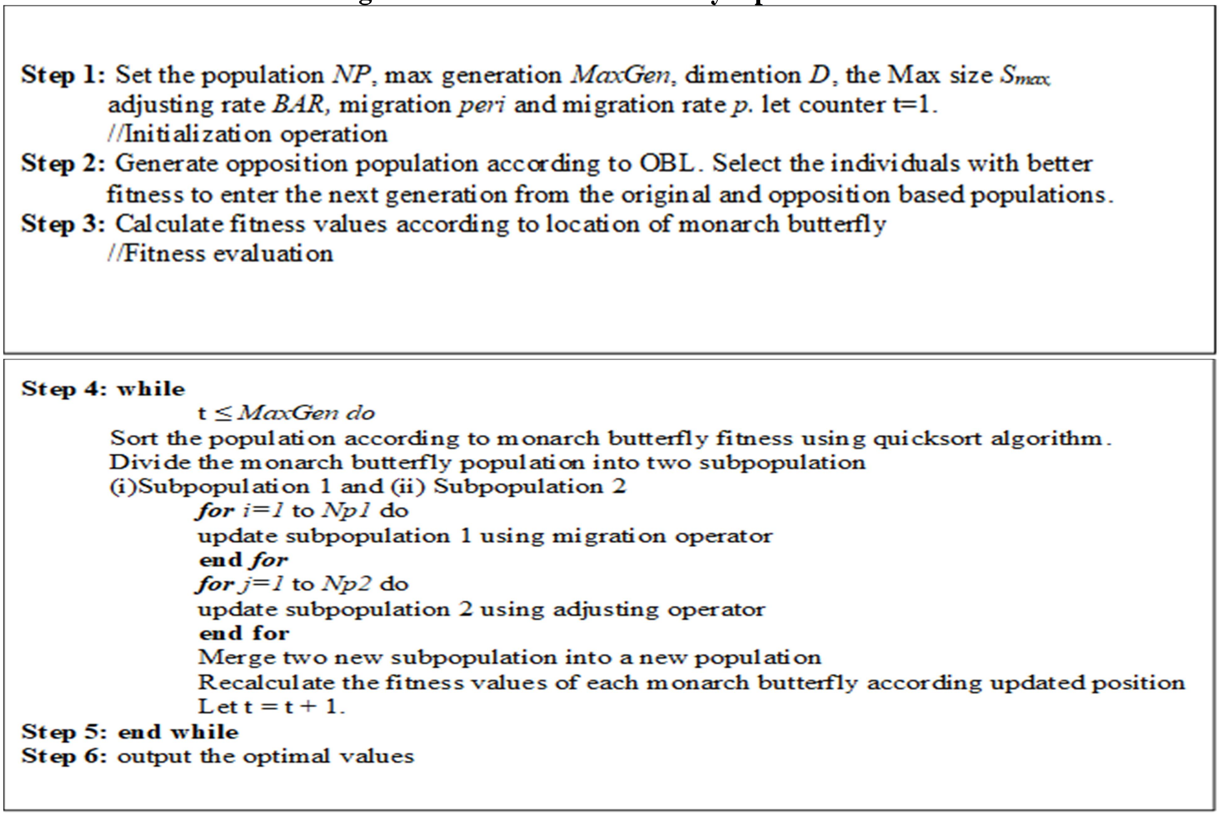

Generate and select with better fitness
Divided population into two
Migration Operator
Adjusting Operator
t = t + 1
t⩽MaxGen?
yes
No
Output the optimal values
End

Figure 3: Monarch Butterfly Optimization

Algorithm 1: Monarch Butterfly Optimization



```
Step 1: Set the population $NP$, max generation $MaxGen$, dimention $D$, the Max size $S_{max}$
        adjusting rate $BAR$, migration $peri$ and migration rate $p$. let counter t=1.
        //Initialization operation
Step 2: Generate opposition population according to OBL. Select the individuals with better
        fitness to enter the next generation from the original and opposition based populations.
Step 3: Calculate fitness values according to location of monarch butterfly
        //Fitness evaluation

Step 4: while
                $t \leq MaxGen$ do
        Sort the population according to monarch butterfly fitness using quicksort algorithm.
        Divide the monarch butterfly population into two subpopulation
        (i)Subpopulation 1 and (ii) Subpopulation 2
                for $i=1$ to $Np1$ do
                update subpopulation 1 using migration operator
                end for
                for $j=1$ to $Np2$ do
                update subpopulation 2 using adjusting operator
                end for
                Merge two new subpopulation into a new population
                Recalculate the fitness values of each monarch butterfly according updated position
                Let t = t + 1.
Step 5: end while
Step 6: output the optimal values
```

3.2 Edwards25519 (Ed25519)

The Ed25519 algorithm uses EdDSA and Curve25519 for elliptic curve signing. EdDSA is a signature algorithm based on Schnorr and uses the difficulty of the ECDLP problem to generate signatures. A 255-bit curve, such as the Curve25519. Fast single-signature verification is one of the most attractive features of Ed25519, a public-key signature system. According to all metrics, Ed25519 performs the best. Compared to key length, EdDSA offers the highest level of security. A private/public key pair must be generated by the signer, with the private key used for signing and the public key made available for verification. Using the private key with the combined hash, we compute the Ed25519 signature by concatenating the proof options hashes, followed by the credential hash without proof.

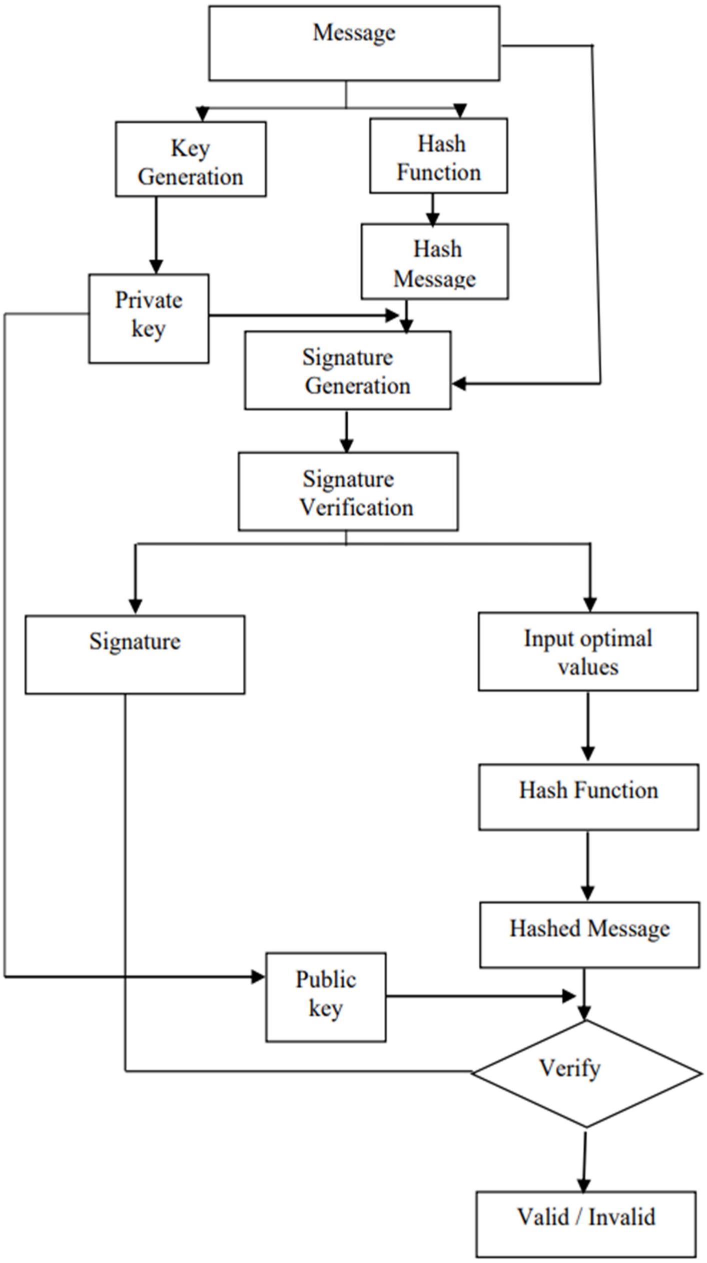
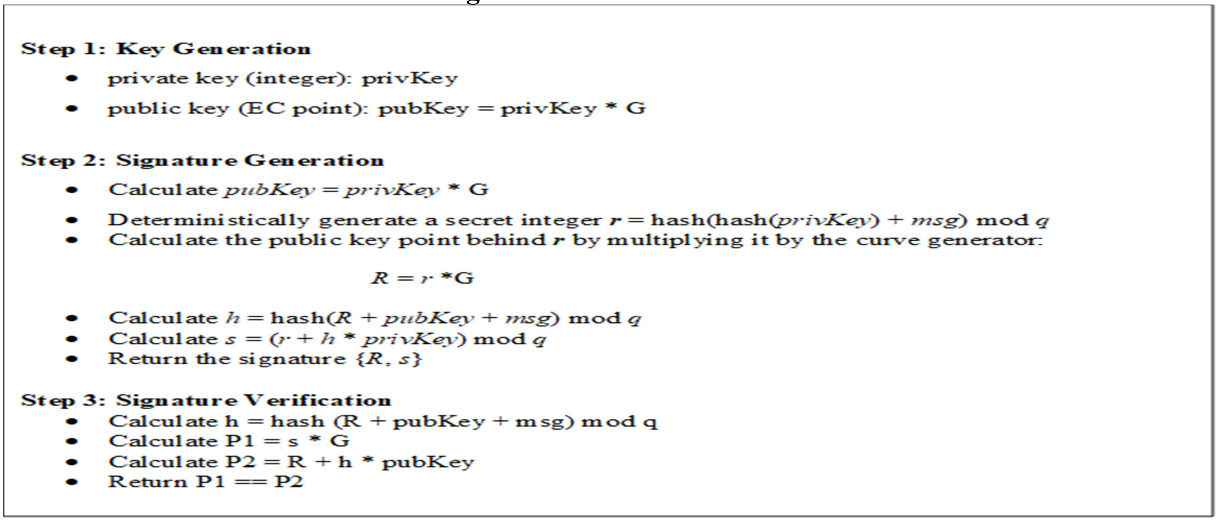


Figure 4: Ed25519 Digital Signature

Algorithm 2: Edwards25519



```
Step 1: Key Generation
    • private key (integer): privKey
    • public key (EC point): pubKey = privKey * G

Step 2: Signature Generation
    • Calculate $pubKey = privKey$ * G
    • Deterministically generate a secret integer $r$ = hash(hash($privKey$) + $msg$) mod $q$
    • Calculate the public key point behind $r$ by multiplying it by the curve generator:
                              $R = r$ *G
    • Calculate $h$ = hash($R$ + $pubKey$ + $msg$) mod $q$
    • Calculate $s$ = ($r$ + $h$ * $privKey$) mod $q$
    • Return the signature {$R$, $s$}

Step 3: Signature Verification
    • Calculate h = hash (R + pubKey + msg) mod q
    • Calculate P1 = s * G
    • Calculate P2 = R + h * pubKey
    • Return P1 == P2
```

3.3 Proposed (MBO-Ed25519)

This work proposes hybrid models that combine Monarch Butterfly Optimization (MBO) with Edwards25519 (Ed25519), with a strong emphasis on signature verification before selecting the optimal values. Each disadvantage is overcome by a hybrid technique proposed in this paper.

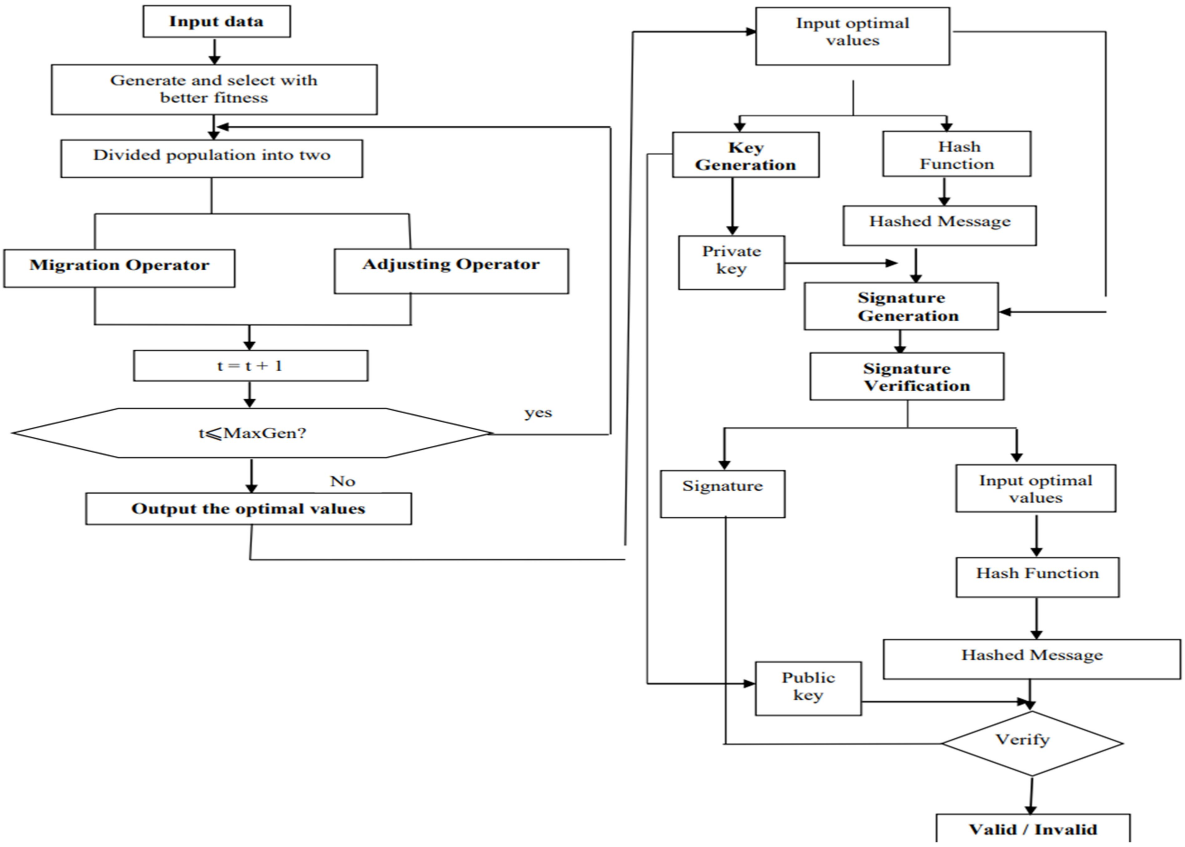
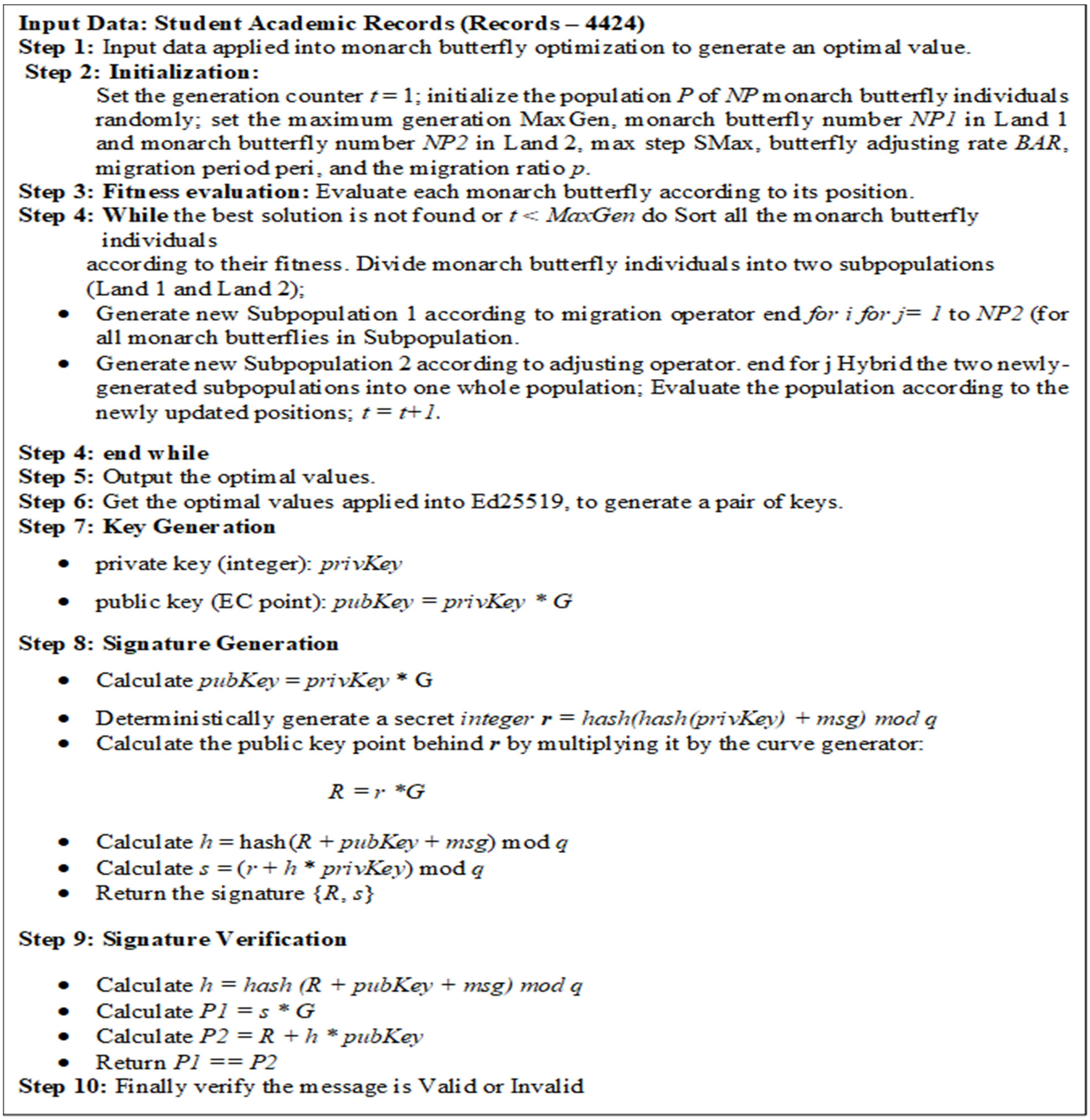


Figure 5: Proposed Flow Diagram of MBO-Ed25519

Algorithm 3: Proposed (MBO-Ed25519)



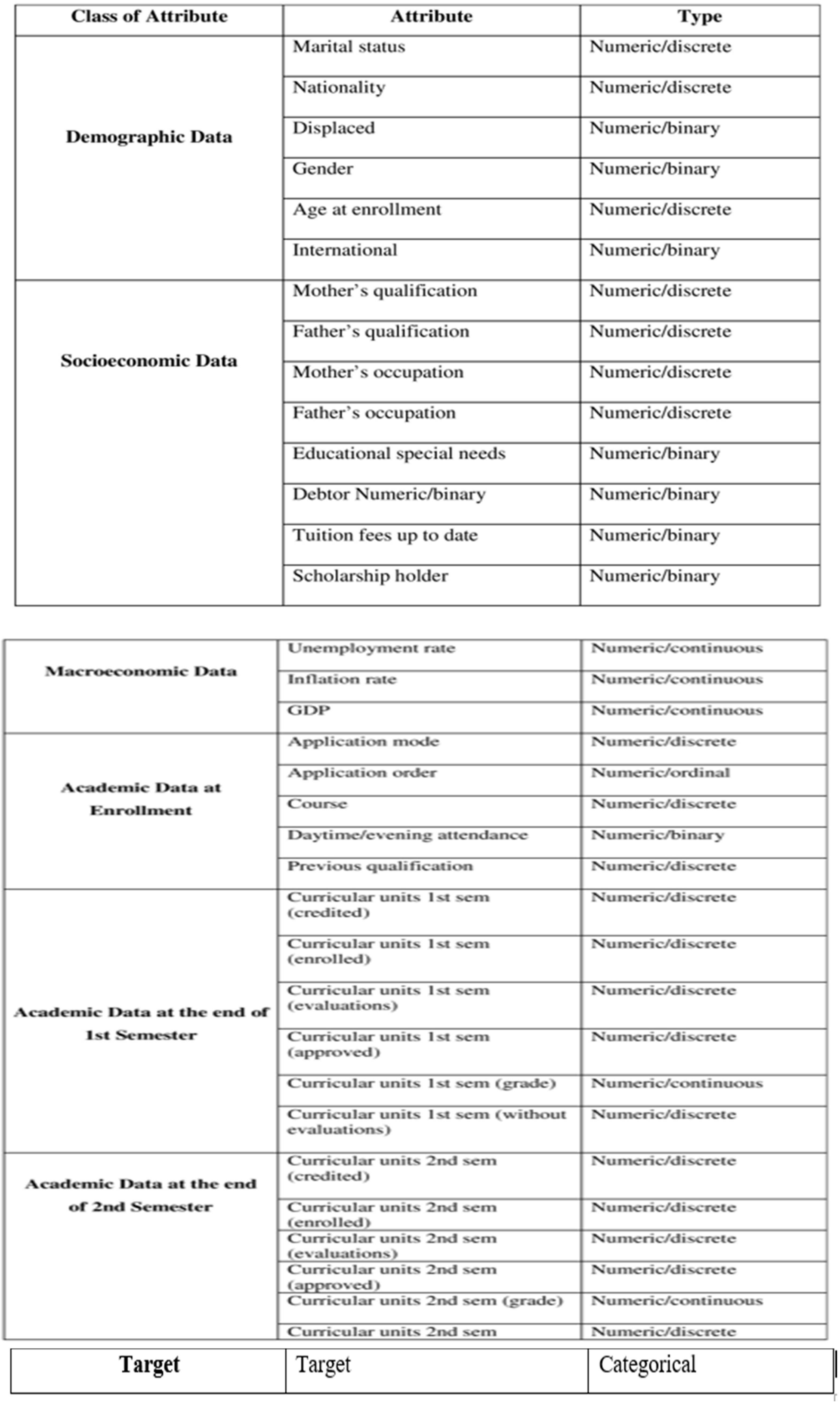
```
Input Data: Student Academic Records (Records – 4424)
Step 1: Input data applied into monarch butterfly optimization to generate an optimal value.
Step 2: Initialization:
        Set the generation counter $t = 1$; initialize the population $P$ of $NP$ monarch butterfly individuals
        randomly; set the maximum generation MaxGen, monarch butterfly number $NP1$ in Land 1
        and monarch butterfly number $NP2$ in Land 2, max step SMax, butterfly adjusting rate $BAR$,
        migration period peri, and the migration ratio $p$.
Step 3: Fitness evaluation: Evaluate each monarch butterfly according to its position.
Step 4: While the best solution is not found or $t < MaxGen$ do Sort all the monarch butterfly
          individuals
       according to their fitness. Divide monarch butterfly individuals into two subpopulations
       (Land 1 and Land 2);
    •   Generate new Subpopulation 1 according to migration operator end for i for j= 1 to NP2 (for
        all monarch butterflies in Subpopulation.
    •   Generate new Subpopulation 2 according to adjusting operator. end for j Hybrid the two newly-
        generated subpopulations into one whole population; Evaluate the population according to the
        newly updated positions; $t = t+1$.

Step 4: end while
Step 5: Output the optimal values.
Step 6: Get the optimal values applied into Ed25519, to generate a pair of keys.
Step 7: Key Generation
    •   private key (integer): privKey
    •   public key (EC point): $pubKey = privKey * G$
Step 8: Signature Generation
    •   Calculate $pubKey = privKey * G$
    •   Deterministically generate a secret integer $r = hash(hash(privKey) + msg) \bmod q$
    •   Calculate the public key point behind $r$ by multiplying it by the curve generator:

                              $R = r * G$

    •   Calculate $h = \text{hash}(R + pubKey + msg) \bmod q$
    •   Calculate $s = (r + h * privKey) \bmod q$
    •   Return the signature $\{R, s\}$
Step 9: Signature Verification
    •   Calculate $h = hash\ (R + pubKey + msg)\ mod\ q$
    •   Calculate $P1 = s * G$
    •   Calculate $P2 = R + h * pubKey$
    •   Return $P1 == P2$
Step 10: Finally verify the message is Valid or Invalid
```

4 EXPERIMENTAL SETUP

4.1 Data Collection

Records of students [3] enrolled between 2008/2009 and 2018/2019 (after the Bologna Process was applied to higher education in Europe). The data was taken from the UCI website. A total of 4424 academic records and 35 attributes are included in the dataset. A description of the types of attributes can be found in Table III.

Table 3: Type of Attributes [3]



Class of Attribute	Attribute	Type
Demographic Data	Marital status	Numeric/discrete
	Nationality	Numeric/discrete
	Displaced	Numeric/binary
	Gender	Numeric/binary
	Age at enrollment	Numeric/discrete
	International	Numeric/binary
Socioeconomic Data	Mother's qualification	Numeric/discrete
	Father's qualification	Numeric/discrete
	Mother's occupation	Numeric/discrete
	Father's occupation	Numeric/discrete
	Educational special needs	Numeric/binary
	Debtor Numeric/binary	Numeric/binary
	Tuition fees up to date	Numeric/binary
	Scholarship holder	Numeric/binary
Macroeconomic Data	Unemployment rate	Numeric/continuous
	Inflation rate	Numeric/continuous
	GDP	Numeric/continuous
Academic Data at Enrollment	Application mode	Numeric/discrete
	Application order	Numeric/ordinal
	Course	Numeric/discrete
	Daytime/evening attendance	Numeric/binary
	Previous qualification	Numeric/discrete
Academic Data at the end of 1st Semester	Curricular units 1st sem (credited)	Numeric/discrete
	Curricular units 1st sem (enrolled)	Numeric/discrete
	Curricular units 1st sem (evaluations)	Numeric/discrete
	Curricular units 1st sem (approved)	Numeric/discrete
	Curricular units 1st sem (grade)	Numeric/continuous
	Curricular units 1st sem (without evaluations)	Numeric/discrete
Academic Data at the end of 2nd Semester	Curricular units 2nd sem (credited)	Numeric/discrete
	Curricular units 2nd sem (enrolled)	Numeric/discrete
	Curricular units 2nd sem (evaluations)	Numeric/discrete
	Curricular units 2nd sem (approved)	Numeric/discrete
	Curricular units 2nd sem (grade)	Numeric/continuous
	Curricular units 2nd sem	Numeric/discrete
Target	Target	Categorical

4.2 Performance Metrics

4.2.1 Encryption Time

It refers to the time taken to encrypt a message or data. It is usually measured in seconds.

$$Encryption\ time = \frac{Size\ of\ data\ to\ be\ encrypted}{Encryption\ speed} \qquad (5)$$

4.2.2 Decryption Time

It refers to the time taken to decrypt an encrypted message or data. It is usually measured in seconds.

$$Decryption\ time = \frac{Size\ of\ data\ to\ be\ decrypted}{Decryption\ speed} \qquad (6)$$

4.2.3 Total Time

It refers to the total time taken for encryption and decryption, including any setup time required for key generation or other pre-processing steps.

$$Total\ time = Encryption\ time + Decryption\ time \qquad (7)$$

4.2.4 Throughput

It refers to the amount of data that can be encrypted or decrypted in a given amount of time.

$$Throughput\ of\ encryption = \frac{Total\ plain\ text}{Encryption\ time\ (second)} \qquad (8)$$

4.2.5 Energy Consumption

It refers to the amount of energy consumed during the encryption and decryption process. It is usually measured in joules or watts.

$$Energy\ consumption\ (in\ joules) = Power\ (in\ watts)\ x\ Time\ (in\ seconds) \qquad (9)$$

4.3 Results

A performance evaluation is based on Encryption Time, Decryption Time, Total Time, Throughput, and Energy Consumption. We compare the performance of EdDSA, Ed448 and Ed25519. It achieves a lower complexity of Encryption Time of 14.07(s) using the proposed (MBO-Ed25519) approach.

Table 4: Performance Analysis

Algorithm	**Number of Records**	**Encryption Time (s)**	**Decryption Time (s)**	**Total Time(s)**	**Throughput**	**Energy Consumption**
EdDSA	4424	39.04	28.95	67.99	89.6857	73.3957
Ed448		35.85	23.72	59.57	73.2895	69.8902
Ed25519		32.72	23.95	56.67	72.8965	67.2925
Proposed (MBO - Ed25519)		14.07	20.67	34.74	55.4829	51.9876

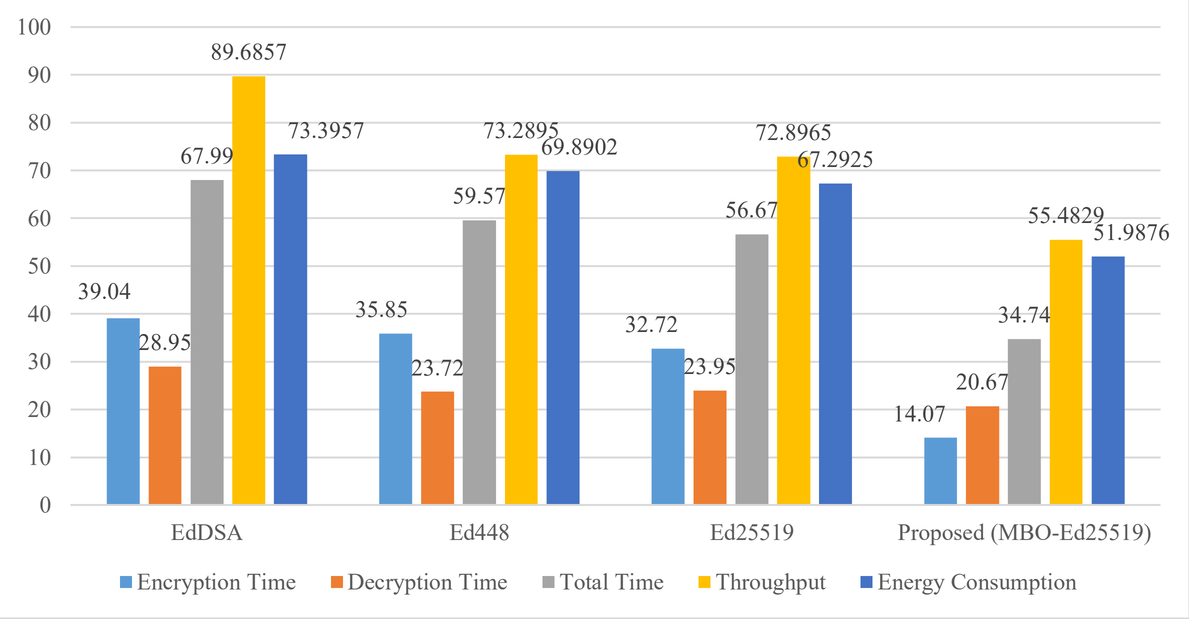


Figure 6: Performance Analysis of Graphical Representation

5 CONCLUSION

In the proposed hybrid (MBO-Ed25519) algorithm, the Monarch Butterfly Optimization algorithm is combined with the Edwards25519 algorithm and it has several advantages. Because it combines the advantages of existing algorithms, it can be applied to a high complexity time. An optimization technique is used to generate optimal values from the data in order to select the most appropriate attributes. To enhance the security of the data, another Ed25519 signature used optimal values. We obtained the highest performance among these algorithms with our algorithm. It overcame all the other algorithms with a shorter time frame, increasing the level of security. The proposed (MBO-Ed25519) approach shows significant improvements against three closely related techniques, achieving a less encryption time of 14.07(s). Using the student academic records dataset, our approach proved efficient in monarch butterfly optimization. Data security is less complex with our model.

REFERENCES

1. Alweshah, Mohammed, et al. "The monarch butterfly optimization algorithm for solving feature selection problems.", Neural Computing and Applications, pp. 1-15, 2020.
2. Bisheh-Niasar, Mojtaba, Reza Azarderakhsh, and Mehran Mozaffari-Kermani. "Cryptographic accelerators for digital signature based on Ed25519.", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 29 Issue.7, pp. 1297-1305, 2021.
3. Binh Kieu-Nguyen, Cuong Pham-Quoc, Ngoc-Thinh, Cong-Kha Pham and Trong- Thuc Hoang, "Low-Cost Area -Efficient FPGA-based Multi-Functional ECDSA/EdDSA", Cryptography, Vol. 3, Issue. 2, pp. 25, May 2022.
4. Elhoseny, Mohamed, et al. "Hybrid optimization with cryptography encryption for medical image security in Internet of Things." Neural computing and applications, Vol. 32, pp. 10979-10993, 2020.
5. Feng, Yanhong, et al. "Monarch butterfly optimization: a comprehensive review.", Expert Systems with Applications, pp. 1-27, 2021.
6. Faz-Hernández, Armando, Julio López, and Ricardo Dahab. "High-performance implementation of elliptic curve cryptography using vector instructions.", ACM Transactions on Mathematical Software (TOMS), Vol. 45, Issue. 3, pp. 1-35, 2019.
7. Huang, Shixin, et al. "Application of Improved Monarch Butterfly Optimization for Parameters' Optimization.", Mathematical Problems in Engineering, pp. 1-10, 2023.
8. Javed R. Shaikh, Marina Nenova, Georgi Iliev and Zlatka Valkova-Javis, "Analysis of standard elliptic curves for the implementation of elliptic curve cryptography in resource-constrained E-commerce applications", 2017 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems (COMCAS). IEEE, pp. 1-4, 2017.
9. Kumar, Manish, Akhlad Iqbal, and Pranjal Kumar. "A new RGB image encryption algorithm based on DNA encoding and elliptic curve Diffie–Hellman cryptography.", Signal Processing, Vol. 12, Issue. 5, pp. 187-202, 2016.
10. Md. Navid Bin Anwar, Mahmud Hasan, Md. Mahade Hasan, Jafrin Zafar Lorenand S.M.Tanjim Hossain, "Comparative Study of Cryptography Algorithms and its Applications", International Journal of Computer Networks and Communications Security, Vol. 7, Issue. 5, pp. 96-103, May2019.
11. Nagaraj, G., et al. "Enhancing performance of cell formation problem using hybrid efficient swarm optimization.", Soft Computing, Vol. 24, pp. 16679-16690, 2020.
12. Nagarajan, G., and K. Sampath Kumar. "A Novel Monarch Butterfly Optimization with Attribute based Encryption for Secure Public Cloud Storage.", Vol. 12, Issue. 4, pp. 1044-1054, 2021.
13. Verri Lucca, Arielle, et al. "A review of techniques for implementing elliptic curve point multiplication on hardware.", Journal of Sensor and Actuator Networks, Vol. 10, Issue. 3, 2020.
14. Omar S. Saleh, Osman ghanzali and Qusay AL Maatouk, "Graduation Certificate Verification Model: A Preliminary Study", International Journal of Advanced Computer Science and Application s, Vol. 10, Issue. 7, pp. 575-582, 2019.
15. Realinho, Valentim, et al. "Predicting Student Dropout and Academic Success." Data, Vol. 7, Issue .146, pp. 1-17, 2022.
16. Sagar Hossen, Md, et al. "Digital signature authentication using asymmetric key cryptography with different byte number.", Evolutionary Computing and Mobile Sustainable Networks: Proceedings of ICECMSN 2020. Springer Singapore, pp. 1-35, 2021.
17. Sghaier, Anissa, et al. "Design and implementation of low area/power elliptic curve digital signature hardware core.", Electronics, Vol. 6, Issue. 46, pp. 1-23, 2017.
18. Sasdrich, Pascal, and Tim Güneysu. "Efficient elliptic-curve cryptography using Curve25519 on reconfigurable devices.", Reconfigurable Computing: Architectures, Tools, and Applications: 10th International Symposium, Proceedings 10. Springer International Publishing, pp. 1-13, 2014.
19. Shankar, Gauri, et al. "Improved Multisignature Scheme for Authenticity of Digital Document in Digital Forensics Using Edward-Curve Digital Signature Algorithm.", Security and Communication Networks, pp. 1-18, 2023.

20. Shankar, K. and Perumal Eswaran. “An Efficient Image Encryption Technique Based on Optimized Key Generation in ECC Using Genetic Algorithm.”, Artificial Intelligence and Evolutionary Computations in Engineering Systems, pp. 705 -714, 2016.
21. Verri Lucca, Arielle, et al. "A review of techniques for implementing elliptic curve point multiplication on hardware.", Journal of Sensor and Actuator Networks, Vol. 10, Issue. 3, pp. 1-17, 2020.
22. Yi, Jiao-Hong, Jian Wang, and Gai-Ge Wang. "Using monarch butterfly optimization to solve the emergency vehicle routing problem with relief materials in sudden disasters.", Open Geosciences, Vol. 11, Issue. 1, pp. 391-413, 2019.
23. Zhou, Huan, et al. "A hybrid butterfly optimization algorithm for numerical optimization problems.", Computational Intelligence and Neuroscience, pp. 1-14, 2021.